electronics

Article

Named Entity Recognition for Few-Shot Power Dispatch Based on Multi-Task

Zhixiang Tan [1], Yan Chen [1,2,*], Zengfu Liang [1], Qi Meng [3] and Dezhao Lin [1]

[1] School of Computer and Electronic Information, Guangxi University, Nanning 530004, China
[2] Guangxi Intelligent Digital Services Research Center of Engineering Technology, Nanning 530004, China
[3] Guangxi Power Grid Co., Ltd., Nanning 530022, China
* Correspondence: cy@gxu.edu.cn

Abstract: In view of the fact that entity nested and professional terms are difficult to identify in the field of power dispatch, a multi-task-based few-shot named entity recognition model (FSPD-NER) for power dispatch is proposed. The model consists of four modules: feature enhancement, seed, expansion, and implication. Firstly, the masking strategy of the encoder is improved by adopting whole-word masking, using a RoBERTa (Robustly Optimized BERT Pretraining Approach) encoder as the embedding layer to obtain the text feature representation, and an IDCNN (Iterated Dilated CNN) module to enhance the feature. Then the text is cut into one Chinese character and two Chinese characters as a seed set, the score for each seed is calculated, and if the score is greater than the threshold value ω, they are passed to the expansion module as candidate seeds; next, the candidate seeds need to be expanded left and right according to offset γ to obtain the candidate entities; finally, to construct text implication pairs, the input text is used as a premise sentence, the candidate entity is connected with predefined label templates as hypothesis sentences, and the implication pairs are passed to the RoBERTa encoder for the classification task. The focus loss function is used to alleviate label imbalance during training. The experimental results of the model on the power dispatch dataset show that the precision, recall, and F1 scores of the recognition results in 20-shot samples are 63.39%, 61.97%, and 62.67%, respectively, which is a significant performance improvement compared to existing methods.

Keywords: power dispatch; multi-task; few-shot; RoBERTa; IDCNN; named entity recognition

Citation: Tan, Z.; Chen, Y.; Liang, Z.; Meng, Q.; Lin, D. Named Entity Recognition for Few-Shot Power Dispatch Based on Multi-Task. *Electronics* **2023**, *12*, 3476. https://doi.org/10.3390/electronics12163476

Academic Editor: Ahmed Abu-Siada

Received: 10 July 2023
Revised: 13 August 2023
Accepted: 14 August 2023
Published: 17 August 2023

1. Introduction

In recent years, China has attached great importance to the development of the electric power business, and through the implementation of a series of policies and measures, the electric power industry has been innovating in the direction of intelligence [1,2] and sustainable development [3], which has become a strong support for China's economic development. Along with the rapid development of information technology and the comprehensive construction of intelligent power grids, a large amount of power dispatch text data has been accumulated, of which about 80% are unstructured and semi-structured data and only about 20% are structured data. In order to improve the efficiency of staff and assist professionals in making decisions on power dispatch, it is necessary to extract useful information from relevant texts and build a power dispatch knowledge graph [4,5] system. Named entity recognition (NER) is a sub-task of information extraction that is widely used in fields such as intelligent question and answer [6], recommendation systems [7], and knowledge graphs [8], and its main task is to identify meaningful entities from relevant texts and classify them. For example, the task of named entity recognition in the field of power dispatch is to identify the relevant entities in the power dispatch data. Entity recognition is a fundamental part of building knowledge graphs, and improving the recognition of entities in power dispatch texts is of great significance for building power dispatch knowledge graphs.

Named entity recognition tasks have the following main challenges in the field of power dispatch:

(1) Lack of annotated public datasets.
(2) The power dispatch text has strong specialized terms, and it is difficult for the conventional model to identify the entities in it. There is also the problem of nested entities in the text, such as when the entity "移动基站专变" contains two entities ("移动基站专变" and "专变"). Text annotation using sequential annotation requires special processing to recognize nested entities.
(3) Conventional tokenizers do not consider the existence of word boundary ambiguity or separators to represent word boundaries in Chinese. In addition, there is no word splitter for power dispatch text splitting, and there is a certain error in the splitting results when using conventional word splitters.
(4) When the pre-trained language model trains the Chinese corpus, the character masking strategy is adopted, and the semantic information extracted is only at the character level, which cannot fully capture the contextual semantic information in the text. In addition, using the character masking strategy model may predict the content of the masking position in advance, but it cannot effectively infer professional terms.

In order to solve the above problems, a model based on few-shot power dispatch entity recognition is proposed to improve the recognition of power dispatch entities, and the contributions of this paper are as follows:

(1) Pre-processing the unstructured data of power dispatch provided by Guangxi Power Grid, referring to the national standard electrical terminology specification, adopting the span-based approach to standardized text data for entity annotation, and constructing the named entity dataset of power dispatch.
(2) Proposing a multi-task-based few-shot named entity recognition model for power dispatch and training it using a dataset based on span representation so that the model can effectively identify nested entities.
(3) Improving the dynamic character masking strategy used by the RoBERTa encoder by replacing the WordPiece masking strategy with the whole-word masking strategy.
(4) Using the focal loss function [9] (focal loss, FL) to solve the sample size imbalance problem.

Compared with the BERT-CRF, BERT-LSTM-CRF, NNShot, and StructShot models, the experimental results show that the proposed model has better recognition performance.

2. Related Work

In early entity recognition tasks [10], the rule-based approach achieved high accuracy and low recall because it was relatively easy to implement and did not require training. However, the shortcomings of this method require the construction of specialized domain knowledge and a large amount of human resources, as well as the improvement in generalization capability. Compared with the rule-based approach, the core of the statistical model-based approach is to select a suitable training model for a specific research context, omitting many tedious rule designs and spending less time to train a manually annotated corpus to improve training efficiency. At the same time, the statistical model-based approach only needs to retrain the model for the domain-specific training set in the face of different domain-specific rules, so this approach has better robustness. With the rapid development of deep learning, the application of deep learning to named entity recognition tasks is gradually becoming mainstream and achieving better results. Among the typical models are: Li [11] built a name entity recognition model of selectable word vectors based on bidirectional long short-term memory (BiLSTM) and conditional random field [12] (CRF) to identify defective texts of power equipment. Wang [13] proposed a character set entity recognition model based on several features. The model combines character embedding, left-neighbor entropy, and lexicality to represent domain features of power dispatch text, using BiLSTM to predict character sequence labels and CRF to optimize the predicted labels. Zheng [14] proposed the Bi-order-Transformer-CRF model, which trains power billing word vectors and designs neural cosine similarity functions to distinguish similar entities and alleviate the problem of entity boundary ambiguity. Liu [15] studied the more widely used entity recognition model (BERT-CRF) and proposed an active learning strategy based on uncertainty that considers both intermediate and input results and alleviates the model's reliance on large amounts of manually labeled data. Meng [16] fused the pre-trained models BERT [17], BiLSTM, and CRF to construct a faulty entity recognition model for electrical equipment. Attention mechanisms [18] have developed rapidly in the field of computer vision, and researchers have used convolutional neural networks (CNNs) for entity recognition tasks to obtain features of high-dimensional data. Zheng [19] combined the attention mechanism with CNNs to obtain the contextual relationships of the electricity billing text and then used BiGRU to extract high-level features and CRF to obtain the output label sequences. Power dispatch text data present many kinds of entities in a small volume, but there is an insufficient corpus set for deep learning model training and data annotation has a high cost, so the above traditional entity recognition model cannot achieve better results.

