frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 13 June 2024
DOI 10.3389/fenrg.2024.1395621

OPEN ACCESS

EDITED BY
Yang Yang,
Nanjing University of Posts and Telecommunications, China

REVIEWED BY
Fei Mei,
Hohai University, China
Zhihao Yang,
Yangzhou University, China
Jianyong Zheng,
Southeast University, China
Tuan Anh Le,
Technical University of Berlin, Germany
Deyagn Yin,
Changzhou University, China

*CORRESPONDENCE
Haiou Cao,
✉ haiou79013@163.com

RECEIVED 04 March 2024
ACCEPTED 21 May 2024
PUBLISHED 13 June 2024

CITATION
Cao H, Zhang Y, Ge Y, Shen J, Tang C, Ren X and Chen H (2024), Intelligent substation virtual circuit verification method combining knowledge graph and deep learning.
Front. Energy Res. 12:1395621.
doi: 10.3389/fenrg.2024.1395621

Intelligent substation virtual circuit verification method combining knowledge graph and deep learning

Haiou Cao[1]*, Yue Zhang[2], Yaming Ge[1], Jiaoxiao Shen[3], Changfeng Tang[1], Xuchao Ren[1] and Hengxiang Chen[2]

[1]State Grid Jiangsu Electrical Power Company, Nanjing, China, [2]State Grid Nanjing Power Supply Company, Nanjing, China, [3]State Grid Suzhou Power Supply Company, Suzhou, China

The correctness of the intelligent electronic devices (IEDs) virtual circuit connections in intelligent substations directly affects the stability of the system operation. Existing verification methods suffer from low efficiency in manual verification and lack uniformity in design specifications. Therefore, this paper proposes a virtual circuit automatic verification method that combines knowledge graphs with deep learning. Firstly, this method utilizes expert knowledge and relevant standard specifications to construct a knowledge graph of virtual circuits, integrating knowledge from historical intelligent substation configuration files into the knowledge graph. Then, leveraging multi-head attention mechanisms and Siamese neural networks, it achieves matching between the textual descriptions of virtual terminals and standard virtual terminal descriptions. Additionally, a verification process for the virtual terminal port address string is incorporated. Finally, experimental validation confirms the effectiveness of the proposed method and strategy, further enhancing the accuracy of virtual circuit verification.

KEYWORDS

intelligent substation, virtual circuit, knowledge graph, natural language processing, Chinese short-text matching

1 Introduction

As intelligent substations advance rapidly, the IED within substations has exhibited a notable surge in the variety of types, a marked increase in automation levels, and a continuous strengthening of safety requirements (Song et al., 2016; Huang et al., 2017). The use of optical fibers in smart substations replaces the signal transmission through cables in traditional substations. The correct configuration of the virtual circuit formed by the logical connection of virtual terminals in the configuration file is a prerequisite for the reliable and stable operation of the substation. For a typical 220 kV substation, there can even be thousands of virtual circuits, and manual verification alone is difficult to ensure the accuracy of virtual circuit configuration. In terms of automatic verification, it is affected by non-standard design, resulting in low verification accuracy and poor universality. To illustrate the need for validation and the validation process, consider a simplified example of a bus protection device and a line protection device in an intelligent substation. These devices need to communicate correctly through virtual circuits to ensure system stability. The bus protection device may send a signal indicating a fault condition to the line protection device.

If the virtual circuit connecting these devices is misconfigured, the fault signal may not be received correctly, leading to potential system failures. Validating this virtual circuit involves ensuring that the virtual terminal descriptions and port address data match the expected configurations. These characteristics present both opportunities and challenges for the development of automatic verification of virtual circuits for IED. IED enable communication by adhering to the unified digital standard protocol IEC 61850 (Selim Ustun and M. Suhail Hussain, 2020). Similarly, the testing of IED's virtual circuits is also based on parsing IED files in accordance with the IEC 61850 protocol to perform virtual circuit verification (Zhao et al., 2023).

Currently, in verifying IED in intelligent substations through digital network communication, manual inspection is required to assess the correctness of corresponding virtual circuit circuits, including events, signals, and data, based on the descriptions of secondary equipment virtual terminals (Cui et al., 2018). This process is essential for validating the effectiveness of protection and assessing the accuracy of device parameter settings (Fan et al., 2020). However, due to the lack of a unified standard for virtual terminal descriptions, the verification of virtual circuits often relies on manual inspection. With a multitude of equipment in substations, the manual verification of virtual circuits is characterized by inefficiency, prolonged testing cycles, and susceptibility to variations due to personnel experience and working conditions. This approach is prone to omissions and errors, necessitating substantial effort for error correction following detection.

In addressing this issue, scholars have explored automated verification methods for virtual circuit circuits. Hao et al. (2020) instantiated the configuration of IED virtual circuits through sub-template matching of substation configuration description (SCD) files, offering a novel approach to virtual circuit verification. However, the scalability and generalizability of this method need improvement. Zhang et al. (2015) proposed an expert system based on SCD files to match virtual terminals and perform intelligent substation verification, but the classification is relatively simple, and the process is time-consuming. Some scholars used deep learning and other intelligent algorithms to solve the verification problem in intelligent substations. Oliveira et al. (2021) proposed a deep learning based intelligent substation schedule monitoring method. Chen et al. (2021) proposed the implementation of IED self configuration based on the use of natural language processing technology. Ren et al. (2020) utilized DCNN for text classification of intelligent recorder configuration files to achieve port address mapping. While effective for longer texts, this method has limitations with short texts and does not consider other port address data. Wang et al. (2018) calculated the semantic similarity of virtual terminals using word embedding techniques for virtual circuit matching, demonstrating good matching results. However, their tokenization method for word embedding does not consider the global semantic information of the text, leaving room for improvement. Through the above discussion, it is evident that automated verification schemes for IED virtual circuits primarily rely on two approaches: one utilizing template-based matching that has high accuracy but low generalizability, and the other employing intelligent algorithms or deep learning to classify or calculate similarity in configuration information, offering strong adaptability but it is limited to single-device configurations.

