applied sciences

Article

A Combined Semantic Dependency and Lexical Embedding RoBERTa Model for Grid Field Relational Extraction

Qi Meng [1], Xixiang Zhang [1,*], Yun Dong [1], Yan Chen [2,3] and Dezhao Lin [2]

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

Abstract: Relationship extraction is a crucial step in the construction of a knowledge graph. In this research, the grid field entity relationship extraction was performed via a labeling approach that used span representation. The subject entity and object entity were used as training instances to bolster the linkage between them. The embedding layer of the RoBERTa pre-training model included word embedding, position embedding, and paragraph embedding information. In addition, semantic dependency was introduced to establish an effective linkage between different entities. To facilitate the effective linkage, an additional lexically labeled embedment was introduced to empower the model to acquire more profound semantic insights. After obtaining the embedding layer, the RoBERTa model was used for multi-task learning of entities and relations. The multi-task information was then fused using the parameter hard sharing mechanism. Finally, after the layer was fully connected, the predicted entity relations were obtained. The approach was tested on a grid field dataset created for this study. The obtained results demonstrated that the proposed model has high performance.

Keywords: grid field; relational extraction; RoBERTa; semantic dependency; lexical embedding

check for updates

Citation: Meng, Q.; Zhang, X.; Dong, Y.; Chen, Y.; Lin, D. A Combined Semantic Dependency and Lexical Embedding RoBERTa Model for Grid Field Relational Extraction. *Appl. Sci.* **2023**, *13*, 11074. https://doi.org/10.3390/app131911074

Academic Editor: Yijun Mo

Received: 30 August 2023
Revised: 4 October 2023
Accepted: 7 October 2023
Published: 8 October 2023

1. Introduction

Relational extraction tasks involve the extraction of relational facts from unstructured or semi-structured data to identify interactions and attributes between entities [1]. The construction of a knowledge graph typically involves the extraction of information from unstructured information, which is also known as entity-relationship extraction. In relationship extraction, the effective characterization of domain knowledge is challenging [2,3]. Due to the intricate and complex nature of the expertise involved in recording relevant data within the power system [4,5], the effective identification of relationships in this specialized domain is challenging.

A fundamental knowledge graph can be established by first conducting named entity recognition on unstructured data within a specialized domain and subsequently extracting relationships from the identified entities. In the deep learning field, named entity recognition and relation extraction are important tasks in natural language processing [6]. The relationships between different entities were extracted using a rule-based approach. Hou [7] proposed a bootstrap labeling rule discovery approach for robust relation extraction. However, this rule-based approach had low accuracy and was unable to characterize the semantics of lexical elements through vectors via deep learning approaches. Ke [8] proposed a RoFormerV-BiLSTM-CRF based fusion model for medical entity recognition, which used a knowledge graph to analyze the relationships between the medical entities identified in single and multiple patient medical records. Guo [9] proposed a framework for the automatic construction of a process knowledge base in the processing domain based on a knowledge graph. He also developed a knowledge extraction framework that employed BERT-BiLSTM-CRF for the automatic retrieval of knowledge from the process text. Wan [10] proposed a span-based multimodal attention network (SMAN) for joint entity and relation extraction, and introduced a completion mechanism to simultaneously extract the context and span position information. Liu [11] proposed a new pipelined relationship extraction framework that utilized an attentional mechanism to fuse contextual semantic representations, which was able to capture entity location information and type information that are challenging to incorporate into joint models.

The extraction of entity relationships through deep learning is mainly categorized into two methods. The first method is the joint extraction model, where entity recognition and relationship extraction are treated as a whole [12,13]. The second method is the pipeline model, in which entity recognition and relationship extraction are considered as distinct tasks that are handled independently [14–16]. The pipeline approach does not require manual feature construction, which makes it more widely used. The entity recognition task focuses on the identification of real words in the text, while the relationship extraction task tends to the modeling of the links between entities; the separate recognition of entities and relationships enables targeted improvement of the two tasks. The joint extraction can consider the two entities and relations, which avoids the negative impact of errors generated through the entity recognition of the pipelined model on the subsequent task of relation extraction. However, recent studies on the pipelined model led to the improvement of the error propagation problem, which resulted in enhanced recognition through the joint extraction model. Zhong [17] sliced the English vocabulary into its roots and utilized span annotation. The enumerated candidate entities were spliced with the sentences as a training example. This approach effectively enhanced the accuracy of the downstream task of relationship extraction. Ye [18] proposed a neighborhood-oriented packing strategy to pack spans with the same starting lexical elements into a training example in order to better distinguish the entity boundaries and extract the relationships through strategic packing. Through leveraging the pipelined model of span representation, state-of-the-art performance can be attained through fine-tuning BERT.

The relationship extraction task associates different entities and recognizes the type of relationship between them, which can be abstractly represented as edges and nodes in the graph theory for the relationships and entities. Semantic dependency directly links dependent arcs of linguistic units through immediate semantic connections and annotates them with relevant semantic relationships. Semantic dependency focuses on the semantic-factual or logical relations between real words and is able to express deeper semantic information [19]. Yin [20] proposed an approach which consisted of incorporating the glyph information of Chinese characters to enhance the model's ability to deeply characterize the text in named entity recognition for power equipment maintenance records. Sun [21] proposed the semantic enhancement of words with multiple meanings and similar glyphs through incorporating pinyin and glyph information. Jeena [22] proposed a typed Tree-LSTM model that embedded sentence meanings into dense vectors using sentence dependency parsing structures and dependency types. Relationship extraction and named entity recognition are similar tasks that belong to the same natural language processing field. Based on the idea of feature fusion and the characteristics of relationship extraction, this paper combined the semantic dependency information and the lexical embedding information with BERT. It aimed to improve the entity association and semantic characterization capacities of the BERT model.