Recently, few-shot named entity recognition, which aims to identify entities based on a small number of label instances in each category, has received a lot of attention. Cui [20] proposed the TemplateNER model to enumerate all text spans of a text as candidate entity labels by an exhaustive method and construct the corresponding prompt templates [21] to pass into the Seq-to-Seq model [22] for classification. Ma [23] abandoned the template construction process and retained the word prediction paradigm of pre-trained models to predict similarly related tagged words at entity locations while also exploring principled approaches to automatically search for suitable tagged words, reformulating the NER task as a language modeling problem without templates. Wang [24] proposed a seminal span-based prototypical network (SpanProto) that tackles few-shot NER via a two-stage approach, including span extraction and mention classification, in order to make more efficient use of boundary information. Chen [25] designed Self-describing Networks (SDNets), a Seq2Seq generation model that can universally describe mentions using concepts, automatically map novel entity types to concepts, and adaptively recognize entities on demand. However, the above three models are all general-domain-oriented, the datasets used by the models are all in English, and the datasets used for training are based on sequence labeling methods, which cannot effectively identify nested entities.

In summary, the research on named entity recognition of power dispatch text has not been carried out in depth, and the existing dataset of power dispatch is not sufficient to support the effective learning of traditional deep learning models. Therefore, this paper proposes a FSPD-NER model, which uses a span-based representation to represent entities to solve the problem of nested entities and a few-shot approach to train the model to solve the problem of an insufficient training corpus.

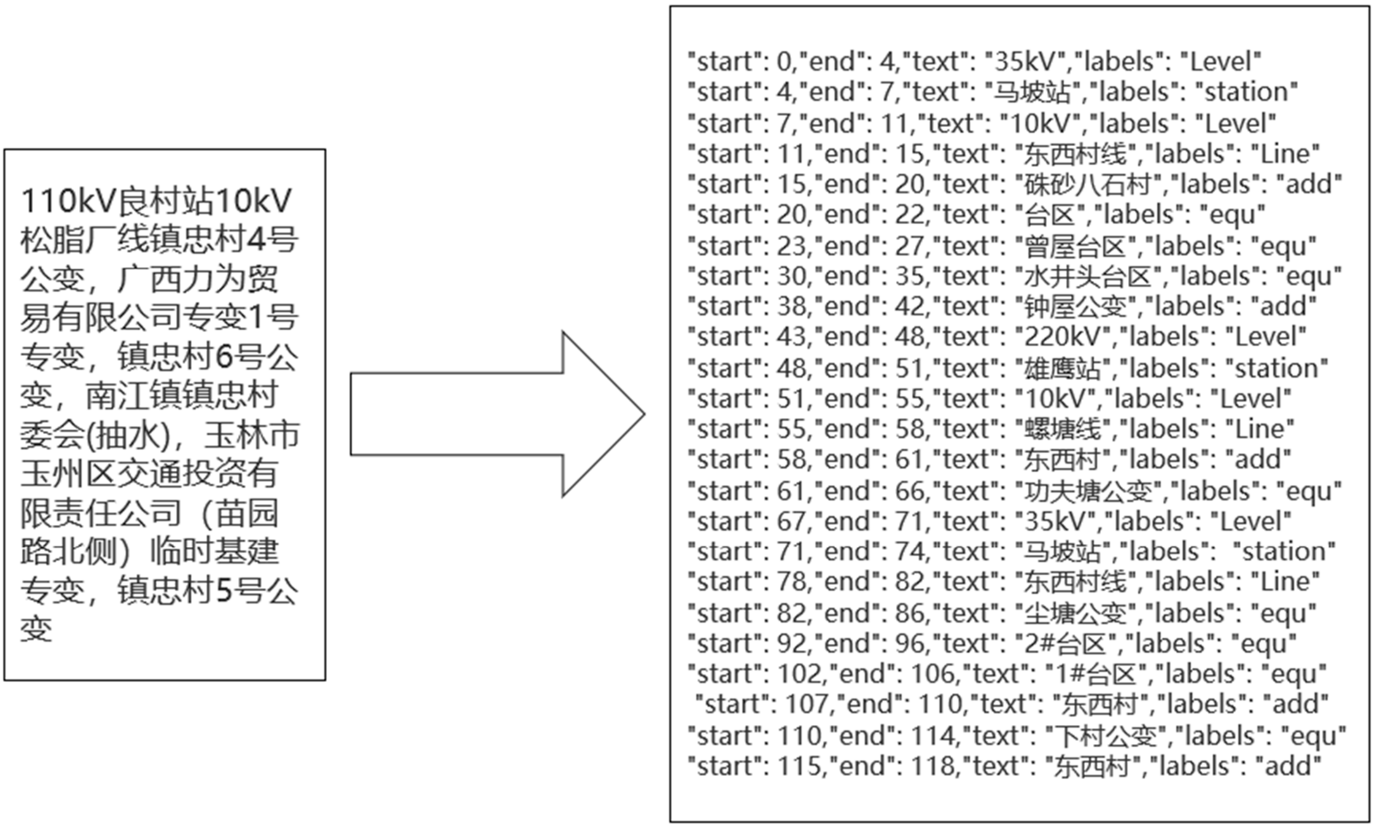
3. Constructing Corpus Datasets

3.1. Data Acquisition and Preprocessing

The structured and unstructured data were extracted from the power dispatch data provided by Guangxi Power Grid. Before labeling the data, all data were converted to unstructured data, organized, and denoised, and finally, 828 pieces of data were extracted.

3.2. Entity Annotation

Referring to the national electric power industry terminology specification dictionary and Guangxi power grid business needs, nine kinds of entity labels were designed: time, voltage level, transmission line, station, organization, equipment, person name, address, and other. Entity annotation was performed by means of the label-studio tool, which labels the start position, end position, and entity type of the entity in the sentence. A labeling example is shown in Figure 1.



110kV良村站10kV松脂厂线镇忠村4号公变，广西力为贸易有限公司专变1号专变，镇忠村6号公变，南江镇镇忠村委会(抽水)，玉林市玉州区交通投资有限责任公司（苗园路北侧）临时基建专变，镇忠村5号公变

```
"start": 0,"end": 4,"text": "35kV","labels": "Level"
"start": 4,"end": 7,"text": "马坡站","labels": "station"
"start": 7,"end": 11,"text": "10kV","labels": "Level"
"start": 11,"end": 15,"text": "东西村线","labels": "Line"
"start": 15,"end": 20,"text": "硃砂八石村","labels": "add"
"start": 20,"end": 22,"text": "台区","labels": "equ"
"start": 23,"end": 27,"text": "曾屋台区","labels": "equ"
"start": 30,"end": 35,"text": "水井头台区","labels": "equ"
"start": 38,"end": 42,"text": "钟屋公变","labels": "add"
"start": 43,"end": 48,"text": "220kV","labels": "Level"
"start": 48,"end": 51,"text": "雄鹰站","labels": "station"
"start": 51,"end": 55,"text": "10kV","labels": "Level"
"start": 55,"end": 58,"text": "螺塘线","labels": "Line"
"start": 58,"end": 61,"text": "东西村","labels": "add"
"start": 61,"end": 66,"text": "功夫塘公变","labels": "equ"
"start": 67,"end": 71,"text": "35kV","labels": "Level"
"start": 71,"end": 74,"text": "马坡站","labels":  "station"
"start": 78,"end": 82,"text": "东西村线","labels": "Line"
"start": 82,"end": 86,"text": "尘塘公变","labels": "equ"
"start": 92,"end": 96,"text": "2#台区","labels": "equ"
"start": 102,"end": 106,"text": "1#台区","labels": "equ"
 "start": 107,"end": 110,"text": "东西村","labels": "add"
"start": 110,"end": 114,"text": "下村公变","labels": "equ"
"start": 115,"end": 118,"text": "东西村","labels": "add"
```

Figure 1. Example of anticipatory annotation.