Knowledge graph is essentially a semantic network that includes various semantic connections between different entities (Chen et al., 2020), exhibiting superior interpretability and data storage structural performance (Wang et al., 2017). It has been widely applied in various fields, such as data retrieval, recommendation systems, and knowledge reasoning (Guan et al., 2019). For the power system, knowledge graph can integrate dispersed knowledge within the power system, effectively excavating useful latent rules from massive textual information within the power system (Liu et al., 2023). At the same time, the graph data structure of knowledge graph also provides great convenience for human understanding. Currently, research on knowledge graph in the field of power is still in its nascent stage, with relevant literature mainly focusing on application exploration and macro framework design. Li and Wang (2023) proposed a multi-level, multi-category knowledge graph application framework for assisting decision-making in power grid fault handling and preliminarily elaborated on the key technologies and solution approaches within the framework; Tian et al. (2022) utilized the graph structure of knowledge graph to express textual information and their relationships, extracting the information required to construct knowledge graph from operation and maintenance reports, realizing the automatic construction of knowledge graph, and proposing an automatic retrieval method for equipment operation and maintenance.

Therefore, this study proposes a secondary virtual circuit automatic verification method combining knowledge graph with deep learning. This method integrates the advantages of prior knowledge matching and similarity calculation of virtual terminal information, utilizes knowledge graph for virtual circuit information querying and extraction, and employs an improved Siamese neural network to calculate the similarity of virtual terminal information. Thus, achieving accurate and efficient secondary virtual circuit automatic verification.

To provide a comprehensive understanding of our proposed method, this study is organized as follows. Section 2 introduces the construction method of a secondary virtual circuits knowledge graph. Section 3 provides a detailed explanation of the automatic matching process of virtual terminal information based on the improved Siamese neural network model, along with the automated verification process of virtual circuits. Section 4 presents the experimental results and performance evaluation, demonstrating the effectiveness of the proposed method. Finally, Section 5 discusses the implications of our findings and suggests avenues for future research in this domain.

2 Knowledge graph construction

2.1 Intelligent substation configuration file structure

In the intelligent substation secondary system, a single optical fiber can transmit multiple channels of data and the one-to-one correspondence of data transmission is ensured by using virtual terminals. Virtual terminals are not physical terminals; they are used

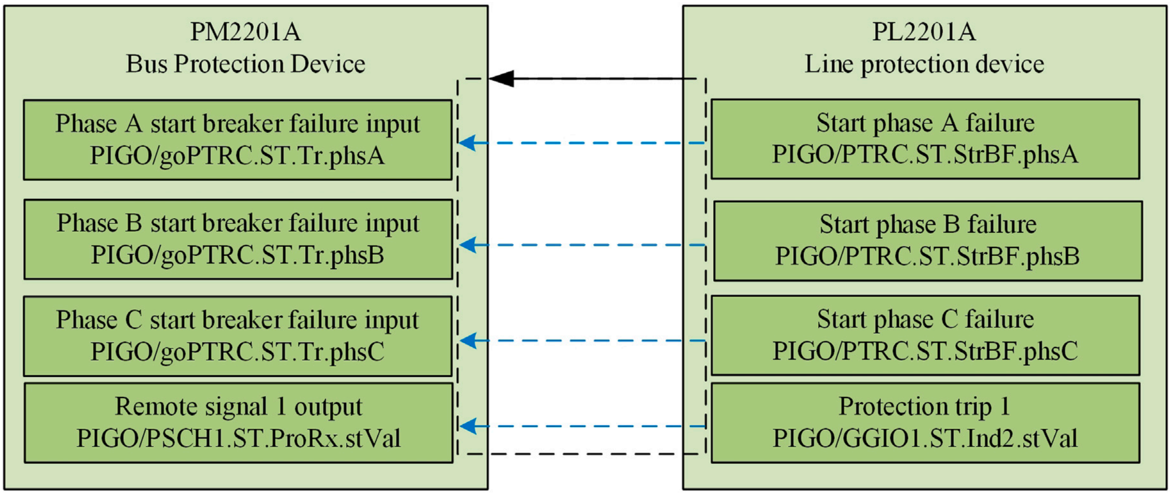


FIGURE 1
Schematic diagram of virtual terminal connections.

to identify loop signals between IEDs and serve as signal connection points during the transmission of generic object-oriented substation event (GOOSE) and sampled value (SV) messages. From the perspective of information transmission, there are two main categories: input virtual terminals and output virtual terminals. From the perspective of message types, there are two main categories: SV virtual terminals and GOOSE virtual terminals.

Taking a certain bus protection device and line protection device as an example, a schematic diagram of virtual terminal connections is illustrated in Figure 1. The bus protection device and line protection device are connected via an optical fiber (solid line with arrow in the diagram), facilitating unidirectional transmission of information in the form of datasets. The virtual terminals are interconnected through virtual connections (dashed line with arrow in the diagram), forming a virtual circuit and thereby achieving a one-to-one correspondence of data.

IEC 61850 is a communication standard among IEDs in intelligent substations that ensures interoperability between different devices. The data modeling technique in IEC 61850 is object-oriented and characterized by a hierarchical tree structure. The order from top to bottom is as follows: physical device, logical device (LD), logical nodes (LNs), data object (DO), and data attribute (DA). Each object in this data structure has a unique data index within the model. The virtual terminal data format specified by the IEC 61850 standard is represented as LD/LN. DO (DA), corresponding to logical device/logical node. data object (data attribute). For the SV virtual terminal, the data attribute (DA) is generally left blank. Therefore, under the IEC 61850 standard, it is ensured that virtual terminals exhibit significant similarities in their data formats.

Additionally, to facilitate interoperability among devices from different manufacturers, designers often include a brief Chinese description on virtual terminals during the equipment design process. This practice aims to aid designers in better understanding and distinguishing virtual terminals associated with devices from various manufacturers.