Compared with the English language, the most obvious feature of the Chinese language is the ambiguity of word boundaries and the absence of separators to represent word boundaries [23,24]. In English, there are separators between the words that identify the boundaries with each word having a distinct meaning, which is not the case in Chinese. Therefore, the relational extraction in Chinese text requires its segmentation. However, there is no established lexicon accessible for the segmenter to employ in the power dispatching domain. Employing a general-domain segmenter within the power grid field leads to considerable inaccuracies. Therefore, word-based encoding is used.

The cross-entropy loss function is a common loss function used to measure the gap between the model output and the actual label in classification problems. It is widely used in various models, such as classification models in machine learning and neural network models in deep learning. In the classification tasks, the cross-entropy loss function is employed to assess the dissimilarity between the probability distribution generated via the model's output and the actual distribution of labels. During the training process, the model will continuously adjust the parameters using the gradient descent algorithm to make the overall loss function as small as possible. Through minimizing the cross-entropy loss function, the model can more accurately predict the class labels of each sample in the classification problem, which improves its performance.

The main contributions of this paper are summarized as follows:

(1) Lexical and semantic dependency dictionaries were constructed, and the RoBERTa word embedding layer was embedded by effectively fusing lexical and semantic dependency information. This made the model learn more dependencies allowing for the extraction of the relationships between the different entities, measuring the model loss through the cross-entropy loss function, and optimizing its parameters through back-propagation;

(2) The cascading effect of downstream tasks, which was caused by the word segmentation error of the Chinese words in the specialized field of electric power, was mitigated via word embedding;

(3) Due to the fact that the existing relational extraction dataset in the field of electric power was relatively small, the self-constructed relational extraction dataset in the field of electric power dispatching was used to support the data requirements of deep learning.

The experimental results demonstrated that the proposed model had higher recognition performance compared with conventional models such as the BERT-Cross Entropy, BERT-CRF, and BERT-BILSTM-CRF.

The remainder of this paper is organized as follows. Section 2 presents the dataset construction. Section 3 describes the relationship extraction method for the grid field, combining the semantic dependency and the lexical embedding constructed for this study. Section 4 details the evaluation of the effectiveness of the proposed model through comparative experiments. Finally, Section 5 presents the conclusion.

2. Materials and Methods

A significant volume of unstructured behavioral data are recorded in the Guangxi regional smart grid system. From this data, textual information such as accident investigation details, audit risk statistics, on-site inspection information, and device operation data were selected to build a power corpus. At present, the system's utilization of this data is low, only supporting simple text queries without in-depth analysis. Thus, the embedded behavioral knowledge cannot be fully utilized. In addition, the existing manual mining method is inefficient and expensive. In this study, a deep learning approach was introduced for analytical modeling. An electric power corpus was then leveraged to construct an entity-relationship dataset within the grid domain, which will be used to train deep learning models.

The corpus employed in this paper consisted of a substantial volume of unstructured data. Screening was performed to eliminate sentences that had unclear meanings, structural flaws, and redundant semantics. Finally, 2316 high-quality data points were extracted and considered as the corpus for the training and testing. Taking into consideration the attributes of the corpus, the entity types were organized into nine categories including: plant and station, voltage level, transmission equipment, equipment and appliances, address, time, person's name, other, and organization. The relationship types were divided into five categories: time, located, subordinate, equivalent, and cause (The relational extraction dataset is shown in Table 1). It aimed to extract information from the unstructured data, which facilitated the subsequent construction of a knowledge graph network for applications in specific areas such as fault analysis, maintenance, and equipment life cycle management. The dataset contained 2316 training data, 17,433 entities, 9354 relationships, and more than 140,000 Chinese and English characters. The training set, validation set, and test set were divided into the ratio of 7:2:1.

Table 1. Relational dataset.

Type Name	**Number of Relations**
time	493
reason	414
local	3317
subordinate	2461
same	2669

This study employed a span-based annotation method to mark entities within sentences, using a visual interface provided by the Label-Studio annotation platform. The span annotation involved defining the start position, end position, and entity type of an entity within a sentence.

3. A Relational Extraction Approach for Grid Field Combining Semantic Dependency and Lexical Embedding RoBERTa Models

This paper proposed a relationship extraction method based on spanning representation, by fusing semantic dependency information and lexical information. This method used the RoBERTa pre-training model to obtain more in-depth semantic representation information allowing it to separately learn entity and relation information. In addition, a multi-task parameter hard-sharing mechanism was used to allow the model to take into account the influence of different tasks by simultaneously training multiple tasks. The effects were reflected in the shared parameters until all the tasks converged. Taking the entity and relationship information into account, the deep semantic representation information of RoBERTa was fully utilized, and the relationship was finally predicted through the fully connected layer. The specific process is shown in Figure 1.

Table 2. Example of a labeled diagram. Numbers in Entities and Relations indicate the corresponding index position in Text.

Text	On 30 June, 110 kv Kunlun station Guangkun line
Entities	[0, 2, time], [4, 4, level], [4, 6, station], [7, 8, line]
Relations	[0, 2, 4, 6, time]

The relationships were labeled as quintupled spanning pairs and relations (i.e., s_1, e_1, s_2, e_2 and relation-type). The variables of the set were, in order, the start index position of entity 1 in the sentence, the end index position of entity 1, the start index position of entity 2, the end index position of entity 2, and the type of relationship (the specific labeling is shown in Table 2. This example eliminated the modifier part of the sentence for a better demonstration, keeping only the part that contained entities and relationships). A main entity was selected from the dataset. The rest of the entities were considered as guest entities and formed a training instance with the original sentence to generate span-based training instances, which were transformed into model input vectors through the embedding layer. Each entity in the text was selected once as the main entity and the rest as guest entities so that multiple training instances were generated. This generation process was performed through automatic enumeration in the program. The link between entities was strengthened through categorizing different subjects into different groups for parallel training.