The corpus contains a total of 7240 entities, including 261 time entities, 1017 voltage level entities, 1328 transmission line entities, 609 station entities, 678 organization entities, 1819 equipment entities, 336 person name entities, 886 address entities, and 306 other entities. The statistics of the number of entities in the corpus, train set, evaluation (eval) set, and test set are shown in Table 1, and the distribution of entities is shown in Figure 2.

Table 1. Statistical information on the number of corpus entities.

Entity Type/ Acronym	Corpus	Train Set			Eval Set			Test Set
	All	5-Shot	10-Shot	20-Shot	5-Shot	10-Shot	20-Shot	All
time/time	261	5	13	15	5	17	15	22
voltage level/time	1017	6	9	24	9	16	23	379
transmission line/line	1328	6	16	32	8	15	26	462
station/station	609	7	12	18	6	14	22	205
organization/org	678	5	10	23	7	11	20	238
equipment/equ	1819	6	10	21	9	25	27	743
person name/name	336	9	11	18	3	8	23	17
address/add	886	6	12	18	5	9	25	311
other/other	306	4	8	18	3	11	11	44

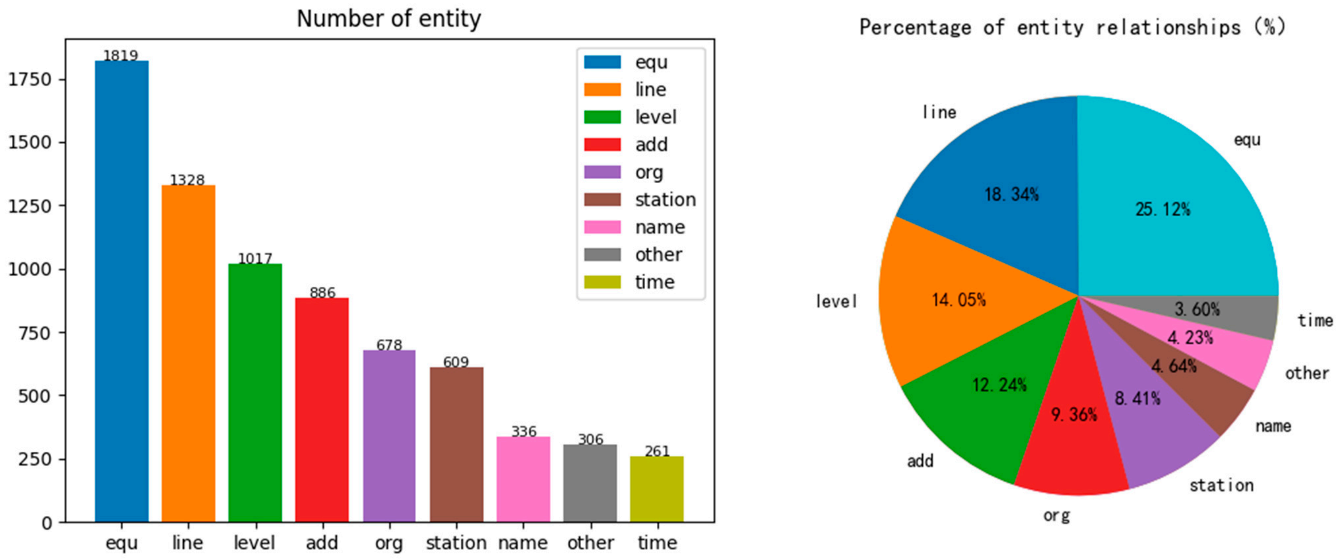


Figure 2. Entities Distribution Chart.

4. Methods

4.1. Method Flow

In view of the lack of public datasets in the field of power dispatch, the lack of a large amount of corpora for training, and the existence of many professional terms and nested entities in the dataset, the traditional model is poor at entity recognition. Therefore, a multi-task-based named few-shot entity recognition method in the field of power dispatch is proposed. The method is mainly divided into five parts: power dispatch corpus construction, construction of seed tags, tag expansion, construction of textual implication pairs, and tag classification. The flow chart of the method is shown in Figure 3, and the model architecture is shown in Figure 4.

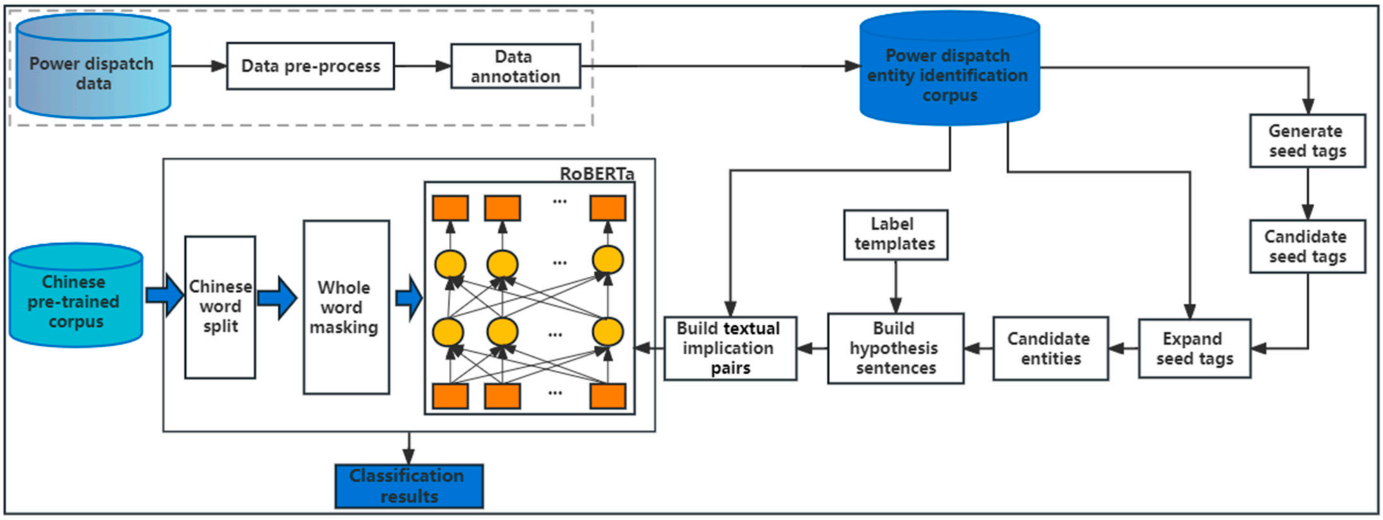


Figure 3. Power dispatch data named entity recognition flow chart.