2.2 Virtual circuit knowledge graph

In an intelligent substation, IEDs encompass a variety of devices, including relay protection devices, merging units, smart terminals, and intelligent recorders. The virtual circuits corresponding to different IEDs are markedly distinct, and virtual circuits of the same type of IED may exhibit certain variations under different states. The verification of virtual circuits involves three aspects:

(1) Precisely determining all the virtual terminals and IEDs essential for configuring the virtual circuit verification.
(2) Clearly defining the hierarchical paths and descriptive features corresponding to different types of IED data formats.
(3) Assessing the correctness of the mapping relationships of virtual terminals based on their distinctive features.

The construction process of knowledge graph involves the following steps:

Step 1: Collect SCD files and other relevant documentation from various intelligent substations. This data provides the raw input needed to build the knowledge graph.

Step 2: Use natural language processing techniques to parse the SCD files and extract relevant information such as IED types, virtual terminal descriptions, and port addresses.

Step 3: Map the extracted information to the ontology. This involves identifying the appropriate entities and relationships in the knowledge graph and ensuring that the extracted data fits into this structure.

Step 4: Use a Siamese neural network with multi-head attention to calculate the semantic similarity between extracted virtual terminal descriptions and the standardized descriptions in the ontology.

Step 5: Integrate the matched data into the knowledge graph, creating links between historical data and standardized models. Manual verification is performed for matches below a predefined threshold to ensure accuracy.

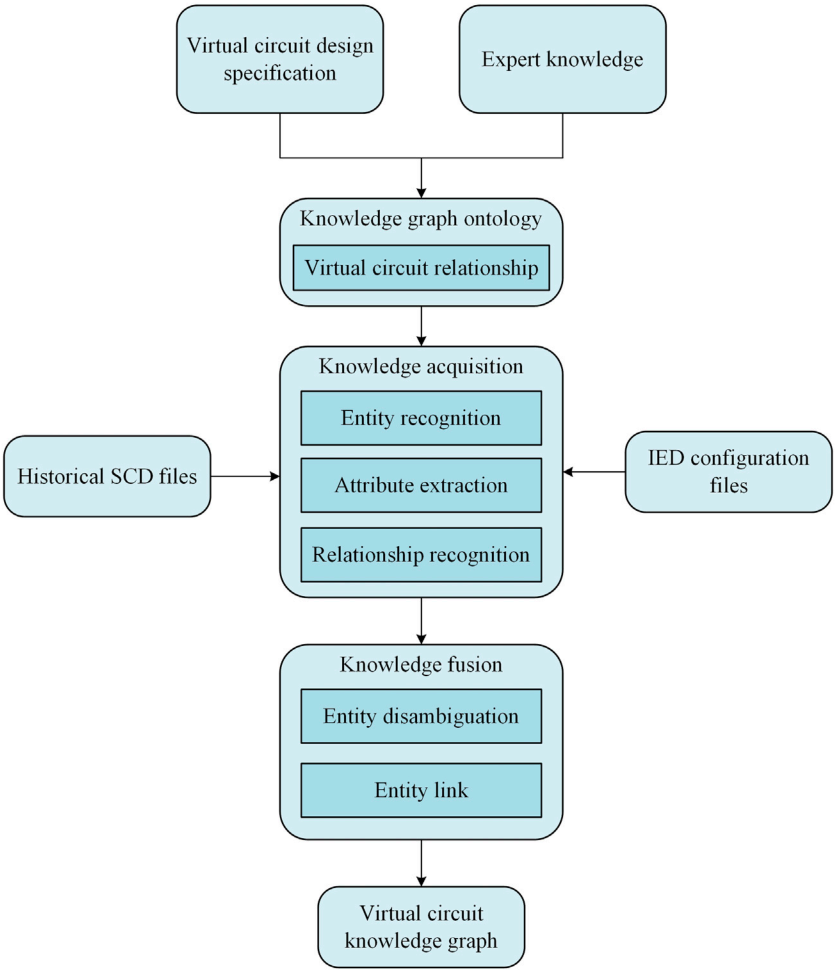


FIGURE 2
Knowledge graph construction method.

Step 6: Continuously update the knowledge graph with new data and validation results to improve its accuracy and comprehensiveness.

This study employed a top–down approach to construct a knowledge graph for the association of virtual circuits. Standardization of virtual terminals for relevant IEDs in intelligent substations was achieved, establishing a standardized model for virtual terminals. Expert knowledge in the context of intelligent substations refers to the domain-specific insights and rules provided by experienced engineers and technicians. This knowledge encompasses the correct configurations of virtual circuits, typical faults, and the standard practices for designing and maintaining these systems. By modeling virtual circuits based on expert knowledge according to IED types and associating them with the standardized virtual terminal data relationships established according to relevant specifications, a knowledge graph ontology was formed. The construction process of knowledge graph is shown in Figure 2. The ontology includes:

(1) Entities: These are the core components such as IEDs, virtual terminals, LNs, DO, and attributes.
(2) Relationships: These define how entities interact with each other. For example, an IED may have multiple virtual terminals, and each terminal can be linked to specific data attributes.
(3) Attributes: These are the properties or characteristics of the entities, such as the type of data transmitted, the logical node identifiers, and port address configurations.

The ontology provides a standardized model that facilitates consistent representation and querying of virtual circuit information.

Utilizing historical data from intelligent substations for knowledge learning, this study parsed historical SCD files of intelligent substations to acquire the associative relationships of IED virtual circuits. Based on the standardized virtual circuit model in the knowledge graph ontology, relationships between standardized virtual circuits and actual virtual circuits within IEDs are established. The constructed virtual circuit knowledge graph is shown in Figure 3. Although there are significant format differences between virtual terminal address configuration data and textual description information, information logic among the same type of IEDs shares strong correlations, and the naming and expressions of identical entities are generally uniform.