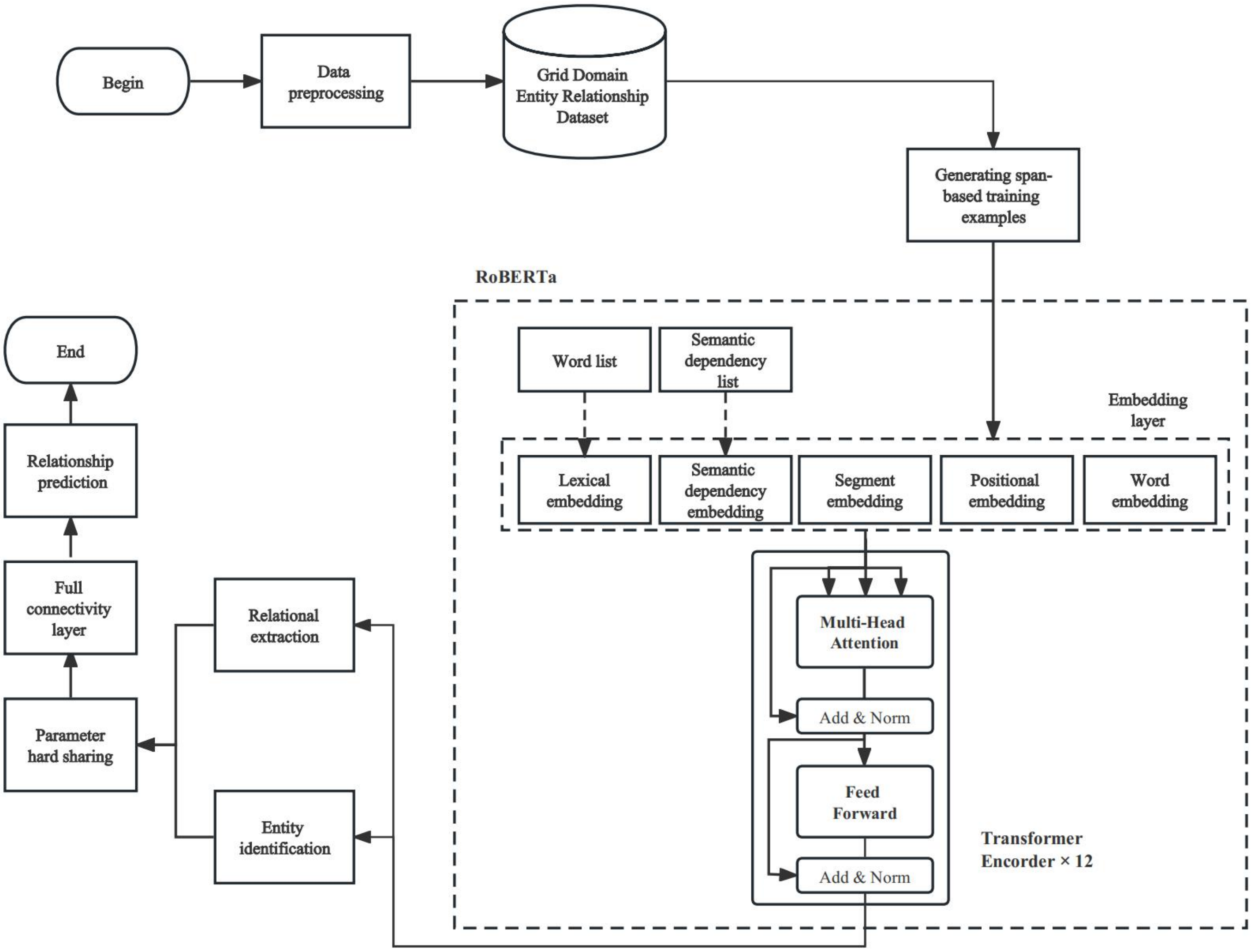

Figure 1. Flow chart. Examples of data from the database are shown in Table 2. In the figure, Transformer Encorder × 12 means 12 layers of Transformer Encorder are stacked.

3.1. Pre-Training Language Models

The RoBERTa-wwm-ext pre-training model was used, which was based on the transformer architecture [25]. It was pre-trained unsupervised on large-scale Chinese textual data to learn rich a priori knowledge and has achieved excellent performance in many natural language processing tasks. RoBERTa is a variant of BERT [26], and based on BERT, it made the following changes:

The dynamic masking strategy can result in distinct mask positions for each training sample during various training iterations. The lexical elements were randomly selected for masking, for the training sample "110 kV Kunlun station," the first round of training replaced the training sample with the special lexical element "110 kV Kunlun <mask>," and the second round of training replaced the training sample with "<mask>10 kV Kunlun Station," and the mask position may change again in the third and fourth rounds. This dynamic strategy enhanced the randomness of the model's input data, consequently boosting the model's learning capacity.

RoBERTa employed entire sentences as input across documents and eliminated the need for next-sentence prediction.

It leveraged larger training batches and a more extensive pre-training dataset to enhance the generalization capacity of the model.

3.2. Semantic Dependencies and Lexical Embedding

The encoding layer transformed the input text sequence into a sequence of high-dimensional vector representations. These vectors incorporated information regarding the word encoding, paragraph context, and positional characteristics of the input text. They were designed to capture dependencies within the input sequence over extended distances and provided a more comprehensive representation of the profound semantic information of the text.

The generated training examples were fed into the RoBERTa embedding layer. Word, position, and paragraph embedding were fused to introduce semantic dependency and lexical embedding so that the embedding layer obtained semantic dependency representations through semantic embedding. The lexical embedding also allowed its layer to obtain lexical representations. This allowed the model to learn the connection between different entity representations, which improved its performance. The process of the encoding layer is summarized as follows:

1. Encoding Layer:

A semantic dependency lexicon was constructed using the language technology platform (LTP) [27] to perform semantic dependency disambiguation of utterances, mapping it to its index.

