electronics

MDPI

Article

A Study on Toponymic Entity Recognition Based on Pre-Trained Models Fused with Local Features for Genglubu in the South China Sea

Yinwei Wei, Yihong Li * and Xiaoyi Zhou

School of Cyberspace Security, Hainan University, Haikou 570228, China;
21220854000097@hainanu.edu.cn (Y.W.); xy.zhou@hainanu.edu.cn (X.Z.)
* Correspondence: 990638@hainanu.edu.cn

Abstract: Toponymic entity recognition is currently a critical research hotspot in knowledge graphs. Under the guidance of the national ancient book protection policy and the promotion of the wave of digital humanities research, this paper proposes a toponymic entity recognition model (ALBERT-Conv1D-BiLSTM-CRF) based on the fusion of a pre-trained language model and local features to address the problems of toponymic ambiguity and the differences in ancient and modern grammatical structures in the field of the Genglubu. This model extracts global features with the ALBERT module, fuses global and local features with the Conv1D module, performs sequence modeling with the BiLSTM module to capture deep semantics and long-distance dependency information, and finally, completes sequence annotation with the CRF module. The experiments show that while taking into account the computational resources and cost, this improved model is significantly improved compared with the benchmark model (ALBERT-BiLSTM-CRF), and the precision, recall, and F1 are increased by 0.74%, 1.28%, and 1.01% to 98.08%, 96.67%, and 97.37%, respectively. The model achieved good results in the field of Genglubu.

Keywords: toponymic entity recognition; Genglubu corpus; pre-trained language model; local feature; digital humanities

Citation: Wei, Y.; Li, Y.; Zhou, X. A Study on Toponymic Entity Recognition Based on Pre-Trained Models Fused with Local Features for Genglubu in the South China Sea. *Electronics* **2024**, *13*, 4. https://doi.org/10.3390/electronics13010004

Academic Editors: Lanting Fang and Yubo Song

Received: 9 November 2023
Revised: 11 December 2023
Accepted: 15 December 2023
Published: 19 December 2023

1. Introduction

Genglubu is a kind of ancient navigation guidebook formed through fishermen's long-term practical activities and accumulated experience in fishery production on the islands of the South China Sea and its waters [1]. As a national intangible cultural heritage, Genglubu has constructed a perfect navigation system for the South China Sea, recording numerous geographical entities, including features such as shape and orientation, as well as a large amount of information such as the distance between islands and reefs, the direction of sea currents, and navigational relationships, among which the toponymic entities are the most valuable type of geographic information resources [2]. A toponym is an exclusive name that refers to a specific spatial location [3]. Its naming is usually influenced by a variety of factors, such as history, culture, geology, climate, and folklore, and reflects the comprehensive characteristics of a region in terms of natural conditions, human history, and social culture, and plays an essential role in various fields such as territorial management, mapping, navigation, tourism, and cultural heritage [4]. Toponymic entity recognition (TER), which refers to determining the location boundaries of toponymic entities from natural language texts and making type judgments about them [5], is a subset of the Named Entity Recognition (NER) task, which is of great significance as an upstream task for constructing a knowledge graph in the geography field. Most of the existing research in Genglubu has been conducted from a traditional humanities and social sciences perspective and technology. It has several problems, including a lack of clear knowledge organization and weak correlation between geographic entities. The Opinions on Promoting the Work of Ancient

Books in a New Era states that new technologies should be used to realize the digitization and intelligence of ancient books [6]. Knowledge graphs can not only help the implicit relationships and enhance the correlation between data but can also reduce the difficulty of knowledge dissemination in related fields.

For low-resource domains such as Genglubu, under the guidance of national policy and the promotion of artificial intelligence technology, this paper designs and constructs a toponymic entity recognition dataset (Genglubu Toponym Data, GTData) for the TER task in Genglubu. Meanwhile, we propose a toponymic entity recognition model (ALBERT-Conv1D-BiLSTM-CRF) and analyze and compare it with other models to obtain superior results. It aims to provide a database for the construction of geographic knowledge graphs of the South China Sea and to promote cross-disciplinary development.

The remainder of this paper is organized as follows. In Section 2, we introduce the literature review. Section 3 describes the model structure in detail. Section 4 describes the experimental setup and the analysis of the results. Section 5 summarizes the results.

2. Literature Review

In recent years, deep learning models have been effective in areas such as CV and NLP [7,8], especially in TER [9,10]. Compared with traditional entity recognition methods, deep learning can automatically learn critical features and higher-order abstract features from the original dataset, avoiding the need for domain experts to define rules or carry out complex feature engineering manually. Because it can use many parameters, it has apparent advantages in applying deep semantic knowledge and alleviating data sparsity [11]. It maps raw textual data into a vector or matrix space. It maps words to their corresponding entity classes using different neural networks [12]. The toponymic entity recognition model based on deep learning mainly comprises an embedding layer, a feature encoding layer, and a label decoding layer. Among them, the feature encoding layer mainly includes Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM), Gated Recurrent Units (GRUs), as well as Hybrid Neural Networks and Attention Mechanisms. CNNs can increase computation speed by parallelizing data processing and, therefore, have a faster computational efficiency. Gritta M et al. proposed a new approach to systematically encoding geographic metadata in conjunction with CNNs [13]. This approach involves converting place names in natural language text into corresponding latitude and longitude coordinates and combining them with map information to improve the robustness of the model with a joint training approach. Kumar A et al. faced the problems of unreliable fields, grammatical errors, and non-standard abbreviations of place name information on Twitter. They proposed a CNN-based model that extracts geolocation information from Twitter and achieves an F1 of 96.0% [14]. CNNs have the problem of missing contextual information when processing long text or sequence data. RNNs are superior to CNNs in terms of performance in processing sequence data and can better capture the dependencies between sequence data. By analytically comparing ARIMA, LSTM, and BiLSTM models, Siami-Namini et al. verified that training data in the opposite direction helps sequence modeling and can significantly improve the accuracy of time series [15]. Chen T et al. proposed a divide-and-conquer approach by first classifying sentences into three different types using Bilstm-CRF and then utilizing 1d-CNNs to perform sentiment analysis on each type of sentence, which effectively improves the performance of sentence-level sentiment analysis [16]. Shen Si et al. proposed an RNN-based Chinese character-level annotation model by combining RNNs and Chinese word features, which significantly improved the F1 value of toponymic entities [17]. Rhanoui M et al. addressed the problems of large data size and conflicting viewpoints in the task of document-level sentiment analysis by constructing a CNN-BiLSTM model for extended text viewpoint analysis using word embedding in Doc2vec. They obtained excellent results on a French newspaper article [18]. Peng N et al. attempted to train NER and disambiguation as a joint task using an LSTM-CRF model, which improved the F1 value by almost 5% on the results of previous studies [19]. Lu W et al. constructed a

model for prediction by combining CNNs, BiLSTM, and Attention Mechanisms for nonlinear time series such as stock price and obtained better results [20]. Dong C et al. constructed a BiLSTM-CRF model based on character-level and part-level feature representations for Chinese-named entity recognition. They achieved the best F1 value of 90.95% on the MSRA dataset [21]. With the development of pre-trained language models (PTLM) such as BERT, PTLM can capture most of the semantic information of Chinese characters in a better way compared with previous studies. Ning X et al. introduced bi-directional attention routing and sausage measure to project data onto complex surfaces with nonlinear mapping, which enables the approximation of any nonlinear function with arbitrary accuracy and maintains the local responsiveness of the capsule entities. The experimental results are excellent [22]. Ma K et al. proposed a neural network model based on BERT-BiLSTM-CRF for Chinese place name entity recognition, which performs well on MSRA, GeoTR-20, and other datasets [23]. Ziniu W et al. proposed a hybrid neural network model based on BERT to address the problems of not fully considering the context and ignoring the local features in the NER task by combining the BiLSTM and IDCNN models to extract the features, which resulted in a 4.79% improvement in the F1 value compared with the baseline model on the CLUENER dataset [24].