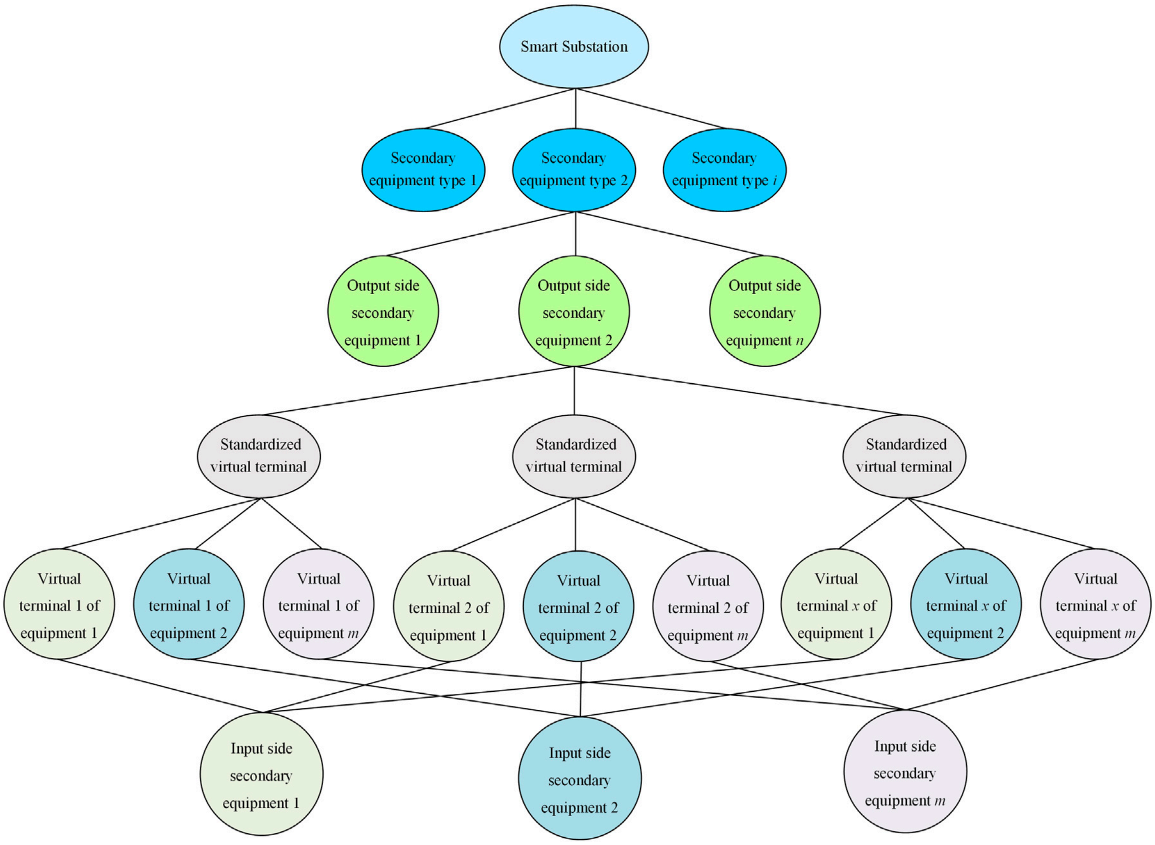

Smart Substation
Secondary equipment type 1
Secondary equipment type 2
Secondary equipment type *i*
Output side secondary equipment 1
Output side secondary equipment 2
Output side secondary equipment *n*
Standardized virtual terminal
Standardized virtual terminal
Standardized virtual terminal
Virtual terminal 1 of equipment 1
Virtual terminal 1 of equipment 2
Virtual terminal 1 of equipment *m*
Virtual terminal 2 of equipment 1
Virtual terminal 2 of equipment 2
Virtual terminal 2 of equipment *m*
Virtual terminal *x* of equipment 1
Virtual terminal *x* of equipment 2
Virtual terminal *x* of equipment *m*
Input side secondary equipment 1
Input side secondary equipment 2
Input side secondary equipment *m*

FIGURE 3
Knowledge graph structure diagram.

Using dictionary list data obtained from processing SCD files, three categories of information were extracted: IED device type, virtual terminal address configuration data, and textual description. The entities are linked in triplets: {device type, virtual terminal address configuration data, textual description}. The initial extraction results contained a considerable amount of redundant data. This study removed numerical information from virtual terminal address configuration data, retaining only alphanumeric string information. An index is then constructed by merging IED type and textual description information. The number and types of virtual terminal address configuration data in each index are tallied. Based on the statistical results, the triplets are transformed, retaining only one triplet for each identical virtual terminal address configuration data within the same index.

Establishing a knowledge graph using virtual circuit information from historical SCD files has limitations. Therefore, in this study, based on the similarity calculation between the virtual circuit and the standardized model of virtual circuit data, IED virtual terminals are automatically matched to standardized virtual terminals. Subsequent manual verification is conducted to establish the relationship between IED virtual terminals information and the standard virtual terminals information, achieving knowledge fusion. Knowledge fusion involves integrating information from various sources to create a comprehensive and coherent knowledge graph. In our method, knowledge fusion occurs in two main steps:

(1) Historical Data Integration: We parse historical SCD files to extract virtual circuit configurations from previously implemented substations. This data includes IED types, virtual terminal descriptions, and port address configurations.
(2) Standardization and Matching: The extracted data is compared against the standardized models defined in the ontology. Virtual terminals are matched to their standardized counterparts based on semantic similarity calculations and expert-defined rules. This step ensures that the knowledge graph accurately reflects both historical configurations and standardized practices.