The generated training examples were fed into the RoBERTa embedding layer. Word, positional, and paragraph embedding were fused to introduce semantic dependency and lexical embedding so that the embedding layer obtained semantic dependency representations through semantic embedding and lexical representations through lexical embedding. This allowed the model to learn the connection between different entity representations, which improved its performance. The process is summarized as follows:

A semantic dependency lexicon was first constructed using the language technology platform (LTP) to perform semantic dependency disambiguation of utterances, mapping it to its index.

The semantic dependency information was mapped onto a graph. For example, the sentence "On June 30, 110 kV Kunlun station Guangkun line," was decomposed into semantic dependency information via LTP: 'TIME', 'TIME', 'TIME', 'TIME', 'TIME', 'TIME', 'TIME', 'mPUNC', 'FEAT'. This allowed the model to learn more information about semantic dependency related information to more effectively model the relationship between entities, as shown in Figure 2.

A lexically labeled word list was then constructed, and the utterances were lexically labeled using jieba disambiguation, which mapped to the index value of the lexically labeled word list.

The semantic dependency analysis operated independently of the syntactic structure. It established direct connections between dependency arcs of linguistic units based on immediate semantic associations, and annotated them with the relevant semantic relations. It focused on the semantic factual or logical relationships between real words. The structure of the syntax tended to vary with literal words, while the semantics were able to transcend changes in the surface of a sentence to reach its essence. Compared with the syntactic dependency analysis, the semantic dependency analysis expressed deeper semantic information, which was especially suitable for the Chinese language.

After the text was labeled with semantic dependency annotation and lexical annotation, it was converted to the index values presented in Tables 3 and 4, and the two vectors were embedded in the RoBERTa coding layer.

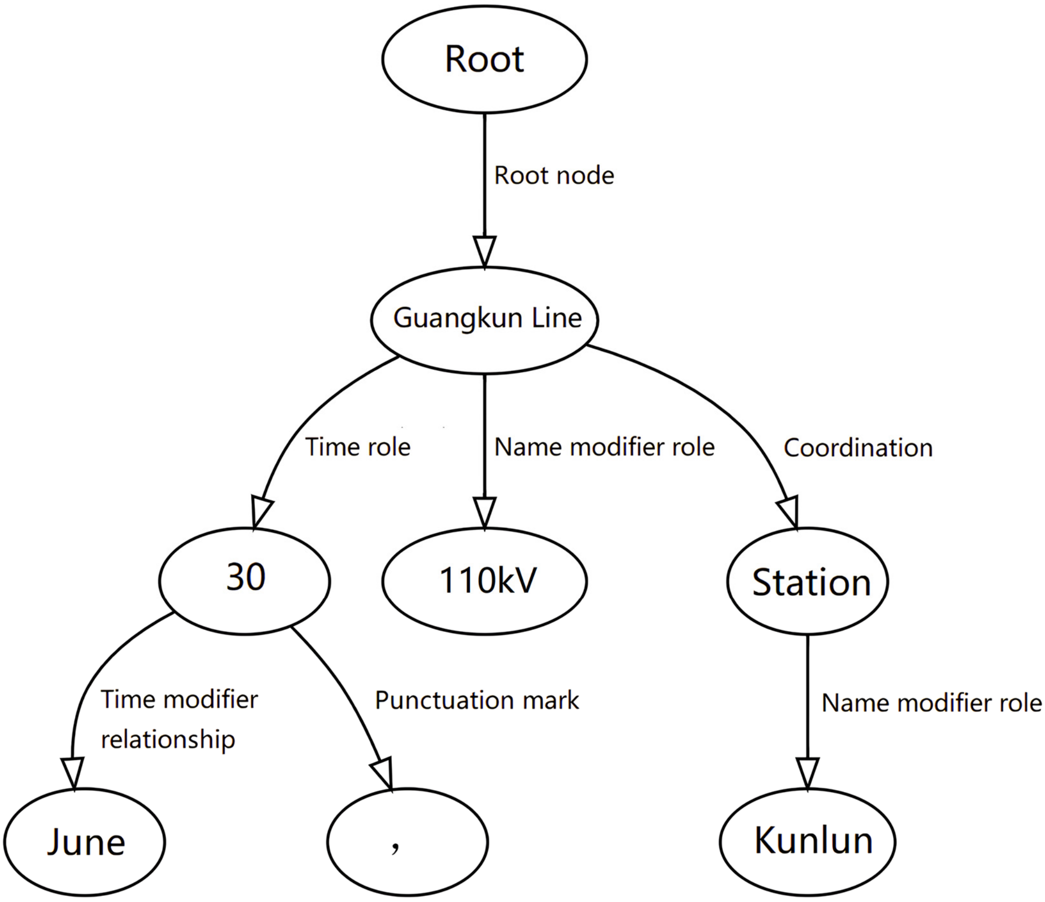

Figure 2. Example of a semantic dependency graph showing the semantic dependency analysis of "30 June, 110 kV Kunlun station Guangkun line," and the construction of a semantic dependency analysis diagram.

Table 3. Construction of a semantic dependency mapping index lexicon (only partially shown).

Type of Relationship	Tag	Example	Index
Agent	Agt	I sent her a bouquet of flowers.	1
Description	Feat	He is fat (grow --> fat)	2
Time	Time	There was a Li Bai in the Tang Dynasty (Tang Dynasty <-- there was)	3
Possessor	Poss	He has a good book (He <-- has)	4
Punctuation Marker	mPunc	, . !	5
Content	Cont	He heard firecrackers (hear --> firecrackers)	6
Root	Root	Core nodes of the sentence	7

Table 4. Construction of a lexical mapping index word list (only partially shown).

Tag	Part of Speech	Index
NN	noun	1
JJ	adjective	2
DT	determiner	3
IN	preposition	4
MD	modal	5
RP	particle	6

As the RoBERTa model was pre-trained by a large number of corpora and saved a large amount of corpus information, a direct addition to the original embedding layer can perturb the original corpus information and generate noise. Parameters a and b were set via the neural network to respectively learn the weights of semantic dependency information and lexical embedding information, and to learn the appropriate fusion weights with the optimization of the model.