Deep learning has made superior progress in TER tasks. However, problems that need to be addressed, such as data scarcity, difficulty in contextualizing contextual understanding, and place name ambiguity. These problems mentioned above are also reflected in the field of Genglubu. First, Genglubu has high scarcity as it is a kind of literature containing unique geographical features. Its data samples are small, and the data available for training are limited. These problems cause difficulties for the model in discovering hidden features in the data. Second, the way of documentation in Genglubu differs from modern times, especially the rich contextualized information in Chinese. Finally, the phenomenon that the same name can be used as both a place name and an orientation exists in Genglubu, increasing the ambiguity in the corpus. Based on the above research results, this paper utilizes deep learning models to research toponymic entity recognition in Genglubu.

3. Methods

3.1. Architecture of the Model

The ALBERT-Conv1D-BiLSTM-CRF toponymic entity recognition model proposed in this paper consists of four main components. From top to bottom are the global feature extraction module, local feature extraction module, sequence modeling module, and decoding module, as shown in Figure 1. First, the global feature extraction module consists of the ALBERT layer. It is mainly responsible for mapping text sequences into vectors in a high-dimensional vector space and learning high-quality features and global semantic information of the input text. Second, the local feature extraction module consists of the Conv1D layer. It performs a convolution operation on the output of the ALBERT layer to extract local features that may involve individual characters or phrases by sliding a learnable convolution kernel over the sequence. Again, the sequence modeling module consists of BiLSTM layers. It mainly performs bi-directional modeling of the outputs of the upper layer to better capture the deep semantics and long-distance dependency information of the sequences and further improve the model's ability to characterize the sequences. Finally, the decoding module consists of the CRF layer. It accepts the output sequences from the BiLSTM layer. It improves the accuracy and robustness of the sequence labeling task by calculating the label scores and transfer probabilities to obtain the optimal labeled sequences. Each layer takes on a different function and has a clear order to work together to accomplish the sequence labeling task.

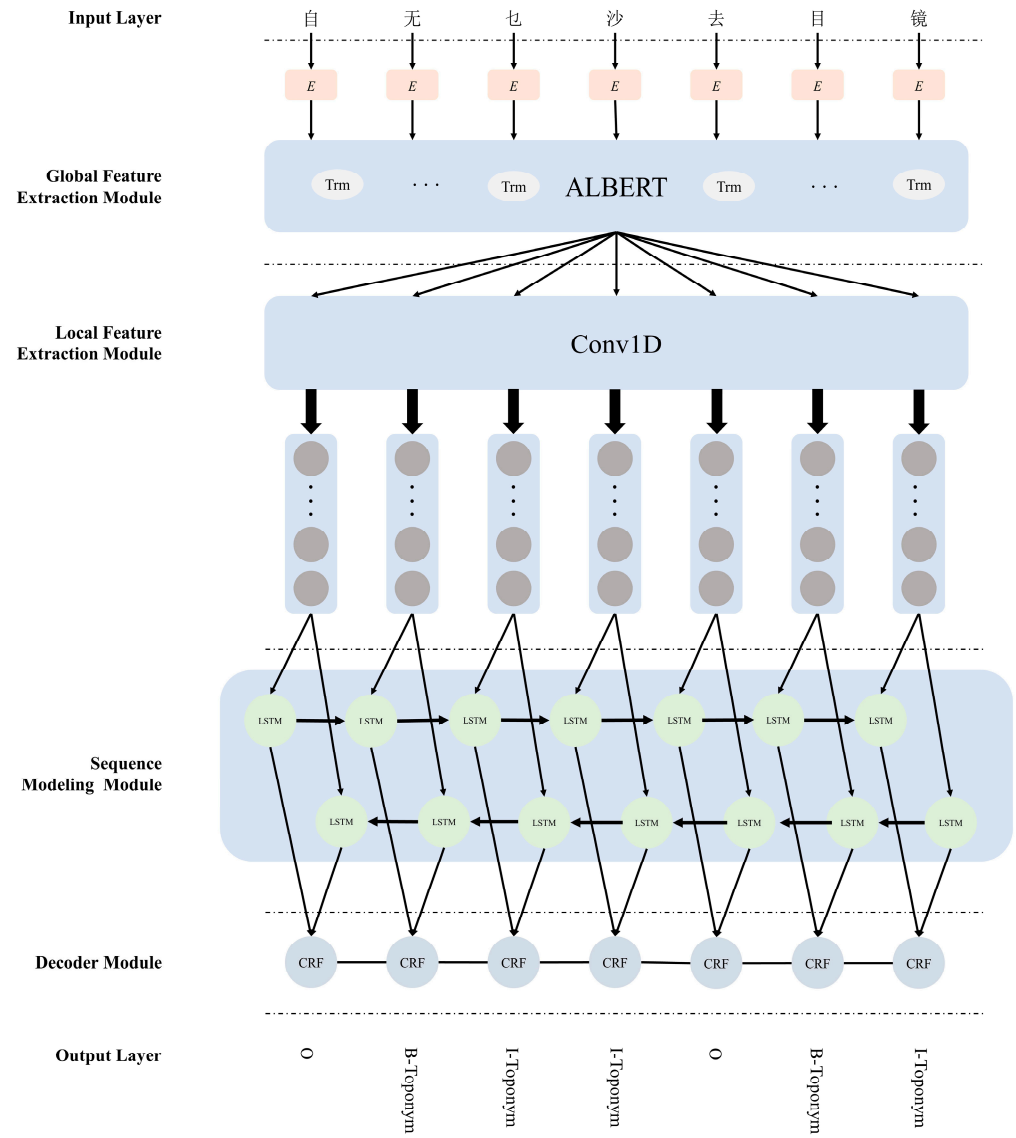

Figure 1. The overall framework of the model.

3.2. ALBERT

A Lite BERT (ALBERT) model is a pre-trained language model that was improved and optimized based on the BERT model, which allows ALBERT to perform better in some natural language processing tasks [25,26]. The ALBERT model uses techniques such as parametric factorization of embedding vectors and cross-layer parameter interactions to reduce the number of parameters significantly, thus improving the model's training speed and generalization performance. It also introduces the Sentence Order Prediction (SOP) task instead of the Nest Sentence Prediction (NSP) task to improve the performance of the downstream tasks [27]. It has the same basic structure as the BERT model: a deep bi-directional coded representation model based on the Transformer encoder. The multi-head attention mechanism in Transformer can make the same word in different sentences form different vector representations, which is effective for solving the multiple meanings of a word.