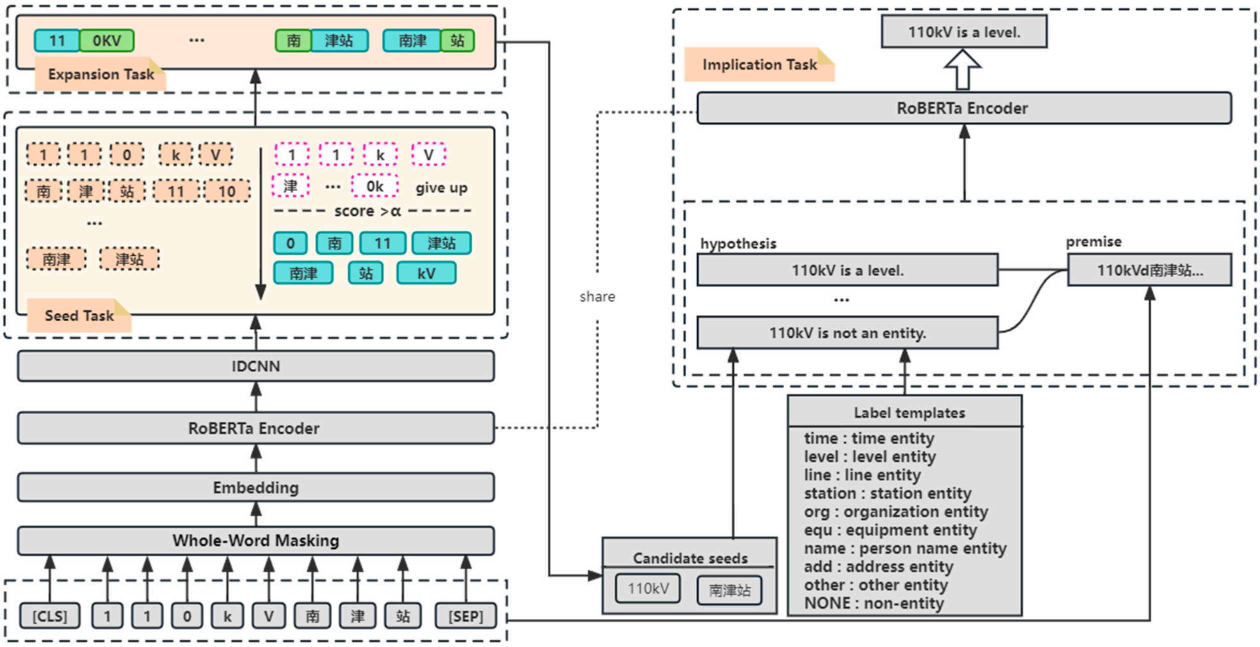


Figure 4. Model Architecture.

Firstly, the power dispatch dataset is input into the RoBERTa [26] encoder to obtain a deep bidirectional representation, and the IDCNN module is used to obtain longer-distance contextual and entity boundary information. The parallelism of the GPU is fully utilized to accelerate the training speed. Second, the text is sliced into one character and two characters as seed entity sets, e.g., “1”, “1”, “0”, …, “南津”, “津站”, and a candidate seed for each seed score that is greater than the threshold ω is calculated. The candidate seeds are expanded to the left or to the right to obtain the candidate entities. Finally, the candidate entities are spliced with the predefined label templates as the hypothesis of the implication task, and the input text is used as the premise of the implication task to obtain a textual implication pair. The textual implication pairs are input into RoBERTa for training and classification.

4.2. RoBERTa and Whole-Word Masking

RoBERTa is an extension and improvement of BERT. The model builds on BERT by replacing the static masking strategy with a dynamic masking strategy (Masked Language Modeling, MLM) that randomly masks each training to improve the adaptability of the model. Meanwhile, RoBERTa uses larger datasets, larger batch sizes, longer pre-trained times, longer input sequences, and more data augmentation techniques to improve the model’s capabilities. Liu [26] found during their iterative experiments on the RoBERTa model that adding the Next Sentence Prediction (NSP) task did not improve the performance of the model, so the NSP task was removed. After the above improvements, RoBERTa obtained a more stable and robust performance than BERT.

For text sequences, after encoding by the encoder, the output Embedding of each word in the set consists of three parts: Token Embedding, Segment Embedding, and Position Embedding, as shown in Figure 5. The sequence vector is input into the bidirectional transformer for feature extraction, and finally, the sequence vector containing rich semantic features is obtained.

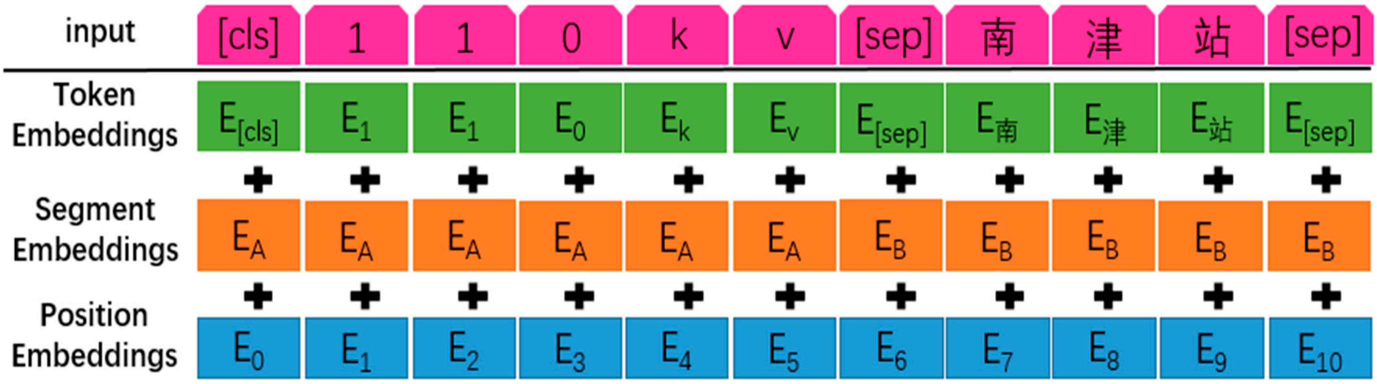


Figure 5. RoBERTa generates input vectors.

The RoBERTa model uses the dynamic masking strategy to train the text and extract the bidirectional representation of depth. This strategy dynamically and randomly masks part of the tokens, allowing the pre-trained model to infer the contents of the masked position, and the position of the MASK of the model is different in each round. For example, in the sentence "10 kV怀集线", the masked sentence in the first round of training is "10 kV怀集[MASK]", the second round is "10 kV[MASK]集线", and so on. However, in order to improve the reasoning ability of the power dispatch dataset, the character masking strategy is changed to the whole-word masking strategy.

Unlike the whole-word masking proposed by Cui [27], this paper incorporates proper nouns from the grid dictionary before using the word splitter to make the model splitting results more accurate; then multiple consecutive [MASK] marks are used to mask the complete words, and the model predicts the masked words, which makes the model obtain more feature information and alleviates the semantic incompleteness when making Chinese predictions due to bias, as shown in Figure 6. An example of the improved masking of text is shown in Table 2.