The training data input was passed through the RoBERTa coding layer to obtain the word embedding containing positional information, paragraph information, word encoding information, semantic dependency information, and lexical embedding (as shown in Figure 3), $X_{embedding}$:

$$X_{\text{embedding}} = X_{\text{word}} + X_{\text{segment}} + X_{\text{positional}} + a \cdot X_{\text{semantics}} + b \cdot X_{\text{phrase}} \tag{1}$$

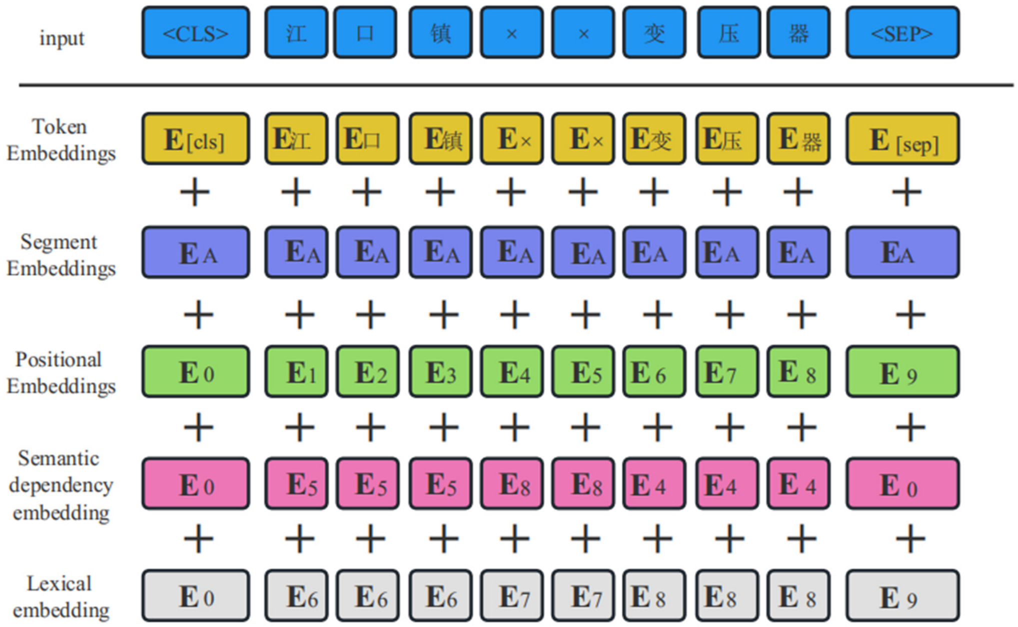

Figure 3. Embedding layer fusion. The subscripts of semantic dependent embeddings and lexical embeddings were the index values of the processed mapping to the word list.

3.3. Semantic Dependencies and Lexical Embedding

The training data were encoded with RoBERTa word embedding to learn certain contextual features.

The weights were first learned for the word embedding $X_{embedding}$ after the attention mechanism:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad \text{where } Q, K, V = X_{\text{embedding}} \tag{2}$$

In the RoBERTa architecture, multi-head attentional learning of word embeddings was required for learning multi-channel information:

$$\text{MultiHead}\left(X_{\text{embedding}}, X_{\text{embedding}}, X_{\text{embedding}}\right) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O \quad \text{where } \text{head}_i = \text{Attention}\left(X_{\text{embedding}}, X_{\text{embedding}}, X_{\text{embedding}}\right) \tag{3}$$

To mitigate the problems of gradient explosion and gradient vanishing within deep models, the $X_{embedding}$ vectors were residually concatenated with the multi-head attention:

$$\mathrm{sy}' = \mathrm{X_{embedding}} + \mathrm{MultiHead}\left(\mathrm{X_{embedding}}, \mathrm{X_{embedding}}, \mathrm{X_{embedding}}\right) \tag{4}$$

A layer normalization of sy' was performed to compute the mean and variance on each sample to normalize the hidden layers in the neural network to a standard normal distribution and accelerate the convergence:

$$\begin{gathered}\mathrm{LN}(\mathrm{sy}') = \alpha \times \frac{\mathrm{sy}' - \mu_L}{\sqrt{\sigma_L^2 + \epsilon}} + \beta \\ \text{where } \mu_L = \frac{1}{m}\sum_{i=1}^{m} \mathrm{sy}' \\ \text{where } \sigma_L^2 = \frac{1}{m}\sum_{i=1}^{m} (\mathrm{sy}' - \mu_L)^2\end{gathered} \tag{5}$$

where the scaling parameters α and β were learnable parameters. ϵ prevents the equation from dividing by the zero value and m was the number of neurons.

Next, the output of layer normalization was passed through a feed-forward neural network:

$$\mathrm{FFN}(x) = \max(0, xW_1 + b_1)W_2 + b_2 \tag{6}$$

The above formula consists of two linear transformations, with a ReLU activation in the middle, and x denoting the output $LN(\mathrm{sy}')$ of the layer normalization.

Finally, the residuals were connected and the layer normalized:

$$H = \mathrm{FFN}(x) + x \tag{7}$$

The output value H after 12 layers of encoder was obtained via the above formula.

The span-based data annotation format were different to the traditional sequence annotation. This allowed for the strengthening of the boundary characteristics of the candidate span and to connect more closely with the textual information. In addition, the representations of the span start position and the end position were spliced. The corresponding formulas are given here:

$$\mathrm{h\_o_{start}} = H_{12}[:, O_{start}] \tag{8}$$

$$\mathrm{h\_o_{end}} = H_{12}[:, O_{end}] \tag{9}$$

$$\mathrm{obj} = \mathrm{Concat}(\mathrm{h\_o_{start}}, \mathrm{h\_o_{end}}) \tag{10}$$

where H_{12} denotes the output of the last layer of RoBERTa, O-start denotes the start index of the entity and O-end denotes the end index. Equations (8) and (9) yielded the trained features for the span start position and end position, respectively. Equation (10), concat, spliced these three features so that the model contained the relevant information of the guest entity along with the relationship extraction.