This paper uses the ALBERT layer to preprocess the input text. It is responsible for extracting the contextual semantic information of the sequence data as a global feature extraction module. Take data "自无乜线至深圈使壬丙三更" as an example, as shown in Figure 2. Suppose the input sequence is $X = (X_o, X_1, \ldots, X_n)$, where X_i denotes the ith token. For each X_i, token embedding, segment embedding, and position embedding are

used to obtain E^i_{token}, $E^i_{segment}$, $E^i_{position}$, respectively. The three embedding vectors are weighted and summed according to a specific ratio, and the computation process is as follows:

$$E = \alpha_1 E^i_{token} + \alpha_2 E^i_{segment} + \alpha_3 E^i_{position} \tag{1}$$

where $\alpha_1, \alpha_2,$ and α_3 denote the scaling factor of the three embedding vectors, generally being a real number between [0, 1] and satisfying $\sum_{j=1}^{3} \alpha_j = 1$.

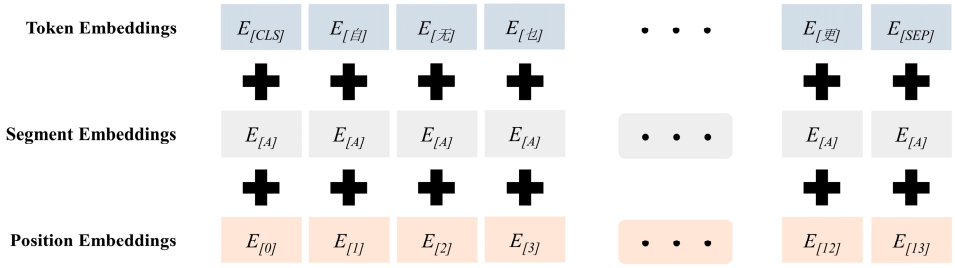

Figure 2. Input vector representation of the ALBERT model.

Then, the ALBERT layer learns the semantic information of each token in the context and is able to capture the long-range dependencies in the sequence. After the encoding process of the multi-layer Transformer, a high-dimensional vector representation $T = (T_0, T_1, \ldots, T_n)$ is obtained, and the specific learning process is shown in Figure 3.

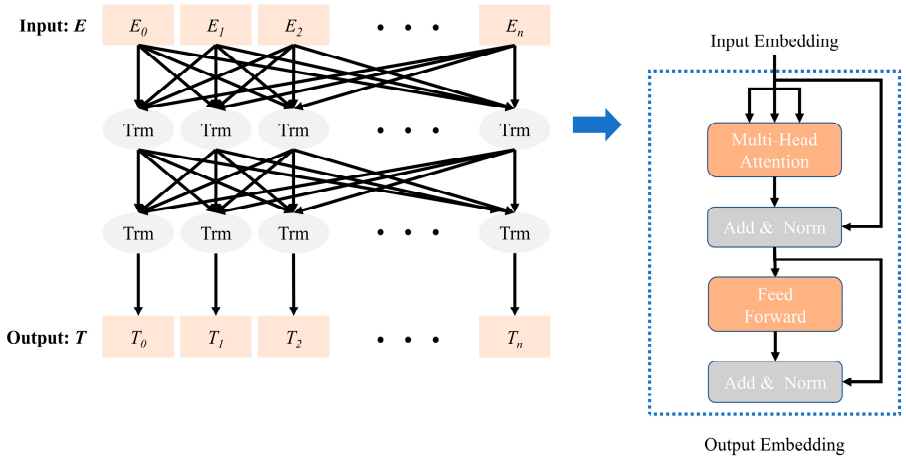

Figure 3. Structure of the ALBERT model.

Among them, the *i*-layer Transformer encoder calculation process is shown in Equations (2) to (6):

$$t'_i = Layer\ Norm(t_{i-1}) \tag{2}$$

$$t''_i = Multi\ Head\ Attention(t'_i, t'_i, t'_i) \tag{3}$$

$$t'''_i = Layer\ Norm(t'_i + t''_i) \tag{4}$$

$$t''''_i = Feed\ Forward(t'''_i) \tag{5}$$

$$t_i = Layer\ Norm(t'''_i + t''''_i) \tag{6}$$

where *Layer Norm* is the normalization operation that reduces the effect of internal variable displacements, thus making the model more stable. *Multi Head Attention* is used to

capture deep information in the input sequence better. *Feed Forward* is the feed-forward fully connected network that improves the Transformer's nonlinear modeling capabilities and increases the model's degrees of freedom and its representational capabilities [28].

Since Genglubu belongs to the low-resource domain and small-scale dataset, to reduce the risk of overfitting and improve the generalization performance and the training efficiency, the parameters of the ALBERT layer are frozen during the training process. Freezing the ALBERT module makes the training focus on the task-specific layer you added, helping the model better adapt to name entity recognition.

3.3. Conv1D

Conv1D (1D Convolution) is a one-dimensional convolutional layer used in this paper as a local feature extraction module to extract local features from the input sequence with convolution operation [29]. The ALBERT model has excellent results in extracting global features of the input sequence but falls short in extracting local features. When analyzing the corpus features, the local features are crucial in the field of Genglubu, and an enhancement in the local features helps to classify the toponymic entities more accurately. In addition, the spatial structure of the input sequences can be learned, thus helping to improve the model's generalization and reduce the overfitting problem.

After accepting the output vector $T = (T_0, T_1, \ldots, T_n)$ as the input of this module, a series of convolution operations were performed to extract the local features of the sequence and obtain a new feature sequence with more expressive capability $C = (C_0, C_1, \ldots, C_n)$. The computation process is shown in Equations (7) and (8). Assuming that the convolution kernel size is k and the number of output channels is c, the convolution output can be expressed as Equation (7):

$$h = Conv1D(T, W) + b \tag{7}$$

where $W \in R^{k \times d \times C}$ is the convolution kernel tensor and $b \in R^c$ is the bias vector. For the ith output channel, the convolution operation can be expressed as Equation (8):

$$h_{i,j} = relu\left(\sum_{s=1}^{k}\sum_{t=1}^{d} W_{s,t,i} \times T_{j+s-1,t}\right) \tag{8}$$

where $j = 1, 2, \ldots, n-k+1$ denotes the starting position of the convolution operation, and each position j will be given a feature vector $h_j \in R^c$ of c dimension. Based on the above calculation, the new feature sequence is $C = (C_0, C_1, \ldots, C_n)$.