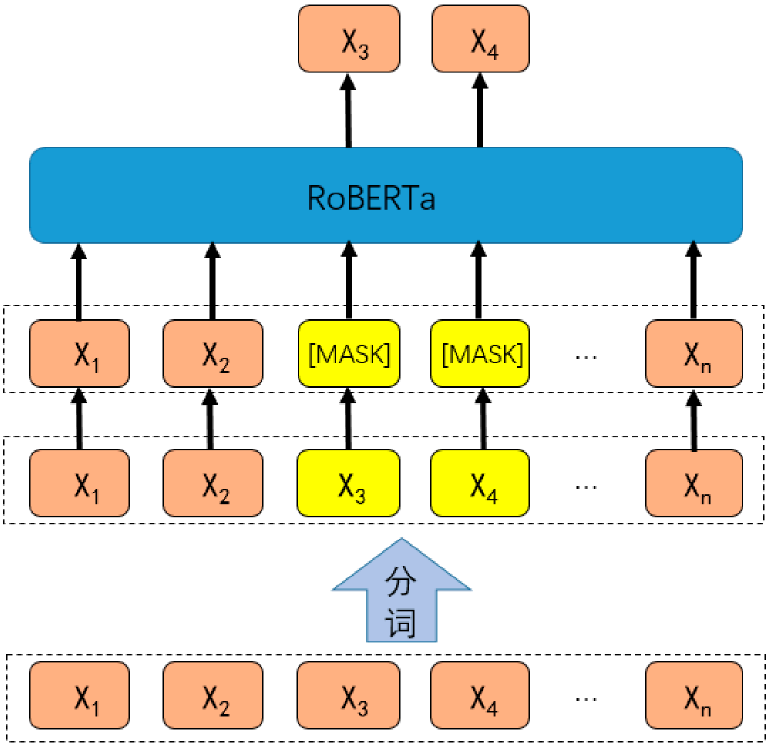


Figure 6. Whole-word masking strategy. X_1~X_n denote the characters in the input sequence, and [MASK] denotes the masked part.

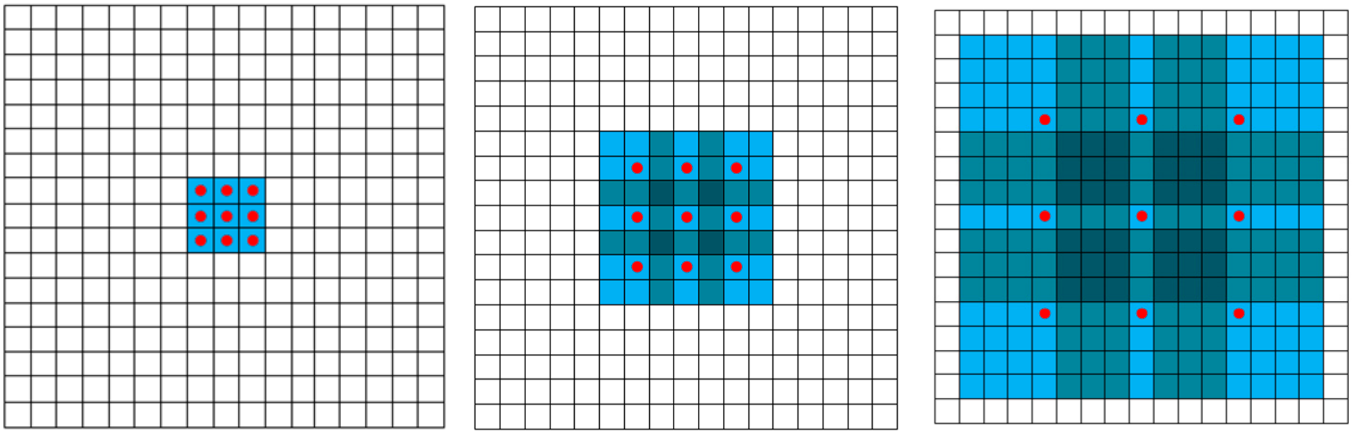
Table 2. Example of whole-word masking.

Masking Strategy	Example
Original text	拉开10 kV鹿山线鸿森胶合板厂
WordPiece Masking	拉开10 kV鹿山[MASK]鸿森胶合板厂
Whole-Word Masking	拉开10 kV[MASK][MASK][MASK]鸿森胶合板厂

4.3. IDCNN Module

When convolutional neural networks (CNNs) are applied to process text, after the convolution operation, the last layer of the network may only acquire a small portion of the information from the original input data. In order to obtain more contextual information, more convolutional layers are added, and the network becomes complex and prone to overfitting. To solve this problem, the Dropout mechanism is introduced, but it also introduces more hyper-parameters, making the model large, redundant, and difficult to train.

To solve the above problem, Yu [28] proposed Dilation Convolution, which expands the perceptual field of a convolutional neural network and reduces information loss by increasing the spacing of convolutional kernels. Dilation Convolution increases the expansion width on top of the original convolutional neural network. During the convolution operation, the size of the convolution kernel does not change, and the convolution kernel skips the data in the middle of the expansion width. In this way, a convolutional kernel of the same size can acquire wider input matrix data, thus increasing the perceptual field of view of the convolutional kernel. The schematic diagram of the inflated convolution is shown in Figure 7. Figure 7a indicates the normal convolution operation with a convolution kernel size of 3 × 3; Figure 7b indicates the convolution with an expansion width of 2, and the perceptual field of view increases to 7 × 7, skipping the center adjacent nodes and causing two voids to appear, which capture the nodes adjacent to the center directly; Figure 7c shows an expansion width of 4, and the perceptual field of view expands to 15 × 15, causing six voids to appear, which capture a larger range of node information. The expansion convolution maximizes the effectiveness and accuracy of the model.



(**a**) 3 × 3 field of view (**b**) 7 × 7 field of view (**c**) 15 × 15 field of view

Figure 7. IDCNN diagram.

The IDCNN-specific implementation process is as follows: first of all, the input width L_n of the *n*th layer is calculated as shown in Equation (1); with the increase in the number of layers *n*, the length of the covered sequence increases exponentially, so as to cover a long sequence of text.

$$L_n = 2^{n+1} - 1 \quad (1)$$

With c_i as the input representation of RoBERTa, the output of the next layer is obtained after the inflated convolution of the input of the previous layer. As the number of layers of the inflated convolution increases, the area of the perceptual field of view of the convolution kernel also improves. The calculation process for each layer of convolution is shown in Equation (2), where D represents the nth layer of convolution with input width L_n and σ is the ReLU activation function.

$$c_i^n = \sigma\left(D_{L_n-1}^{n-1} \cdot c_i^{n-1}\right) \tag{2}$$

Multiple inflated convolutional layers are stacked to form an inflated convolutional block, and $B()$ represents the convolutional block, for which k iterations are performed to obtain c_i^k, as shown in Equation (3).

$$c_i^k = B\left(c_i^{k-1}\right) \tag{3}$$

The Dropout mechanism is introduced to improve model generalization by randomly discarding the output of iteratively inflated convolutional blocks at a certain rate. p_i^k is the final model probability output, as shown in Equation (4).

$$p_i^k = Dropout\left(c_i^k\right) \tag{4}$$

4.4. Seed Module

Given an input text X = {x$_1$, x$_2$, x$_3$, …, x$_n$} of n characters, the input text is sliced into a set S = {s_1, s_2, s_3, …, s_{2n-1}} of one character and two characters, where s_i = (l_i, r_i) and l_i and r_i denote the left boundary (start position) and right boundary (end position) of the seed, respectively.