A similar approach was used for the master entity:

$$\mathrm{h\_s_{start}} = H_{12}[:, S_{start}] \tag{11}$$

$$\mathrm{h\_s_{end}} = H_{12}[:, S_{end}] \tag{12}$$

$$\mathrm{sub} = \mathrm{Concat}(\mathrm{h\_s_{start}}, \mathrm{h\_s_{end}}) \tag{13}$$

The contextual information of the main and guest entities was passed through the fully connected layer to obtain the predicted scores. The two were then added together to obtain the predicted probabilities of the various relationships that were then passed through the softmax layer to obtain the final predicted relationship types:

$$\mathrm{E_p} = \mathrm{sigmoid}(\mathrm{W_1} * \mathrm{obj} + \mathrm{b_1}) \tag{14}$$

$$\mathrm{r_p} = \mathrm{sigmoid}(\mathrm{W_2} * \mathrm{sub} + \mathrm{b_2}) \tag{15}$$

$$\mathrm{R_p} = \mathrm{E_p} + \mathrm{r_p} \tag{16}$$

$$\mathrm{R_{type}} = \mathrm{softmax}(\mathrm{R_p}) \tag{17}$$

The guest entities were passed through a fully connected layer and then a softmax layer to obtain the predicted entity type:

$$\mathrm{E_{type}} = \mathrm{softmax}(\mathrm{E_p}) \tag{18}$$

The predicted entity types and predicted relationships performed cross-entropy loss with the true values and the two loss values were added together (i.e., the parameters were hard-shared) to jointly participate in the optimization of the model. This made the model take into account both entities and relationships to reduce its error propagation:

$$\mathrm{E_{loss}} = -\sum(\mathrm{E_{real}} * \log(\mathrm{E_{type}}) + (1 - \mathrm{E_{real}}) * \log(1 - \mathrm{E_{type}})) \tag{19}$$

$$\mathrm{E_{loss}} = -\sum(\mathrm{R_{real}} * \log(\mathrm{R_{type}}) + (1 - \mathrm{R_{real}}) * \log(1 - \mathrm{R_{type}})) \tag{20}$$

$$\mathrm{loss} = \mathrm{E_{loss}} + \mathrm{R_{loss}} \tag{21}$$

4. Experiments and Results Analysis

The experimental setup included the PyTorch framework, CUDA version 11.1, Ubuntu operating system, and an NVIDIA RTX 3090 (24 G) graphics card. A learning rate linear warm-up strategy was implemented to ensure a high model stability during the initial stages of training and to accelerate the convergence. A model evaluation was conducted every 2500 training steps to save the models that had high accuracy at this stage. The remaining parameters of the model are shown in Table 5.

Table 5. Parameter setting.

Parameter	Value
Learning rate	2×10^{-5}
Batch size	3
Epoch	20
lstm_embedding_size	1024
Hidden size	768
Bert model	RoBERTa-wwm-ext
Embedding size	512
Optimizer	AdamW

4.1. Criteria for Evaluation

In this experiment, the precision, recall and F_1 value were used to evaluate the performance of the model:

$$\mathrm{precision} = \mathrm{correct}_{\mathrm{num}} / \mathrm{predict}_{\mathrm{num}} \tag{22}$$

$$\text{recall} = \text{correct}_{\text{num}} / \text{golden}_{\text{num}} \tag{23}$$

$$\text{F}_1 = \frac{2*\text{precision} * \text{recall}}{\text{precision} + \text{recall}} \tag{24}$$

Here, precision denoted the precision rate, recall signified the recall rate, and correct $_{num}$ indicated the count of accurate predictions whereas predict$_{num}$ represented the total number of predictions, golden$_{num}$ represented the number of labeled entities, and the F1 value was the average of the precision rate and recall rate. This was capable of balancing the influence of the precision rate and the recall rate, and reflecting the performance of the model in a more comprehensive way.

4.2. Results and Analysis

The performance of the model was evaluated using the F_1 value, F_1-overlap, accuracy, and recall on the entity relationship dataset in the grid field constructed in Chapter 1 for training and evaluation.

A comparison experiment was conducted to verify the effectiveness of the proposed model for the grid data relationship extraction. The grid field data entity relationship extraction model was compared to the BiLSTM-CRF, BERT-CE, BERT-CRF, and BERT-BiLSM-CRF models. The obtained results are shown in Table 6. It can be seen that the proposed model had the optimal recognition effect in grid business data relationship extraction, compared with the other models. On the gird dataset, it had precision, recall, and F_1 values of 89.55, 85.91, and 87.92%, respectively.

Table 6. Comparison between the performance of different relational extraction models. The * symbol denotes the RoBERTa with incorporated semantic dependency and lexical embedding strategies.

Index	Model	Precision/%	Recall/%	F_1-Score/%
1	BiLSTM-CRF	62.08%	64.58%	63.30%
2	BERT-CE	87.07%	83.33%	85.16%
3	BERT-CRF	87.15%	83.51%	85.29%
4	BERT-BiLSM-CRF	87.95%	84.51%	86.19%
5	RoBERTa-CE *	**89.55%**	**85.91%**	**87.92%**

(1) The BiLSTM-CRF model used word2vec as the embedding layer. However, its word vectors were static and cannot be adjusted according to the input context words. Therefore, it had a low performance on the power grid dataset, with an F_1 value of only 63.30%.

(2) The BERT-CE model used the BERT pre-trained language model as the embedding layer to adequately capture the contextual representation of the characters and thus had better access to the deep semantic information. On the grid dataset, the F_1 value of the model was 85.16%.