3.4. BiLSTM

Long Short-Term Memory (LSTM) is a recurrent neural network model that processes sequential data. It consists of a recurrent unit, an input gate, a forgetting gate, and an output gate. It effectively solves the problem of gradient vanishing and gradient explosion that occurs when training on long sequences by introducing a gating mechanism that can memorize the information of the input sequence for a long time [30]. BiLSTM consists of a forward LSTM and a backward LSTM, where the forward LSTM processes the input sequence in chronological order, and the backward LSTM processes the input sequence in reverse chronological order. Combining the outputs of the two directions gives the bidirectional hidden state at each moment [31]. It can capture long-term dependencies in sequences better than unidirectional LSTM and improves the expressive power. Here, the transformer is not used instead of BiLSTM. This is because the transformer is based on attention, which weakens the position information in the computation process (relying only on position embedding). Yan H et al. experimentally verified that BiLSTM outperforms Transformer in the NER task, where relative positional information and orientation information are essential, so the BiLSTM model was chosen [32,33]. The structure of the BiLSTM model is shown in Figure 4. The output $C = (C_0, C_1, \ldots, C_n)$ of the Conv1D layer is used as the input to this module. The long-term dependencies of the input sequences

are captured with the BiLSTM layer, and the output $B = (B_0, B_i, \ldots, B_n)$ is input to the next module.

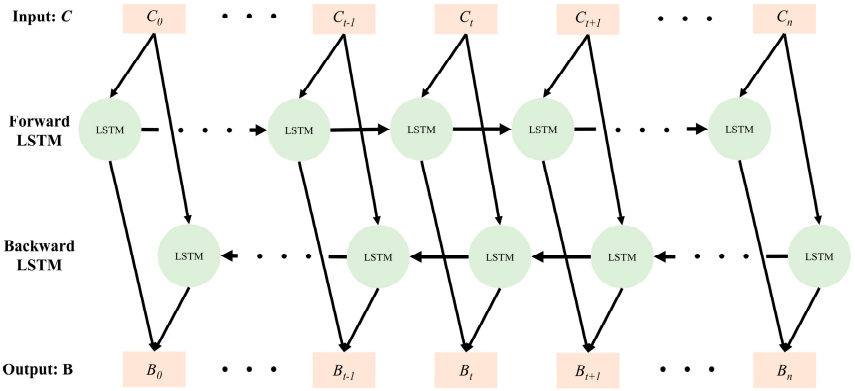

Figure 4. Structure of the BiLSTM model.

For the input sequence $C = (C_0, C_1, \ldots, C_n)$, the BiLSTM module deals with the forward and backward parts of the sequence data, using an LSTM model in different directions with independent parameters. The LSTM module uses C_i to denote the cell state. To avoid confusion and make the formulae clear, the input sequence $C = (C_0, C_1, \ldots, C_n)$ is temporarily expressed as $x = (x_0, x_1, \ldots, x_n)$. In the forward part, for the input x_t, the LSTM unit computes its hidden state and outputs it. Combining with Figure 5, the internal computation process can be expressed as Equations (9) to (14):

$$f_t = \sigma\left(W_f x_t + U_f h_{t-1} + b_f\right) \tag{9}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{10}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{11}$$

$$\widetilde{c_t} = tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{12}$$

$$C_t = f_t \odot C_{t-1} + i_t \odot \widetilde{c_t} \tag{13}$$

$$h_t = o_t \odot tanh(C_t) \tag{14}$$

where σ is the sigmoid function and $tanh$ is the hyperbolic tangent function, both are nonlinear activation functions; f_t, i_t, o_t are the activation values of the forget gate, input gate, and output gate, respectively; C_t is the cell state, and $\widetilde{c_t}$ is an important parameter used to update C_t, which can be understood as the candidate hidden state; W_f, W_i, W_o, and W_c are the weight matrices of the oblivion gate, the input gate, the output gate, and the cell state, respectively; U_f, U_i, U_o, and U_c are the weight matrices of the previous moment, respectively; and b_f, b_i, b_o, and b_c are the corresponding bias terms, respectively.

The above output is the output result of the forward part, so h_t can be understood as $h_t^{forward}$. The calculation process of the reverse part is similar to that of the forward part, and finally, the outputs of the forward and reverse parts are spliced together. The output result of the reverse part h_t is understood as $h_t^{backward}$, and the final representation is formed, as shown in Equation (15), which serves as the input of the downstream task:

$$B_t = \left[h_t^{forward}; h_t^{backward}\right] \tag{15}$$

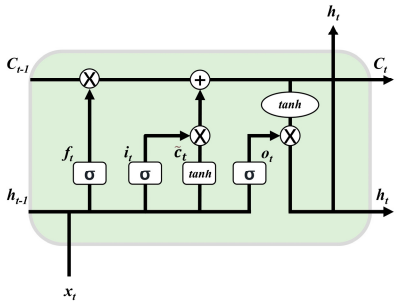

Figure 5. The cell structure of LSTM.

3.5. CRF

Conditional Random Field (CRF) is a discriminative model for predicting the output sequence from the input sequence in a sequence labeling task. It uses conditional probability distributions to model the dependencies between neighboring tags, normalizes the upstream output by learning the transfer probabilities between tags, makes a global optimization of the predicted sequence, and finally, solves for the optimal labeled sequence. In this paper, the CRF layer, as the last part of the model, receives $B = (B_0, B_i, \ldots, B_n)$ as input and gives the optimal labeling sequence by internal computation.

Suppose $B = (B_0, B_i, \ldots, B_n)$ is represented by $x = (x_0, x_1, \ldots, x_n)$ and the label sequence is denoted as $y = (y_0, y_1, \ldots, y_n)$, where y_i is the label corresponding to x_i. This conditional probability calculation process is as follows:

$$P(y|x) = \frac{exp(S(y,x))}{Z(X)} \tag{16}$$

where $Z(X)$ is the normalization factor, which is the sum of the scores computed for all possible labeled sequences and is used to convert the numerator part into a probability distribution. $S = (y, x)$ denotes the scores at a given labeled sequence y and input sequence x, which can be computed from the firing matrix E and the transfer matrix T, as in Equation (17):

$$S(y,x) = \sum_{i=0}^{n} \left(E_{i,y_i} + \sum_{j=0}^{n} T_{y_i,y_j} \right) \tag{17}$$

where the transfer matrix T represents the transfer probability from the previous label to the next label, and the firing matrix E is obtained by mapping the hidden state of each position of the input sequence into the label space. After obtaining the conditional probabilities, optimization algorithms such as stochastic gradient descent are used to maximize the log-likelihood function and update the model parameters. During the prediction process, the predictions are decoded using the Viterbi algorithm to find the labeled sequence predicted with maximum probability.

3.6. Loss Function

The loss function used in this paper is divided into two parts: the cross-entropy loss function and the CRF loss function. The cross-entropy loss function focuses on the model's accurate prediction of entity boundaries and categories. The parameters of the sequence annotation layer are updated by calculating the difference between the predicted sequence labels and the actual sequence labels:

$$Loss_1 = -\sum_{j=1}^{N} \sum_{i=1}^{C} y_{j,i} \ln(S_{j,i}) \tag{18}$$

where N is the number of samples, C is the number of categories, $y_{j,i}$ is the true value of the *i-th* category in the *j-th* sample, and $S_{j,i}$ is the predicted value of the *i-th* category in the *j-th* sample.