The purpose of the seed task is to find one character or two characters that overlap with the full entity and have the potential to expand into the full entity. The text is passed into the RoBERTa encoder to obtain the representation v. For the seed s_i = (l_i, r_i), the fraction calculation p_i^{seed} procedure is as follows:

$$v_i^p = MeanPooling\left(v_{l_i}, \ldots, v_{r_i}\right) \tag{5}$$

$$v_i^{seed} = Concat\left(v_i^p, v^{[cls]}\right) \tag{6}$$

$$p_i^{seed} = Sigmoid\left(MLP\left(v_i^{seed}\right)\right) \tag{7}$$

where v_i^p represents performing mean pooling on all representations of the seed span interval, fusing it with the output of [CLS] as the representation of the seed v_i^{seed}. Finally, the seed representation is passed into the multilayer perceptron, and the final seed fraction p_i^{seed} is obtained. According to the set seed score threshold ω, a seed score greater than the threshold ω is considered a candidate seed.

4.5. Expansion Module

With the help of the regression task, the candidate seeds are expanded to obtain the candidate entities. The set candidate seed s_i = (l_i, r_i) of the left boundary l_i and the right boundary r_i can be shifted at most γ to obtain the longest entity as 2 + 2γ. The value of the seed entity after shifting to the left is w_i^l, the value after shifting to the right is w_i^r, and the span of the expanded entity is w_i. The calculation process is as follows:

$$w_i^l = \max(1, l_i - 2\gamma) \tag{8}$$

$$w_i^r = \min(n, r_i + 2\gamma) \tag{9}$$

$$w_i = \left(w_i^l,\ w_i^r\right) \tag{10}$$

Next, the entity span of the above expansion is calculated to determine the candidate entities. The left and right boundary offsets o_i are calculated as follows:

$$v_i^w = MeanPooling\left(v_{w_i^l}, \ldots, v_{w_i^r}\right) \tag{11}$$

$$v_i^{expand} = Concat\left(v_i^p, v_i^w\right) \tag{12}$$

$$o_i = \gamma \cdot \left(2 \cdot Sigmoid\left(MLP\left(v_i^{expand}\right)\right) - 1\right) \tag{13}$$

where $o_i \in \mathrm{R}^2$; its first element is denoted as o_i^l, which represents the final left offset of the entity span; and the second element is denoted as o_i^r, which represents the final right offset of the entity span, $i \in [-\gamma,\ \gamma]$. The left and right boundaries of the final entity can be determined, and the calculation process is as follows:

$$l_i' = \max\left(1, l_i + \left\lfloor o_i^l + \frac{1}{2} \right\rfloor\right) \tag{14}$$

$$r_i' = \min\left(n, r_i + \left\lfloor o_i^r + \frac{1}{2} \right\rfloor\right) \tag{15}$$

During the calculation of the result, if $o_i^l \ > \ o_i^r$ is considered invalid, it is chosen to be discarded.

4.6. Implication Module

To construct the required implication pairs for the implication task, candidate entities and label templates are spliced as hypothesis sentences, and input text is used as premise sentences to form complete implication pairs. For the *i*th candidate entity span c_i, the implication pair is of the form (X, P_i^j), where P_i^j = {c_i is a/an e_i}, $e_i \in E$, E is the set of label templates, and X is the input text.

The implication pairs are fed to the shared RoBERTa encoder for training, and the $v_{P_i^j}^{[CLS]}$ of the encoder output is obtained. The result of performing text classification is:

$$p_{i,j}^{entailment} = Softmax\left(MLP\left(v_{P_i^j}^{[CLS]}\right)\right) \tag{16}$$

where the implication label can be defined as follows:

$$y_{i,j}^{entaiment} = \begin{cases} true \ \ if \ P_i^j \ is \ valid \\ false \ \ else \end{cases} \tag{17}$$

4.7. Focal Loss

When building the Guangxi power dispatch data corpus, it is found that the number of various types of entities is unevenly distributed according to the statistical results, which leads to an uneven distribution of loss functions during model training and prefers labeled entities with a larger number of samples, making the label recognition of entities with a smaller number of samples less effective. To alleviate this problem, this paper uses the focal loss function to reduce the weight of the samples that are easy to classify and increase the weight of the samples that are difficult to classify. The focal loss function improves the weight factor in the cross-entropy loss function by introducing a focal factor α, which is used to adjust the weights of easy-to-classify samples and hard-to-classify samples. When the focus factor is larger, the model will pay more attention to the hard-to-classify samples, thus improving the classification precision of the model, which is calculated as in Equation (18).

$$\mathrm{FL}(p_t) = -\alpha_t \cdot (1 - p_t)^{\tau} \cdot \log(p_t) \tag{18}$$

where p_t represents the probability that the label is correctly recognized; the larger the p_t, the higher the confidence of recognition, and the easier the label is recognized. $(1 - p_t)^{\tau}$ is the modulation factor, $\tau \geq 0$. The focus factor $\alpha \in [0, 1]$ is used to adjust the weights of the variation samples and balance the distribution of the loss function.

5. Results

The experimental environment uses the Pytorch framework, CUDA version 11.1, an Ubuntu system, and an NVIDIA RTX3090 (24G) graphics card. The threshold ω size of the seed score is 0.6, and the entity offset γ is set to 5. The AdamW optimization algorithm is used, and the learning rate (lr) is 3×10^{-5}. The Dropout mechanism is introduced, and the value is set to 0.5. The batch_size parameter has a value of 4, and the number of iterations (epochs) is 30. One evaluation is performed at the end of each iteration, and the model with the highest F1 score is saved. The dataset is constructed using the named entity recognition corpus for power dispatch included in Section 3 of this paper, and the train and eval sets are set up with three different sample sizes for the comparison experiments (5-shot, 10-shot, and 20-shot sample sizes). The inference phase is validated using the test set, which contains 287 pieces of data. Table 3 depicts other parameter settings: sampling_processes represents the number of threads that process data, with the size of the parameter set depending on the individual's computer; eval_batch_size specifies the batch size of data in the verification phase; and lr_warmup is the ratio of total training iterations to warm-up in a linear increase/decrease program, used for gradient clipping to prevent gradient explosion. The setting of the values of the above parameters is determined by repeated experiments.

Table 3. Parameter Settings.

Index	Parameter	Value	Index	Parameter	Value
1	Hidden size	768	5	lr_warmup	0.1
2	Embedding size	128	6	random_seed	42
3	sampling_processes	4	7	α	0.25
4	eval_batch_size	8	8	τ	2

5.1. Evaluation Indicators

The precision (P), recall (R), and $F1$ scores are used as evaluation indices of model performance, and the specific formula calculation process is as follows:

$$P = \frac{TruePositive}{PredictPositive} \times 100\% \tag{19}$$

$$R = \frac{TruePositive}{ActualPositive} \times 100\% \tag{20}$$

$$F1 = \frac{2PR}{P + R} \times 100\% \tag{21}$$

In the above equation, *TruePositive* denotes the number of entities correctly predicted by the model, that is, the number of samples that the model predicts as positive cases and are actually positive cases; *PredictPositive* denotes the total number of entities identified by the model, including correctly predicted entities and incorrectly predicted entities; *Actual-*

Positive denotes the total number of entities present in the dataset, that is, the number of real entities.