3 Virtual terminal automatic matching

Due to the predominantly Chinese short-text nature of the virtual terminal textual descriptions, a text-matching model is proposed that integrates a multi-head attention mechanism and a Siamese network. The model utilizes a mixed vector of characters and words as input to enhance semantic information, employing a bidirectional gated recurrent unit (Bi-GRU) instead of bidirectional long short-term memory (Bi-LSTM) to reduce parameters and expedite training speed. The multi-head attention mechanism is introduced as a separate module, employing an autoencoding layer to capture semantic features from different perspectives. A Siamese

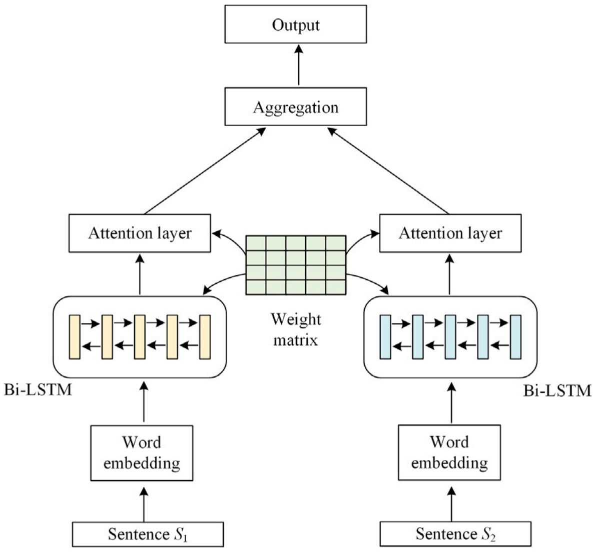


FIGURE 4
Siamese neural network.

network is constructed to transform sentences of varying lengths into sentences of equal length, placing them in the same semantic space. Weight sharing is implemented to reduce half of the training workload. In the interaction layer, the mixed vector of characters and words for one sentence interacts with that of another sentence, utilizing the multi-head attention mechanism to acquire interactive semantic features.

3.1 Siamese neural network model

The schematic diagram of the Siamese neural network model is shown in Figure 4.

The Siamese network model (Liang et al., 2018) is broadly divided into three parts: preprocessing, shared neural network, and information aggregation. The processing flow of the Siamese network is as follows: firstly, preprocess the obtained sentences to obtain word vector representations; then, encode the obtained sentence representations using a Siamese network constructed with Bi-LSTM and attention mechanisms; finally, aggregate information from the sentence representations processed through the Siamese network.

(1) Preprocessing: To begin, sentences containing contextual information are obtained. Each word in the sentence is then represented using word vectors. A sentence is represented as $\bar{V}_{S_1} \in \mathbb{R}^{m\times d}$, $\bar{V}_{S_2} \in \mathbb{R}^{n\times d}$, where m and n represent the lengths of sentences S_1 and S_2 , respectively, and d is the dimensionality of the word vectors.
(2) Shared neural network: After obtaining the representation of sentences, the word vectors pass through a Bi-LSTM algorithm to encode information about the sentences. A standard Bi-LSTM algorithm is employed, representes by Eqs 1–3 as follow:

$$\overrightarrow{h_t} = f\left(U_1 x_t, U_3 \overrightarrow{h_{t-1}}, \overrightarrow{b_t}\right) \tag{1}$$

$$\overleftarrow{h_t} = f\left(U_2 x_t, U_4 \overleftarrow{h_{t+1}}, \overleftarrow{b_t}\right) \tag{2}$$

$$y_t = concat\left(\overrightarrow{h_t}, \overleftarrow{h_t}\right) \tag{3}$$

where, $\overrightarrow{h_t}$, $\overleftarrow{h_t}$, x_t and y_t represent the forward propagation hidden layer state, the backward propagation hidden layer state, the input values of the neurons, and the output values of the hidden layer state at time t, respectively; $\overrightarrow{h_{t-1}}$ represents the forward propagation state at time t-1 and $\overrightarrow{h_{t+1}}$ represents the backward propagation state at time $t+1$; U_1, U_2, U_3 and U_4 denote the weight matrices corresponding to different components, respectively; $\overrightarrow{b_t}$ and $\overleftarrow{b_t}$ represent bias vectors in the forward propagation hidden layer and backward propagation hidden layer, respectively; *concat* denotes concatenate operation.

Following the Bi-LSTM layer, an attention mechanism is introduced. The shared-weight neural network consists of the aforementioned Bi-LSTM layer and an attention mechanism layer.

(3) Information aggregation: The processed representations of the two sentences need to undergo information fusion, and common fusion methods include fully connected neural networks, calculating the cosine similarity, and the Manhattan distance between the two vectors.

3.2 Interactive text matching model integrating multi-head attention mechanism and siamese network

The Siamese network model does not fully leverage interactive information between texts, nor does it adequately capture the representation capabilities of the text. Therefore, this study proposed an improved model, the Interactive Text Matching Model, which integrates a multi-head attention mechanism with a Siamese network, as shown in Figure 5.

Chinese writing is logographic, meaning that each character represents a word or a meaningful part of a word. This differs from alphabetic languages where words are composed of letters. Therefore, the network uses character-based embeddings in addition to word-based embeddings to capture the nuances of Chinese text. This dual representation ensures that both individual characters and their combinations are effectively represented.

This model initially preprocesses two sentences at the input layer, obtaining mixed vectors for both sentences. The resulting mixed vectors undergo normalization through a regularization layer. Subsequently, these processed vectors are separately fed into the difference unit and interaction unit.

In the difference unit, the input texts are first encoded using a bidirectional GRU, constructing a Siamese network. The Manhattan distance is then employed to aggregate the encoded information. Simultaneously, within the interaction unit, the auto-encoding layer is utilized to encode the input sentences separately, forming another Siamese network. Subsequently, the semantic features from the self-encoding layer undergo interaction through the multi-head