The CRF loss considers the dependencies between label sequences to ensure the overall soundness of the generated label sequences. It is modeled using a negative log-likelihood function. For a given input sequence x and actual label sequence y, the $Loss_2$ can be expressed as:

$$Loss_2 = -\ln(P(y|x)) \tag{19}$$

The overall loss function consists of a weighted sum of the two components.

$$Loss = \alpha_1 Loss_1 + \alpha_2 Loss_2 \tag{20}$$

where α_1 and α_2 are generally between [0, 1] and satisfy $\sum_{i=1}^{2} a_i = 1$.

4. Experiment

4.1. Datasets and Data Labeling

The initial data of the GTData dataset comes from more than twenty books of Genglubu, including *Su Deliu*, *Su Chengfen*, *Wang Shitao*, *Zheng Qingneng*, et al. We invited domain experts to annotate and check this dataset to ensure the quality of the dataset, and finally, check it manually as well as by code to ensure the uniqueness of the statements in the dataset. The GTData dataset is shown in Table 1.

Table 1. Genglubu Toponym Data dataset.

Classifications	Training Data	Test Data	Total Data
Number of sentences	2489	620	3109
Number of characters	42,797	10,776	53,573
Number of toponymic entities	5059	1262	6321

The GTData dataset has distinctive features compared with the publicly available datasets MSRA and People's Daily. In terms of average expected length (characters/sentence), GTData (17.23) is much lower than MSRA (48.39) and People's Daily (51.69). In terms of toponymic entity density (toponymic entities/sentence), GTData (2.03) is much higher than MSRA (0.82) and People's Daily (0.87). In addition, there are apparent prefix and suffix phenomena around place-name entities in GTData. Combined with the analysis of the characteristics of the era, these are because the ancient language is shorter and is more conducive to preservation and circulation, so the local characteristics in GTData are critical. The labeling method is the BIO labeling method, which is widely used in NER tasks. For example, for "自鸟仔峙去乙辛，用乙辛，二更收。", the specific annotation format is (O, B-Toponym, I-Toponym, I-Toponym, O, B-Toponym, I-Toponym, O, O, O, O, O, O, O, O, O, O, O, O, O, O, O), in which O denotes a non-toponym, B-Toponym denotes the beginning of the toponym, and I-Toponym denotes the interior of the toponym.

4.2. Experimental Setting and Assessment Indicators

The experimental environment of this paper is shown in Table 2.

Table 2. Experiment environment.

Experimental Environment	Configure
Operating system	Ubuntu Server 18.04 LTS 64 bit
CPU	Intel(R)Xeon(R)Platinum 8255C CPU @ 2.50 GHZ
GPU	NVIDIA T4
RAM	32 GB
Python	3.8.0
Pytorch	1.7.1

Precision (P), recall (R), and the F1-score (F1) were used as the evaluation indexes of the experimental results. Their calculation processes are as follows:

$$P = \frac{TP}{(TP + FP)} \tag{21}$$

$$R = \frac{TP}{(TP + FN)} \tag{22}$$

$$F1 = \frac{2 \times P \times R}{(P + R)} \tag{23}$$

where True Positive (TP) is the number of positive samples correctly predicted; True Negative (TN) is the number of negative samples correctly predicted; False Positive (FP) is the number of negative samples incorrectly predicted as positive samples; and False Negative (FN) is the number of positive samples incorrectly predicted as negative samples.

4.3. Parameter Setting

In order to compare the performance of each model, the parameter configurations were uniformly set as in Table 3, and the overfitting problem was avoided using the dropout mechanism [34].

Table 3. Important parameter configurations.

Parameter Name	Value
Hidden dim	768
Max_sequence_length	128
Transformer_layer_num	12
BiLSTM_layer_num	1
Learning rate	10^{-4}
Dropout	0.5
Batch size	32

4.4. Analysis of Experimental Results

According to the above experimental setup, the trend in each evaluation index with Epoch was obtained, as shown in Figure 6. It is clear from Figure 6a that BERT-BiLSTM-CRF, RoBERTa-BiLSTM-CRF, ALBERT-AM-LSTM-CRF, and the model in this paper have a faster decrease in the loss and can achieve lower loss values. In contrast, the benchmark model (ALBERT-BiLSTM-CRF) performs relatively poorly, worse than the previous four, regarding both the rate of loss decline and the value of loss. A faster loss decline indicates that the model can converge faster during the learning process, indicating that the model can learn the patterns and features in the data faster. A low loss value indicates that the model performs better on the test set, which can measure the generalization ability of a model.

Figure 6b represents the trend in precision with epoch, and it can be seen that the ALBERT-BiLSTM-CRF, RoBERTa-BiLSTM--CRF, and ALBERT-AM-LSTM-CRF models have lower accuracy and almost the same effect. In contrast, BERT-BiLSTM-CRF and this paper's model are more effective, especially at epoch = 18; the effect of these two models tends to stabilize, of which the effect of this paper's model reaches 98%, which is slightly better than the BERT-BiLSTM-CRF model. Compared with the benchmark model, the improvement is nearly 0.74%.