(3) The BERT-CRF model added conditional random field (CRF) to the BERT pre-trained language model, which improved its F_1 value by 0.13% compared with the cross entropy loss module via sequentially annotating the output of BERT.

(4) The BERT-BiLSTM-CRF model also used the BERT pre-trained language model to capture the contextual semantics of the grid business data, while utilizing recurrent neural networks to capture richer meanings. It also used the CRF for classification. It had an F_1 value of 86.19%, which presented an improvement of 1.03% compared with Model 2.

(5) For the RoBERTa-CE model with embedded semantic dependencies and lexicality, the RoBERTa pre-trained language model with dynamic MASK was used to capture the contextual semantics of the grid business data, embed semantic dependencies and

lexicality, and efficiently combine the information of the subject and object in order to improve their associativity and strengthen the linkage of the relational entities. Therefore, compared with the above model, the recognition performance was significantly improved, and the F_1 value was 87.92% on the grid dataset. Compared with Model 2, the F_1 value improved by 2.76%, which presented optimal recognition results.

Table 7 presents the ablation experimental results of the model. The removal of the lexical embedding and semantic dependency embedding under the benchmark of the RoBERTa model reduced the performance of the model by 1.08%, which demonstrated that the correlation between related entities can be enhanced through effective embedding of lexical and semantic dependencies. Model 4, using the original RoBERTa, only differed by 0.78% compared to model 5 using the embedded lexical and semantic dependency BERT, while model 8, without any embedding enhancement, reduced the performance by 1.68% compared to model 4. Since RoBERTa used larger training data and was more powerful than BERT for deep characterization of sentences, the addition of lexical and semantic dependency embedding effectively narrowed the gap between the two and enhanced the deep characterization ability of the model.

Table 7. Ablation experiments, where Pos_id denotes the addition of a lexical embedding vector SemDep_id denotes the addition of a semantic dependency embedding vector, and CE denotes the cross-entropy loss function. A check mark indicates that the module is used.

Index	RoBERTa	BERT	Pos_id	SemDep_id	CE	F_1-Score/%
1	✓		✓	✓	✓	87.92%
2	✓		✓		✓	87.16%
3	✓			✓	✓	87.41%
4	✓				✓	86.84%
5		✓	✓	✓	✓	85.96%
6		✓	✓		✓	85.47%
7		✓		✓	✓	85.66%
8		✓			✓	85.16%

In summary, the proposed model had superior F_1 performance on the entity-relationship dataset within the grid domain, compared with the benchmark models.

5. Conclusions

In this paper, a relationship extraction model for the grid field was designed through combining semantic dependency and lexical embedding using the RoBERTa model. The text context depth characterization information was obtained through the RoBERTa pre-training model. The lexical and semantic dependency information were embedded in the RoBERTa embedding layer, weights were set for the two types of information, and the fused weights were automatically learned based on the model optimization for effective embedding. The cross-entropy function was used for training. The model effectively enhanced the deep semantic characterization ability, which improved the accuracy of relationship recognition between the different entities. The efficiency and superiority of the proposed approach were then verified on a relational extraction dataset which was curated within the grid domain. The obtained results can be summarized as follows:

(1) The combination of the semantic dependency and lexical embedding in the RoBERTa model improved the F_1 value by 2.76% compared with the original BERT model. This indicated that the semantic dependency and lexical embedding effectively enhanced the relationship extraction accuracy.

(2) Model 4, which used the original RoBERTa, exhibited only a 0.78% discrepancy compared with Model 5, which incorporated embedded lexical and semantic dependency. On the other hand, Model 8, which did not use any embedding enhancement, had a

1.68% decrease in performance compared with Model 4. The inclusion of lexical and semantic dependent embeddings effectively narrowed the gap between RoBERTa and BERT, and enhanced the deep characterization ability of the model.

(3) To label the relationship as quintuple spanning pairs and relationships, one entity should be selected as the main entity in the labeled data. The remaining entities should be enumerated as guest entities. The main entity and guest entities should contain entity type information. The generated main entity and guest entity set, along with the text information, can be used as a training set. Different subjects should be classified into different groups for parallel training, which strengthens the connection between the entities.

In future work, we aim to use the trained model to extract information from the data provided by the power grid. A domain knowledge graph will be constructed to manage the data in an appropriate manner. Through leveraging the graph, the ability to extract valuable insights within the grid domain can be enhanced, and well-informed decisions for grid-related enterprises can be made. Furthermore, the evaluation of the capacity of the model for generalization through extending its application to other domains is of interest.

Author Contributions: Methodology, Q.M.; writing—original draft preparation, Q.M.; Conceptualization, X.Z.; Formal analysis, X.Z.; writing—review and editing, Y.D.; visualization, Y.C.; software (Re_Project, v.2023.09.18), D.L.; resources, X.Z.; data curation, D.L. and Y.C.; supervision, Y.D. 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 No. AA20302002-3) and Innovation Project of China Southern Power Grid Co., Ltd. (202201161).

Institutional Review Board Statement: No ethical approval was required for this study.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Conflicts of Interest: Author Qi Meng , Xixiang Zhang and Yun Dong was employed by the company Guangxi Power Grid Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest. The authors declare that this study received funding from Guangxi Power Grid Co. The funder had the following involvement with the study: providing data support for the experiments, participating in the development of the technical program of the thesis, its implementation, and the writing of the thesis.