5.2. Results and Analysis

Three groups of comparison experiments are set up to evaluate the effectiveness of the model. The first group evaluates the performance analysis of different masking strategies; the second group includes the experiments comparing the model of this paper with other models; and the third group includes the ablation experiments on the model.

(1) Performance analysis of different masking strategies

In order to verify that the whole-word masking strategy can effectively improve the recognition effect of the model, the FSPD-NER model using WordPiece masking and whole-word masking is compared and experimented with, and the results are shown in Table 4.

Table 4. Performance comparison of different masking strategies (%).

ID	Whole-Word Masking	BERT	RoBERTa	Precision			Recall			F1 Score		
				5-Shot	10-Shot	20-Shot	5-Shot	10-Shot	20-Shot	5-Shot	10-Shot	20-Shot
1	-	√	-	58.70	58.05	60.94	41.36	49.04	53.81	48.52	53.17	57.15
2	-	-	√	59.32	60.36	62.28	42.56	47.08	56.71	49.56	52.89	59.37
3	√	√	-	60.87	60.94	62.88	44.08	50.19	58.93	51.13	55.05	60.84
4	√	-	√	61.79	61.75	63.39	45.23	50.48	61.97	52.23	55.94	62.67

From the above table, it can be seen that when the FSPD-NER model adopts the WordPiece masking strategy, compared with the BERT pre-trained model, the RoBERTa pre-trained model improves the precision, recall, and F1 scores by 1.34%, 2.90%, and 2.22%, respectively, under 20 samples. This is because the RoBERTa model uses a larger corpus for training to capture the contextual relevance of text, and the model recognition effect is improved. The FSPD-NER model adopts the RoBERTa pre-trained model; compared with the use of the WordPiece masking strategy, the precision, recall, and F1 scores of the whole-word masking strategy increase by 1.11%, 5.26%, and 3.30%, respectively, under 20 samples, which is a comparison of IDs 2 and 4. To prove that using whole-word masking can effectively improve the performance of the model, the FSPD-NER model in the following text uniformly uses the RoBERTa pre-trained model and the whole-word masking strategy.

(2) Comparative analysis of the performance of different models

In order to verify the effectiveness of the model proposed in this paper for named entity recognition in few-shot power dispatch data, four different models were experimentally set up for comparison. The results show that the FSPD-NER model proposed in this paper exhibits high recognition precision in the few-shot case. The specific comparison results are shown in Table 5.

Table 5. Comparison results of different models (%).

ID	Model	Precision			Recall			F1 Score		
		5-Shot	10-Shot	20-Shot	5-Shot	10-Shot	20-Shot	5-Shot	10-Shot	20-Shot
1	BERT-CRF	17.62	24.19	27.79	23.52	23.70	43.45	20.14	23.94	33.90
2	BERT-LSTM-CRF	29.30	30.58	39.21	10.92	19.88	44.54	15.91	24.10	41.70
3	NNShot	17.63	19.20	22.45	26.74	36.76	39.49	21.24	25.23	23.69
4	StructShot	29.97	32.69	34.68	24.09	29.85	30.75	26.71	31.21	32.60
5	FSPD-NER	61.79	61.75	63.39	45.23	50.48	61.97	52.23	55.94	62.67

Table 5 shows the experimental results of different models on the power dispatch dataset. BERT-CRF utilizes the BERT pre-trained model as the embedding layer to fully consider the location information and contextual semantic information of characters and learns the annotation constraint rules and adjacent label information from CRF to obtain a globally optimal annotation sequence to alleviate the annotation bias problem. The F1 scores of the model are 20.14%, 23.94%, and 33.90% under the conditions of 5-shot, 10-shot, and 20-shot samples, respectively. BERT-LSTM-CRF is based on BERT-CRF and incorporates LSTM to consider time series information to alleviate the information-forgetting problem. However, due to the small number of training samples and the fact that the training corpus in the pre-trained model is relatively extensive, it results in the model learning more irrelevant information and introducing more noise during the training process, which makes the recognition of the model less effective, with F1 scores of 15.91%, 24.10%, and 41.70% under the conditions of 5-shot, 10-shot, and 20-shot sample sizes, respectively. The experimental results show that the traditional named entity recognition model does not perform well in corpora with a small sample size. The NNShot [29] model mainly obtains the contextual vector representation of each word in the sentence, calculates the word similarity in the vector space using the nearest-neighbor principle, and selects the nearest word category for labeling. The backbone of the StructShot [29] model is based on NNShot, and the Viterbi algorithm is used to decode the prediction. These two models use corpus sets from other domains to train the models and only use the power dispatch dataset for validation in the prediction phase. The models incorporate more noise, and the obtained semantic information contains more error information. In addition, the corpus of the models is labeled with sequences, which cannot alleviate the entity nested problem and makes the recognition performance of the models poor. The F1 scores of NNShot were 21.24%, 25.23%, and 23.69% for the conditions of 5-shot, 10-shot, and 20-shot sample sizes, respectively; the F1 scores of StructShot were 26.71%, 31.21%, and 32.60%.

The FSPD-NER model includes data enhancement, seed, expand, and entailment modules. This method reconstructs the named entity recognition task into the classification task of the language model and uses the power dispatch dataset for training to reduce the introduction of noise. In addition, using the whole-word masking strategy, the model can learn the contextual semantic information of the whole word and improve its reasoning ability. At the same time, using the span method to label the corpus can solve the problem that the above model cannot make full use of the boundary information. Under 5-shot, 10-shot, and 20-shot sample sizes, the recognition performance was significantly improved compared with other models, with F1 values of 52.23%, 55.94%, and 62.67%, respectively.

(3) Ablation experiments

To verify the effectiveness of the different modules of the FSPD-NER model proposed in this paper, ablation experiments were performed on the model in the 20-shot sample. Table 6 shows the results of the precision, recall, and F1 scores of the model.

Table 6. Results of ablation experiments (%).

ID	Whole-Word Masking	IDCNN	Loss Function	Precision	Recall	F1 Score
1	√	√	FL	63.39	61.97	62.67
2	-	√	FL	62.28	56.71	59.37
3	√	-	FL	61.64	60.89	61.26
4	√	√	CE	62.98	60.21	61.56

The results of the ablation experiments show that the FSPD-NER model adopts the WordPiece masking strategy, and the F1 score is reduced by 3.30%, which proves that the whole-word masking strategy can effectively improve the learning ability of the model and facilitate the model's ability to extract more semantic information. The FSPD-NER model removes the IDCNN module, and the F1 score decreases by 1.41%, indicating that fusing IDCNN modules can extract more contextual semantic information. The FSPD-NER model replaces the focal loss function with the cross-entropy loss function, and the F1 score is reduced by 1.11%, indicating that the focal loss function can alleviate the sample imbalance problem. In short, each part of the model is indispensable.

6. Conclusions

In view of the lack of labeled public datasets, nested entities, and many professional terms in the data of power dispatch, this paper proposes a few-shot power dispatch named entity recognition (FSPD-NER) model based on multi-tasks. This method uses the RoBERTa pre-trained model as the embedding layer, fully considering the position information and contextual semantic information of characters, and improves the masking strategy of the pre-trained model based on the characteristics of Chinese semantics to enhance the inference ability of the model and further extract the semantic information of Chinese text using the IDCNN module, allowing the model to obtain deeper semantic information. The seed module divides the text into one Chinese character and two Chinese characters and selects seeds that overlap with entities as candidate seeds; the expand stage extends the candidate seeds to the left and right directions to obtain the candidate entities. The implication module mainly constructs implication pairs to achieve text classification. The focus loss function is used in model training to alleviate the sample imbalance problem. The unstructured data provided by Guangxi Power Grid are labeled based on the spanwise approach, and the model is experimentally analyzed on this dataset. The experimental results show that the model achieves better performance compared with the benchmark model, with precision, recall, and F1 scores of 63.39%, 61.97%, and 62.67% in the 20-shot sample experiment.