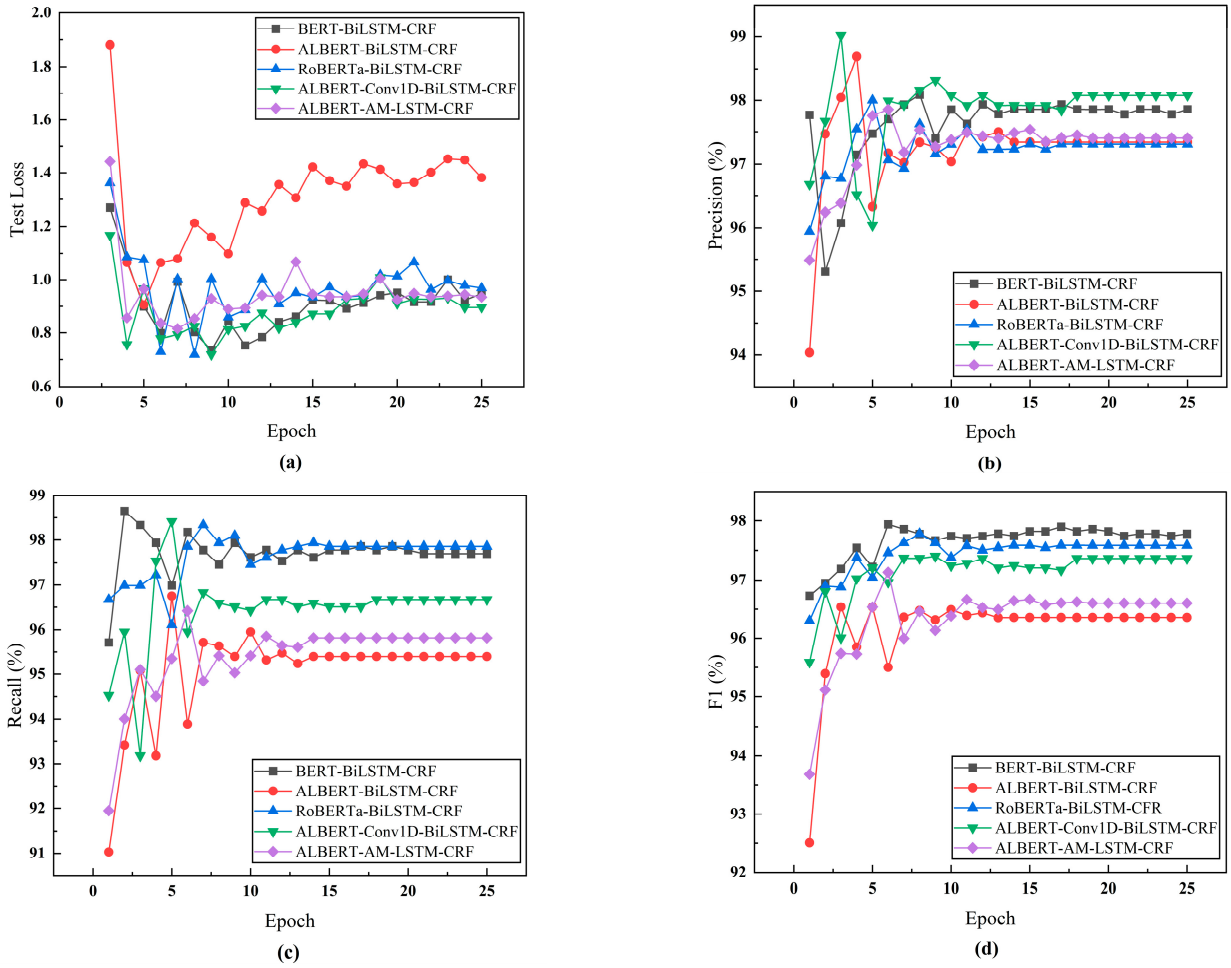

Figure 6. The change trend in each evaluation index with epoch. (**a**) represents the trend in Test Loss with epoch. (**b**) represents the trend in precision with epoch. (**c**) represents the trend in recall with epoch. (**d**) represents the trend in F1 with epoch.

Figure 6c represents the trend in recall with epoch, and it can be seen that the recall of ALBERT-BiLSTM-CRF and ALBERT-AM-LSTM-CRF is significantly lower than the other three models. The BERT-BiLSTM-CRF and RoBERTa-BiLSTM-CRF models have almost the same effect. The model in this paper shows a significant increase in recall compared with the benchmark model due to the introduction of the Conv1D model into the model structure, which can better capture the local features and thus enhance the recognition ability of the model and thus increases the recall from 1.28% to 96.67%. Although it is slightly insufficient compared with BERT-BiLSTM-CRF, it should be considered that in terms of parameters, the model in this paper is much smaller than the previous two models, which significantly saves computational consumption and has higher computational efficiency and lower cost. Therefore, the model in this paper is a superior choice under resource constraints.

Figure 6d represents the trend in F1 with epoch, and the F1-score is an essential indicator for evaluating the model to measure the overall comprehensive performance of the model. It can be seen that the F1 of ALBERT-BiLSTM-CRF and ALBERT-AM-LSTM-CRF have a large gap compared with the other three models. The BERT-BiLSTM-CRF and RoBERTa-BiLSTM-CRF models, as well as the model in this paper, have a more negligible difference in F1, and all perform better. By summarizing the data in the above figure, the performance can be obtained, as shown in Table 4.

Table 4. Effects of different models on toponym entity recognition.

Model	P (%)	R (%)	F1 (%)	Duration of Training (s/Epoch)
BERT-BiLSTM-CRF	97.86	97.70	97.78	66.26
ALBERT-BiLSTM-CRF	97.34	95.39	96.36	60.56
RoBERTa-BiLSTM-CRF	97.31	97.86	97.58	/
ALBERT-AM-LSTM-CRF	97.40	95.82	96.60	59.20
The model in this paper	98.08	96.67	97.37	61.59

We analyzed the four models regarding the loss function, precision, recall, and F1. In summary, in the task of toponymic entity recognition, this paper's model improves by 0.74%, 1.28%, and 1.01% in P, R, and F1 compared with the baseline model ALBERT-BiLSTM-CRF, respectively. As the model in this paper adds a local feature extraction layer based on the baseline model, which enhances the local features of the text data, there is an increase in computational resources, and the training time changes from 60.56 s/epoch to 61.59 s/epoch, which reduces the computational efficiency by 1.7%. Compared with the BERT-BiLSTM-CRF model, F1 is only 0.4% lower. Considering the computational resources and other aspects, the model in this paper is much smaller than the BERT-BiLSTM-CRF model in terms of the number of parameters, so the training efficiency is improved by nearly 7.05%. Therefore, the model in this paper has high performance while significantly saving computational resources and is more suitable for running on equipment with limited computational resources, thus improving the training speed and reducing the training cost.

4.5. Ablation Experiment

In order to further evaluate the impact of each module on model performance, ablation was designed to validate the results by removing the ALBERT, Conv1D, and BiLSTM modules, respectively. The experimental results are shown in Table 5.

Table 5. Results of the ablation experiments.

Model	P (%)	R (%)	F1 (%)
The model in this paper	**98.08**	**96.67**	**97.37**
* ALBERT	93.21	94.47	93.84
* Conv1D	97.34	95.39	96.36
* BiLSTM	95.22	95.31	95.26
* ALBERT-Conv1D	93.37	93.14	93.25
* Conv1D-BiLSTM	94.52	93.86	94.19

Note: * represents the removal of the module.

By ablating the ALBERT, Conv1D, and BiLSTM modules separately, it can be seen that each component contributes to model performance. Among them, the ALBERT module has the most significant impact on model performance, and its extracted contextual semantic information plays a vital role in recognizing place-name entities. Comparing the model with the removal of the BiLSTM module and the full model, we observe a decrease in the performance, which shows that the BiLSTM module plays a vital role in the long-distance dependencies in the whole sequence. In comparison, the Conv1D module had relatively little impact. With Conv1D as the control variable, the performance of the ALBERT-CRF, BiLSTM-CRF, and ALBERT-BiLSTM-CRF models all improved after adding the Conv1D module, which shows that the Conv1D module has a certain positive impact on dealing with the task of toponymic entity recognition in Genglubu.

4.6. Analysis of the Feature Vector

In the TER task, if the labels corresponding to two characters are of the same class, the final higher-order feature vectors of these two characters obtained after deep learning

model training will usually have some similarity. The model will make the samples of the same class closer in the vector representation space during the training process. T-SNE (t-distributed stochastic neighbor embedding) is a nonlinear dimensionality reduction technique that maps high-dimensional data into a low-dimensional space while preserving the relative distance between data points. It tends to map similar samples to neighboring locations in the reduced dimensional space, thus preserving the similarity relationship of the data [35].

Therefore, we take one batch of data for the case study of the feature vector (32 data points, a total of 616 characters, 65 characters of the B-toponym category, 97 characters of the I-toponym category, and 454 characters of the O-category). The corresponding feature vectors of the already-trained data are dimensionality-reduced with the t-SNE algorithm to see if meaningful features can be extracted. As shown in Figure 7, the samples corresponding to the same labels form three types of clusters. It can be seen that the model has extracted meaningful features to some extent.