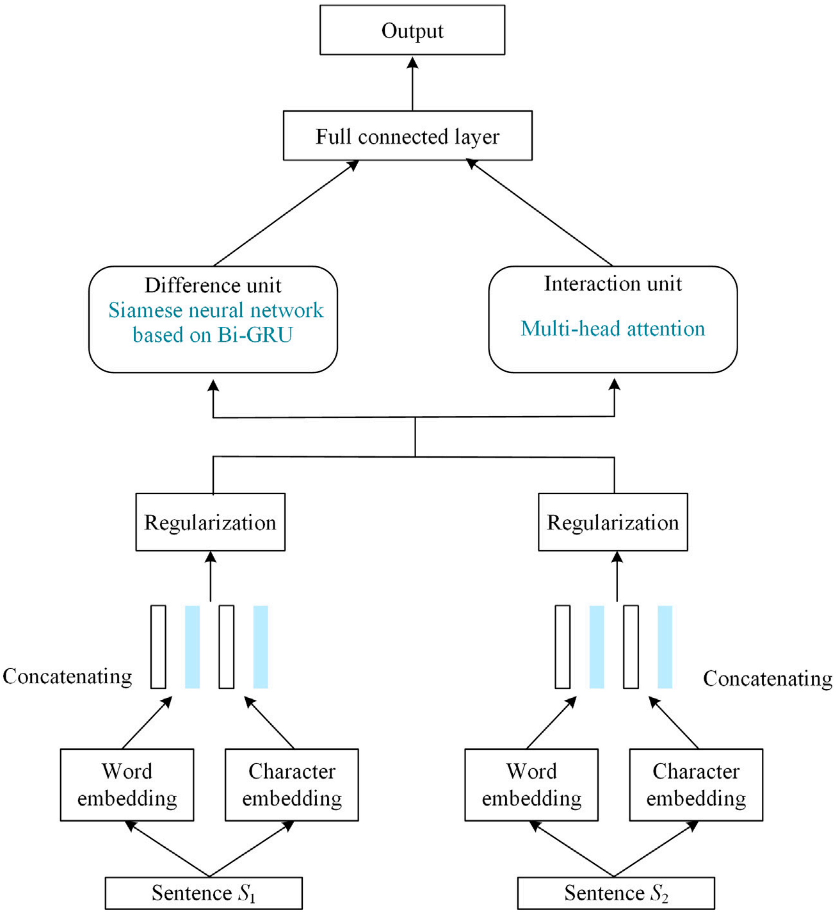


FIGURE 5
Interactive text matching model.

attention mechanism's interaction layer. Finally, the interactive information, along with the information from the difference unit, is output to the fully connected neural network in the output layer. The classification result is obtained through a sigmoid function.

3.2.1 Difference unit

Firstly, the training set is tokenized into word and character level representations using the Jieba segmentation tool. For input sentences S_i, after word and character level segmentation, two representations are obtained: a word sequence and a character sequence. By loading the pre-trained Word2vec model weights, each word or character can be mapped to a vector, extracting the corresponding word/character vectors, the formal description of sentence S_i is given by Eqs 4, 5:

$$S_i = \left\{W_i^1, W_i^2, \ldots, W_i^{lw}\right\} \tag{4}$$

$$S_i = \left\{C_i^1, C_i^2, \ldots, C_i^{lc}\right\} \tag{5}$$

where lw and lc represent the number of words and characters in sentence S_i, respectively.

The length of the word vectors is extended to match the length of the character vectors. Subsequently, concatenating the two representation vectors yields the final hybrid representation vector V_{S_i}, which combines both character and word embeddings. This hybrid word-character vector is then fed into the normalization layer.

$$V_{S_i} = \left\{v_{S_i}^1, v_{S_i}^2, \ldots, v_{S_i}^{N_i}\right\} \tag{6}$$

where N_i represent the length of vector V_{S_i}.

In this study, the Bi-GRU structure was employed to replace the Bi-LSTM algorithm in the Siamese network for text information encoding. A GRU (Cho et al., 2014) is a variant of LSTM with a simplified architecture. It employs an update gate in place of the forget and input gates in LSTM and introduces a new hidden unit. The model structure is simpler than that of LSTM and is represented by Eqs 7–10:

$$z_t = \sigma(W_z x_t + U_z h_{t-1}) \tag{7}$$

$$r_t = \sigma(W_r x_t + U_r h_{t-1}) \tag{8}$$

$$\widetilde{h_t} = \tanh(r_t \cdot U_a h_{t-1} + W_a x_t) \tag{9}$$

$$h_t = (1 - z_t) \cdot h_{t-1} + z_t \cdot \widetilde{h_t} \tag{10}$$

where z_t and r_t represent the update gate and reset gate, respectively; $\widetilde{h_t}$ represents the aggregation of the input x_t and the output of the previous hidden layer h_{t-1}; σ denotes the sigmoid function, and tanh denotes the hyperbolic tangent function; W_z, U_z, W_r, U_r, U_a, and W_a are the weight matrices used in training.

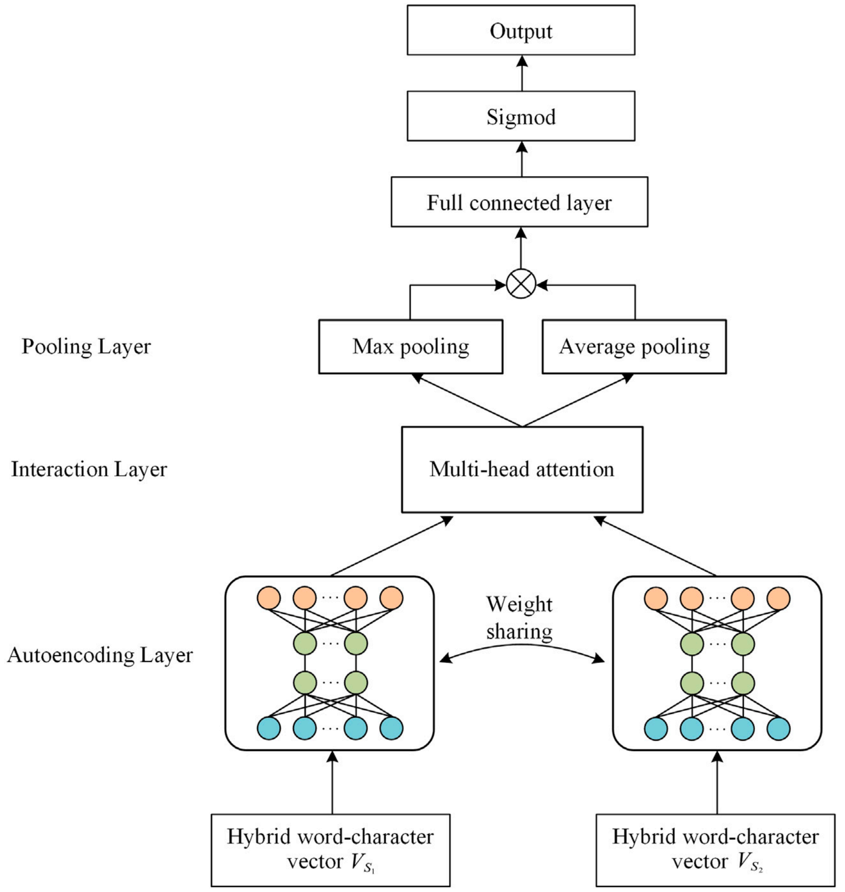


FIGURE 6
Interaction unit structure diagram.