In future work, we will focus on exploring the feature fusion method applicable to the field of power dispatching so as to improve the effect of named entity recognition. In addition, the correlation between entities and entity relationship extraction will be studied to better mine the available information in the field of power dispatching. Eventually, we plan to apply our model to other domains to verify its generalization ability.

Author Contributions: Conceptualization, Y.C.; Methodology, Z.T.; Software, Z.T., Y.C. and Z.L.; Validation, Y.C., Z.L. and D.L.; Formal analysis, Z.L. and D.L.; Resources, Q.M.; Data curation, Z.T., Z.L., Q.M. and D.L.; Writing–original draft, Z.T. and Y.C.; Writing–review & editing, Z.T. and Y.C.; Visualization, Q.M.; Supervision, Q.M.; Project administration, Z.T. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Guangxi Scientific Research and Technology Development Plan Project grant number AA20302002-3 and Innovation Project of China Southern Power Grid Co., Ltd. (202201161).

Data Availability Statement: Since the data uses the real power data of Guangxi Power Grid Co., it is too sensitive and inconvenient to disclose.

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

References

1. Dileep, G. A survey on smart grid technologies and applications. *Renew. Energy* **2020**, *146*, 2589–2625. [CrossRef]
2. Capizzi, G.; Sciuto, G.L.; Napoli, C.; Tramontana, E. Advanced and Adaptive Dispatch for Smart Grids by means of Predictive Models. *IEEE Trans. Smart Grid* **2017**, *9*, 6684–6691. [CrossRef]
3. Hu, J.; Li, P. Energy-sharing method of smart buildings with distributed photovoltaic systems in area. *Energy Rep.* **2022**, *8*, 622–627. [CrossRef]
4. Fan, S.; Liu, X.; Chen, Y.; Liao, Z.; Zhao, Y.; Luo, H.; Fan, H. How to construct a power knowledge graph with dispatching data? *Sci. Program.* **2020**, *2020*, 8842463. [CrossRef]
5. Wang, J.; Wang, X.; Ma, C.; Kou, L. A survey on the development status and application prospects of knowledge graph in smart grids. *IET Gener. Transm. Distrib.* **2021**, *15*, 383–407. [CrossRef]
6. Lin, T.H.; Huang, Y.H.; Putranto, A. Intelligent question and answer system for building information modeling and artificial intelligence of things based on the bidirectional encoder representations from transformers model. *Autom. Constr.* **2022**, *142*, 104483. [CrossRef]
7. Guo, Q.; Zhuang, F.; Qin, C.; Zhu, H.; Xie, X.; Xiong, H.; He, Q. A survey on knowledge graph-based recommender systems. *IEEE Trans. Knowl. Data Eng.* **2020**, *34*, 3549–3568. [CrossRef]
8. Chen, X.; Jia, S.; Xiang, Y. A review: Knowledge reasoning over knowledge graph. *Expert Syst. Appl.* **2020**, *141*, 112948. [CrossRef]
9. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.

10. Goyal, A.; Gupta, V.; Kumar, M. Recent named entity recognition and classification techniques: A systematic review. *Comput. Sci. Rev.* **2018**, *29*, 21–43. [CrossRef]
11. Li, J.; Fang, S.; Ren, Y.; Li, K.; Sun, M. SWVBiL-CRF: Selectable Word Vectors-based BiLSTM-CRF Power Defect Text Named Entity Recognition. In Proceedings of the 2020 IEEE International Conference on Big Data (Big Data), Atlanta, GA, USA, 10–13 December 2020; pp. 2502–2507.
12. Lafferty, J.D.; McCallum, A.; Pereira, F.C.N. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In Proceedings of the Eighteenth International Conference on Machine Learning, San Francisco, CA, USA, 28 June–1 July 2001.
13. Wang, M.; Zhou, T.; Wang, H.; Zhai, Y.; Dong, X. Chinese power dispatching text entity recognition based on a double-layer BiLSTM and multi-feature fusion. *Energy Rep.* **2022**, *8*, 980–987. [CrossRef]
14. Zheng, K.; Yang, J.; Zeng, L.; Gong, Q.; Li, S.; Zhou, S. Constructing Bi-order-Transformer-CRF with Neural Cosine Similarity Function for power metering entity recognition. *IEEE Access* **2021**, *9*, 133491–133499. [CrossRef]
15. Liu, M.; Tu, Z.; Wang, Z.; Xu, X. LTP: A new active learning strategy for BERT-CRF based named entity recognition. *arXiv* **2020**, arXiv:2001.02524. [CrossRef]
16. Meng, F.; Yang, S.; Wang, J.; Xia, L.; Liu, H. Creating knowledge graph of electric power equipment faults based on BERT–BiLSTM–CRF model. *J. Electr. Eng. Technol.* **2022**, *17*, 2507–2516. [CrossRef]
17. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. *Adv. Neural Inf. Process. Syst.* **2017**, *30*, 5998–6008.
19. Zheng, K.; Sun, L.; Wang, X.; Zhou, S.; Li, H.; Li, S.; Zeng, L.; Gong, Q. Named entity recognition in electric power metering domain based on attention mechanism. *IEEE Access* **2021**, *9*, 152564–152573. [CrossRef]
20. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-based named entity recognition using BART. *arXiv* **2021**, arXiv:2106.01760.
21. Yao, Y.; Zhang, A.; Zhang, Z.; Liu, Z.; Chua, T.S.; Sun, M. Cpt: Colorful prompt tuning for pre-trained vision-language models. *arXiv* **2021**, arXiv:2109.11797.
22. Sutskever, I.; Vinyals, O.; Le, Q.V. Sequence to sequence learning with neural networks. *Adv. Neural Inf. Process. Syst.* **2014**, *27*, 3104–3112.
23. Ma, R.; Zhou, X.; Gui, T.; Tan, Y.; Li, L.; Zhang, Q.; Huang, X. Template-free prompt tuning for few-shot NER. *arXiv* **2021**, arXiv:2109.13532.
24. Wang, J.; Wang, C.; Tan, C.; Qiu, M.; Huang, S.; Huang, J.; Gao, M. Spanproto: A two-stage span-based prototypical network for few-shot named entity recognition. *arXiv* **2022**, arXiv:2210.09049.
25. Chen, J.; Liu, Q.; Lin, H.; Han, X.; Sun, L. Few-shot named entity recognition with self-describing networks. *arXiv* **2022**, arXiv:2203.12252.
26. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
27. Cui, Y.; Che, W.; Liu, T.; Qin, B.; Yang, Z. Pre-training with whole word masking for Chinese bert. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2021**, *29*, 3504–3514. [CrossRef]
28. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
29. Yang, Y.; Katiyar, A. Simple and effective few-shot named entity recognition with structured nearest neighbor learning. *arXiv* **2020**, arXiv:2010.02405.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.