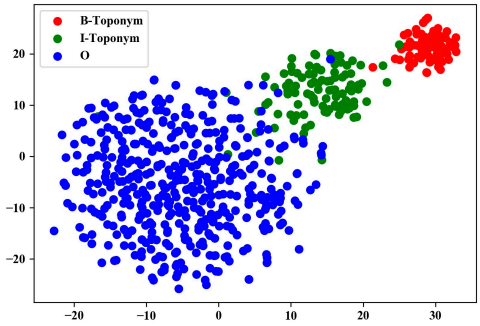

Figure 7. t-SNE visualization of feature vectors.

4.7. Sensitivity Analysis of Hyperparameters

We aimed to provide the performance of these models under different hyperparameter settings by varying the hyperparameters, including the learning rate, batch size, etc., to show the robustness of the proposed method. Due to the excessive permutations of different hyperparameters, we selected several parameter settings with common usage to demonstrate the method.

Based on the hyperparameter tuning experiments, it can be seen from the results in Table 6 that this paper's method exhibits a superior F1 performance than the ALBERT-BiLSTM-CRF model under different parameter settings (the enhancement ranges from 1.40% to 0.66%), which indicates that the model has a certain degree of robustness.

Table 6. Model performance with different hyperparameter settings (F1).

Model	Parameters [1]	Parameters [2]	Parameters [3]	Parameters [4]
BERT-BiLSTM-CRF	97.85	97.78	97.53	97.49
ALBERT-BiLSTM-CRF	96.33	96.36	95.62	96.52
RoBERTa-BiLSTM-CRF	97.35	97.58	97.62	97.42
ALBERT-AM-LSTM-CRF	96.44	96.60	96.27	96.35
The model in this paper	97.40	97.37	97.02	97.18

Note: The parameters are set as follows: [1] The parameters are set to learning rate = 10^{-4}, batch size = 16, epoch = 25, dropout = 0.5. [2] The parameters are set to learning rate = 10^{-4}, batch size = 32, epoch = 25, dropout = 0.5. [3] The parameters are set to learning rate = 10^{-4}, batch size = 64, epoch = 30, dropout = 0.5. [4] The parameters are set to learning rate = 10^{-3}, batch size = 32, epoch = 25, dropout = 0.3.

4.8. Model Generalization Validation

In order to validate the applicability and generalization ability of the model in this paper in the general domain, validation was carried out on the public datasets: the MSRA dataset and the People's Daily dataset, respectively. Since this paper focuses on the task of

toponymic entity recognition, a label reset operation was performed on the public dataset, where all non-toponymic entity labels were corrected to the label 'O' in order to facilitate high-quality evaluation of the model performance.

Figure 8a, b, and c shows the validation results for the GTData, MSRA, and People's Daily datasets, respectively. According to Figure 8b, the model in this paper reduces the precision by 1.01% compared with the ALBERT-BiLSTM-CRF model. However, the recall is improved by 2.17%, the F1 value is improved by 0.77%, and the overall performance is better than the latter. According to Figure 8c, including the pre-trained language model in the BiLSTM-CRF model has degraded the task in terms of performance. This may be because the ALBERT model does not match the features of the toponymic entity recognition task in some respects. It is better at handling sentence-level tasks, while the toponymic entity recognition task focuses more on entity-level tasks. In this paper, the Conv1D model is introduced after the ALBERT module, which makes the model focus on local features better, leading to an improvement in recognition. The experiments show that after introducing the Conv1D model, the precision is improved by 6.77%, the recall is improved by 8.21%, and the F1 is improved by 7.53% compared with the ALBERT-BiLSTM-CRF model. The model in this paper performs better on different datasets, indicating its good generalization and generalization ability.

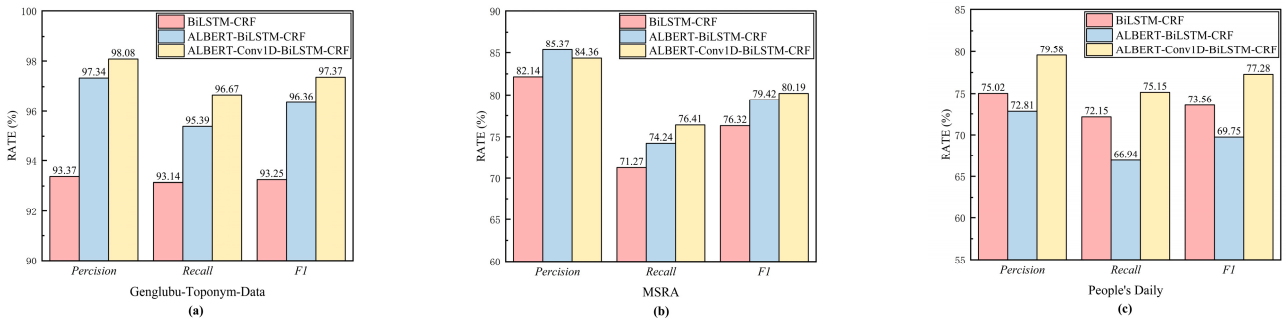

Figure 8. Model validation of common datasets. (**a**) is validation on the GTData dataset. (**b**) is validation on the MSRA dataset. (**c**) is validation on the People's Daily dataset.

5. Conclusions

Most of the existing research in the field of Genglubu in the South China Sea utilizes traditional social science and humanities tools and methods. It lacks effectiveness in the protection and dissemination of Genglubu data. Therefore, facing the problems of a lack of corpus, irregular geographical entity representation, and ambiguity of place names in Genglubu, this paper constructs the GTData dataset and proposes a toponymic entity recognition model based on the fusion of a pre-trained language model and local features (ALBERT-Conv1D-BiLSTM-CRF). Experiments show that compared with other deep learning models, the model in this paper achieves excellent performance on the GTData dataset and the public dataset while considering the computational resources and computational cost. This paper provides a new perspective for the research of Genglubu, helps to reduce the difficulty of its dissemination, and provides a new paradigm for the research of digital humanities. The next step focuses on expanding the featured literature corpus, granularizing the toponymic entities, and improving the model, aiming to improve the performance and robustness of the model and conduct relationship recognition studies to construct a geographic knowledge graph for the South China Sea.

Author Contributions: Conceptualization, Y.W.; methodology, Y.W.; software, Y.W.; validation, Y.W. and Y.L.; formal analysis, Y.W.; investigation, Y.W.; resources, Y.W.; data curation, Y.W.; writing—original draft preparation, Y.W.; writing—review and editing, Y.L. and X.Z.; visualization, Y.W.; supervision, Y.L. and X.Z.; project administration, Y.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Natural Science Foundation of China (No. 62362025, No. 62163010, No. 62162021) and the Hainan Province Key R&D plan project (No. ZDYF2022GXJS224).

Data Availability Statement: The data that support the findings of this study are available on request from the corresponding author, Yihong Li, upon reasonable request.