The update gate indicates how much of the current hidden state is inherited by the new hidden state, while the reset gate indicates how much information from the past hidden state should be ignored and reset using the current input. GRU has fewer parameters compared to LSTM, making training faster and requiring less data.

3.2.2 Interaction unit

The Siamese network model introduces the attention mechanism directly after the neural network to extract essential word information from sentences. However, it does not adequately extract the semantic features of sentence pairs and let them interact. Therefore, this study treated the multi-head attention mechanism as a separate unit for computation, extracting interaction features between text pairs. The basic attention network is segmented into different sub-networks, learning more semantic features from various perspectives, with the aim of achieving a comprehensive interaction between semantic features. The interaction unit comprises the autoencoding layer, interaction layer, and pooling layer, as illustrated in Figure 6.

(1) Autoencoding layer

In the autoencoding layer, the mixed-word vectors V_{S_1} and V_{S_2} are separately encoded. This layer consists of two parts, the encoder and the decoder, forming a symmetrical structure. Both the encoder and decoder typically comprise three-layer neural networks, including input, hidden, and output layers, as illustrated in Figure 7. Here, the output of the encoder serves as the input to the decoder, and the outputs of the encoder and decoder are represented by Formulas 11, 12, respectively:

$$Z_i = f_1(W_e V_{S_i} + b_e) \tag{11}$$

$$V'_{S_i} = f_2(W_h \hat{H} + b_h) \tag{12}$$

where W_e and W_h represent weight matrices; b_e and b_h are bias vectors; f_1 and f_2 are activation functions; $\hat{H}$ represents the hidden layer vector.

The autoencoder network extracts high-dimensional features through the encoder, reducing the dimensionality to process the output text features, denoted as Z_i. The decoder, employing a symmetric network structure, reconstructs the input V_{S_i} of the encoder to obtain V'_{S_i} aiming to fit an identity function using neural networks and enhance the feature extraction capability of the encoder. The reconstruction process utilizes mean square error as the loss function, and L2 regularization is employed to prevent model overfitting, thereby improving the model's performance on the test set.

(2) Interaction layer

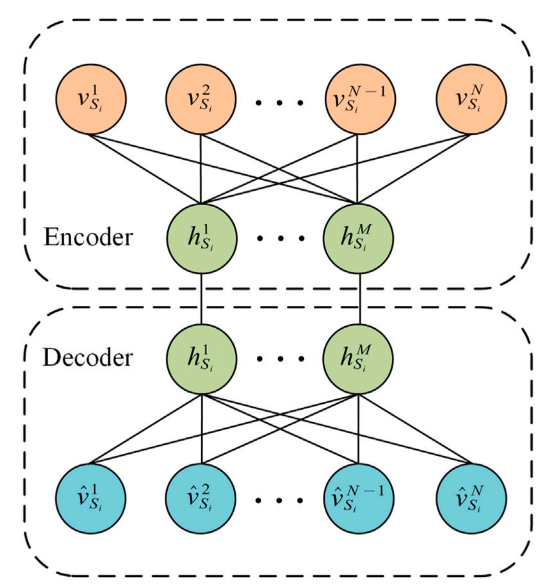


FIGURE 7
Autoencoder structure diagram.

In the encoding layer, each text is individually encoded, obtaining diverse contextual semantic information to enhance the representation of text features. However, due to the Siamese network's inclination to represent semantic feature vectors of different lengths in the same semantic space, the interaction information between texts is overlooked. Compared to typical Chinese text, descriptions of virtual terminals consist of shorter sentences, and there is no contextual semantic relationship between sentences. Therefore, the interaction layer is introduced to separately extract word and sentence interaction features between text pairs. Each word can interact with words in the other text, capturing syntactic and semantic dependencies between text pairs.

This layer is introduced to address the deficiency of the Siamese network in semantic interactions. Multi-head attention mechanisms are employed to extract interaction information between the two texts. The mixed-word vectors after autoencoding $V_{S_1}^{'}$and $V_{S_2}^{'}$are concatenated to form V_{text}, and then utilize multi-head attention based on scaled dot-product attention (Vaswani et al., 2017) to capture interactions between pairs of sentences and can be described by Eq. 13:

$$Attention\,(Q,K,V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}V\right) \tag{13}$$

where Q, K, and V represent the query vector, key vector, and value vector respectively, with d_k being the dimensionality of key vector. In this study, self-attention is employed to extract features of V_{text}, thus Q, K, and V are all equal to V_{text}.

The attention mechanism acts as semantic feature extraction and encoding, providing each word with three vectors. Each operation involves calculating the similarity between a word's query vector and all key vectors through dot-product, resulting in weight coefficients representing the word's relevance to other words. These coefficients are then used to weigh all value vectors to obtain semantic encoding. Multi-head attention allows different attention weights to be assigned to different positions, acquiring better semantic information and effectively preventing overfitting, as describes in Eqs 14, 15 (Guo et al., 2019):

$$Multi-head\,(Q,K,V) = concat\,(head_1, head_2, \ldots, head_H)W^O \tag{14}$$

$$head_i = Attention\left(QW_i^Q, KW_i^K, VW_i^K\right) \tag{15}$$

where W^O, W_i^Q, W_i^K, W_i^V represent the weight matrix.