References

1. Nayak, T.; Majumder, N.; Goyal, P.; Poria, S. Deep Neural Approaches to Relation Triplets Extraction: A Comprehensive Survey. *Cogn. Comput.* **2021**, *13*, 1215–1232.
2. Kumar, S. A survey of deep learning methods for relation extraction. *arXiv* **2017**, arXiv:1705.03645.
3. Cui, M.; Li, L.; Wang, Z.; You, M. A survey on relation extraction. In Proceedings of the China Conference on Knowledge Graph and Semantic Computing, Chengdu, China, 26–29 August 2017; Springer: Berlin/Heidelberg, Germany, 2017; pp. 50–58.
4. Dileep, G. A survey on smart grid technologies and applications. *Renew. Energy* **2020**, *146*, 2589–2625.
5. Wu, N.; Zhao, H.; Ji, Y.; Sun, S. Chinese Named Entity Recognition for a Power Customer Service Intelligent Q&A System. In Proceedings of the 2021 International Conference on Intelligent Computing, Automation and Applications (ICAA), Nanjing, China, 25–27 June 2021; pp. 363–368.
6. Ji, S.; Pan, S.; Cambria, E.; Marttinen, P.; Philip, S.Y. A survey on knowledge graphs: Representation, acquisition, and applications. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 494–514.
7. Wenjun, H.; Liang, H.; Haoshuai, X.; Wei, Y. RoRED: Bootstrapping labeling rule discovery for robust relation extraction. *Inf. Sci.* **2023**, *629*, 62–76.
8. Ke, J.; Wang, W.; Chen, X.; Gou, J.; Gao, Y.; Jin, S. Medical entity recognition and knowledge map relationship analysis of Chinese EMRs based on improved BiLSTM-CRF. *Comput. Electr. Eng.* **2023**, *108*, 108709. [CrossRef]
9. Guo, L.; Yan, F.; Li, T.; Yang, T.; Lu, Y. An automatic method for constructing machining process knowledge base from knowledge graph. *Robot. Comput. -Integr. Manuf.* **2022**, *73*, 102222. [CrossRef]
10. Wan, Q.; Wei, L.; Zhao, S.; Liu, J. A Span-based Multi-Modal Attention Network for joint entity-relation extraction. *Knowl.-Based Syst.* **2023**, *262*, 110228. [CrossRef]

11. Liu, Z.; Li, H.; Wang, H.; Liao, Y.; Liu, X.; Wu, G. A novel pipelined end-to-end relation extraction framework with entity mentions and contextual semantic representation. *Expert Syst. Appl.* **2023**, *228*, 120435. [CrossRef]
12. Tang, R.; Chen, Y.; Qin, Y.; Huang, R.; Zheng, Q. Boundary regression model for joint entity and relation extraction. *Expert Syst. Appl.* **2023**, *229*, 120441.
13. Gao, C.; Zhang, X.; Li, L.; Li, J.; Zhu, R.; Du, K.; Ma, Q. ERGM: A multi-stage joint entity and relation extraction with global entity match. *Knowl.-Based Syst.* **2023**, *271*, 110550. [CrossRef]
14. Jaradeh, M.Y.; Singh, K.; Stocker, M.; Both, A.; Auer, S. Information extraction pipelines for knowledge graphs. *Knowl. Inf. Syst.* **2023**, *65*, 1989–2016. [CrossRef] [PubMed]
15. Barducci, A.; Iannaccone, S.; La Gatta, V.; Moscato, V.; Sperlì, G.; Zavota, S. An end-to-end framework for information extraction from Italian resumes. *Expert Syst. Appl.* **2022**, *210*, 118487. [CrossRef]
16. Fabregat, H.; Araujo, L.; Martinez-Romo, J. Deep neural models for extracting entities and relationships in the new RDD corpus relating disabilities and rare diseases. *Comput. Methods Programs Biomed.* **2018**, *164*, 121–129. [CrossRef] [PubMed]
17. Zhong, Z.; Chen, D. A Frustratingly Easy Approach for Entity and Relation Extraction. In Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Online, 6–11 June 2021; pp. 50–61.
18. Ye, D.; Lin, Y.; Li, P.; Sun, M. Packed Levitated Marker for Entity and Relation Extraction. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), Dublin, Ireland, 22–27 May 2022; Association for Computational Linguistics; pp. 4904–4917.
19. Yan, J.; Bracewell, D.B.; Ren, F.; Kuroiwa, S. Integration of Multiple Classifiers for Chinese Semantic Dependency Analysis. *Electron. Notes Theor. Comput. Sci.* **2009**, *225*, 457–468. [CrossRef]
20. Yin, L.; Gao, Q.; Zhao, L.; Zhang, B.; Wang, T.; Li, S.; Liu, H. A review of machine learning for new generation smart dispatch in power systems. *Eng. Appl. Artif. Intell.* **2020**, *88*, 103372.
21. Sun, Z.; Li, X.; Sun, X.; Meng, Y.; Ao, X.; He, Q.; Wu, F.; Li, J. ChineseBERT: Chinese Pretraining Enhanced by Glyph and Pinyin Information. In Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Online, 1–6 August 2021; Association for Computational Linguistics; pp. 2065–2075.
22. Kleenankandy, J.; Abdul Nazeer, K.A. An enhanced Tree-LSTM architecture for sentence semantic modeling using typed dependencies. *Inf. Process. Manag.* **2020**, *57*, 102362. [CrossRef]
23. Liu, P.; Guo, Y.; Wang, F.; Li, G. Chinese named entity recognition: The state of the art. *Neurocomputing* **2022**, *473*, 37–53.
24. Wu, F.; Liu, J.; Wu, C.; Huang, Y.; Xie, X. Neural Chinese Named Entity Recognition via CNN-LSTM-CRF and Joint Training with Word Segmentation. In *The World Wide Web Conference (WWW'19)*; Association for Computing Machinery: New York, NY, USA, 2019; p. 3342.
25. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17), Long Beach, CA, USA, 4–9 December 2017; Curran Associates Inc.: Red Hook, NY, USA, 2017; pp. 6000–6010.
26. Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1, Minneapolis, MN, USA, 2–7 June 2019; Association for Computational Linguistics; pp. 4171–4186.
27. Che, W.; Feng, Y.; Qin, L.; Liu, T. N-LTP: An Open-source Neural Language Technology Platform for Chinese. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Abu Dhabi, United Arab Emirates, 7–11 December 2022; pp. 42–49.

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.