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

References

1. Wang, X. A preliminary study on the South China Sea voyages GENG LU BU Peng Zhengkai transcript. *Qilu J.* **2015**, *6*, 42–50.
2. Li, W.; Zhiguo, G.; Huang, L. Research on the place names of Siam Bay in Guihai Nian Gengliubu in the perspective of digital humanities. *Geogr. Res.* **2021**, *40*, 1529–1542.
3. Berragan, C.; Singleton, A.; Calafiore, A.; Morley, J. Transformer based named entity recognition for place name extraction from unstructured text. *Int. J. Geogr. Inf. Sci.* **2023**, *37*, 747–766. [CrossRef]
4. Lenc, L.; Martínek, J.; Baloun, J.; Prantl, M.; Král, P. Historical map toponym extraction for efficient information retrieval. In Proceedings of the Document Analysis Systems: 15th IAPR International Workshop, DAS 2022, La Rochelle, France, 22–25 May 2022; pp. 171–183.
5. Aldana-Bobadilla, E.; Molina-Villegas, A.; Lopez-Arevalo, I.; Reyes-Palacios, S.; Muñiz-Sanchez, V.; Arreola-Trapala, J. Adaptive geoparsing method for toponym recognition and resolution in unstructured text. *Remote Sens.* **2020**, *12*, 3041. [CrossRef]
6. Meimei, X. Policy driven ancient books protection and digital humanities. *Libr. Inf.* **2022**, *2*, 122–126.
7. Aslan, M.F.; Unlersen, M.F.; Sabanci, K.; Durdu, A. CNN-based transfer learning–BiLSTM network: A novel approach for COVID-19 infection detection. *Appl. Soft Comput.* **2021**, *98*, 106912. [CrossRef] [PubMed]
8. Hwang, S.; Hong, J.E.; Nam, Y.K. Towards effective entity extraction of scientific documents using discriminative linguistic features. *KSII Trans. Internet Inf. Syst. (TIIS)* **2019**, *13*, 1639–1658.
9. Wang, J.; Hu, Y.; Joseph, K. NeuroTPR: A neuro-net toponym recognition model for extracting locations from social media messages. *Trans. GIS* **2020**, *24*, 719–735. [CrossRef]
10. Wang, X.; Ma, C.; Zheng, H.; Liu, C.; Xie, P.; Li, L.; Si, L. DM_NLP at semeval-2018 task 12: A pipeline system for toponym resolution. In Proceedings of the 13th International Workshop on Semantic Evaluation, Minneapolis, MN, USA, 6–7 June 2019; pp. 917–923.
11. Schmidhuber, J. Deep learning in neural networks: An overview. *Neural Netw.* **2019**, *61*, 85–117. [CrossRef]
12. Lample, G.; Ballesteros, M.; Subramanian, S.; Kawakami, K.; Dyer, C. Neural architectures for named entity recognition. *arXiv* **2016**, arXiv:1603.01360.
13. Gritta, M.; Pilehvar, M.T.; Collier, N. Which Melbourne? augmenting geocoding with maps. In Proceedings of the Association for Computational Linguistics, Melbourne, Australia, 15–20 July 2018.
14. Kumar, A.; Singh, J.P. Location reference identification from tweets during emergencies: A deep learning approach. *Int. J. Disaster Risk Reduct.* **2019**, *33*, 365–375. [CrossRef]
15. Siami-Namini, S.; Tavakoli, N.; Namin, A.S. The performance of LSTM and BiLSTM in forecasting time series. In Proceedings of the 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 9–12 December 2019; pp. 3285–3292.
16. Chen, T.; Xu, R.; He, Y.; Wang, X. Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Syst. Appl.* **2017**, *72*, 221–230. [CrossRef]
17. Si, S.; Danhao, Z. Chinese Place Name Recognition Based on Deep Learning. *Trans. Beijing Inst. Technol.* **2017**, *37*, 1150–1155.
18. Rhanoui, M.; Mikram, M.; Yousfi, S.; Barzali, S. A CNN-BiLSTM model for document-level sentiment analysis. *Mach. Learn. Knowl. Extr.* **2019**, *1*, 832–847. [CrossRef]
19. Peng, N.; Dredze, M. Improving named entity recognition for chinese social media with word segmentation representation learning. *arXiv* **2016**, arXiv:1603.00786.
20. Lu, W.; Li, J.; Wang, J.; Qin, L. A CNN-BiLSTM-AM method for stock price prediction. *Neural Comput. Appl.* **2021**, *33*, 4741–4753. [CrossRef]
21. Dong, C.; Zhang, J.; Zong, C.; Hattori, M.; Di, H. Character-based LSTM-CRF with radical-level features for Chinese named entity recognition. In *Natural Language Understanding and Intelligent Applications: 5th CCF Conference on Natural Language Processing and Chinese Computing, NLPCC 2016, and 24th International Conference on Computer Processing of Oriental Languages, ICCPOL 2016, Kunming, China, 2–6 December 2016*; Proceedings 24; Springer International Publishing: Berlin/Heidelberg, Germany, 2016; pp. 239–250.
22. Ning, X.; Tian, W.; Li, W.; Lu, Y.; Nie, S.; Sun, L.; Chen, Z. BDARS_CapsNet: Bi-directional attention routing sausage capsule network. *IEEE Access* **2020**, *8*, 59059–59068. [CrossRef]
23. Ma, K.; Tan, Y.J.; Xie, Z.; Qiu, Q.; Chen, S. Chinese toponym recognition with variant neural structures from social media messages based on BERT methods. *J. Geogr. Syst.* **2022**, *24*, 143–169. [CrossRef]
24. Ziniu, W.; Meng, J.; Jianling, G.; Meng, Q. Chinese named entity recognition method based on BERT. *Comput. Sci.* **2019**, *46*, 138–142.
25. Devlin, J.; Chang, M.W.; Lee, K.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.

26. Cui, Y.; Che, W.; Liu, T.; Qin, B.; Wang, S.; Hu, G. Revisiting pre-trained models for Chinese natural language processing. *arXiv* **2020**, arXiv:2004.13922.
27. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. Albert: A lite bert for self-supervised learning of language representations. *arXiv* **2019**, arXiv:1909.11-942.
28. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 1–11.
29. Zhong, L.; Hu, L.; Zhou, H. Deep learning based multi-temporal crop classification. *Remote Sens. Environ.* **2019**, *221*, 430–443. [CrossRef]
30. Greff, K.; Srivastava, R.K.; Koutník, J.; Steunebrink, B.R.; Schmidhuber, J. LSTM: A search space odyssey. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *28*, 2222–2232. [CrossRef] [PubMed]
31. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. *arXiv* **2015**, arXiv:1508.01991.
32. Yan, H.; Deng, B.; Li, X.; Qiu, X. TENER: Adapting transformer encoder for named entity recognition. *arXiv* **2019**, arXiv:1911.04474.
33. Guo, Q.; Qiu, X.; Liu, P.; Shao, Y.; Xue, X.; Zhang, Z. Star-transformer. *arXiv* **2019**, arXiv:1902.09113.
34. Srivastava, N.; Hinton, G.; Krizhevsky, A.; Sutskever, I.; Salakhutdinov, R. Dropout: A simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* **2014**, *15*, 1929–1958.
35. Van der Maaten, L.; Hinton, G. Visualizing data using t-SNE. *J. Mach. Learn. Res.* **2008**, *9*, 2579–2605.