(3) Pooling layer

The pooling layer's function is to extract global features from the word vector sequences. This includes both max pooling and average pooling. Each dimension of the word vector reflects different information, and pooling operations help to extract comprehensive information from the word vectors. Considering that max pooling retains prominent information from the word vectors, while average pooling retains information from all word vectors (Bieder et al., 2021), we simultaneously use both max pooling and average pooling, and then concatenate the results, as describes in Eqs 16–18:

$$V_{text,avg} = \sum_{i=1}^{N_{V_{text}}} \frac{V_{text,att}^i}{N_{V_{text}}} \tag{16}$$

$$V_{text,\max} = \max_{i\in\left[1,N_{V_{text}}\right]} V_{text,att}^i \tag{17}$$

$$V_{text,pool} = concat\left(V_{text,avg}, V_{text,\max}\right) \tag{18}$$

where $V_{text,avg}$, $V_{text,\max}$, $V_{text,pool}$ represent average pooling result, maximum pooling result, and pooling layer result. $V_{text,att}$ represents the output result of multi-head attention in the interaction layer; $N_{V_{text}}$ represents the length of vector $V_{text,att}$.

For the output, the results X_{itac} and X_{diff} from the interaction unit and the difference unit are hybrid through a fully connected neural network. The final classification result is obtained using the sigmoid activation function, as describes in Eqs 19, 20:

$$R = \text{sigmod}\left(FC\left(X_{itac}, X_{diff}\right)\right) \tag{19}$$

$$FC\,(X) = \text{relu}\,(Wx + b) \tag{20}$$

where W represents the weight matrix; b represent bias vector. relu represents the rectified linear unit

The model utilizes mean squared error as the loss function. Additionally, the Adam optimization algorithm is applied to enhance the convergence speed, as describes by Eq. 21 (Reyad et al., 2023).

$$LossMSE = \frac{1}{N}\sum_1^N \left(Y_i - \hat{Y}_i\right) \tag{21}$$

where N represents the number of samples; Y_i and $\hat{Y}_i$ represent the true value and predicted value, respectively.

3.3 Verification method based on port address data

Through experiments, it was observed that accurately calculating the textual similarity values can reliably detect the desired virtual terminals. However, in some cases, there may be

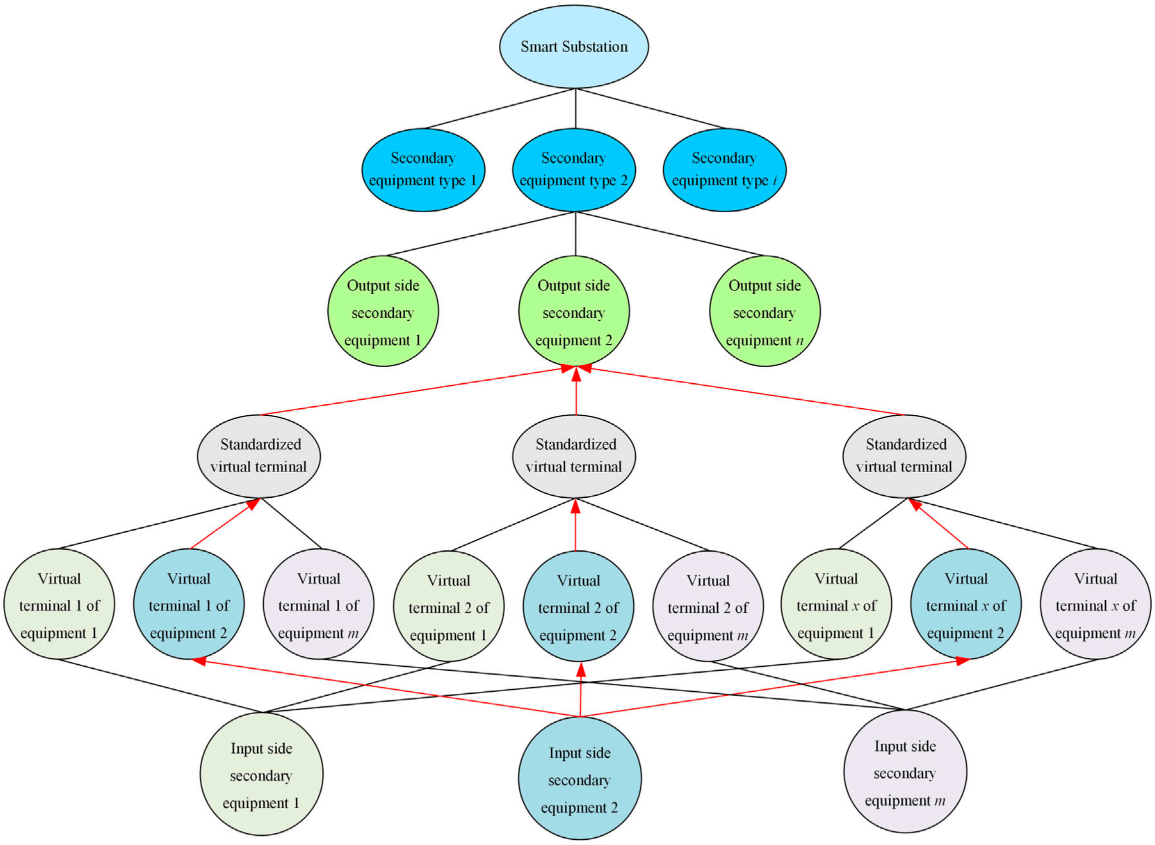

Smart Substation
Secondary equipment type 1
Secondary equipment type 2
Secondary equipment type *i*
Output side secondary equipment 1
Output side secondary equipment 2
Output side secondary equipment *n*
Standardized virtual terminal
Standardized virtual terminal
Standardized virtual terminal
Virtual terminal 1 of equipment 1
Virtual terminal 1 of equipment 2
Virtual terminal 1 of equipment *m*
Virtual terminal 2 of equipment 1
Virtual terminal 2 of equipment 2
Virtual terminal 2 of equipment *m*
Virtual terminal *x* of equipment 1
Virtual terminal *x* of equipment 2
Virtual terminal *x* of equipment *m*
Input side secondary equipment 1
Input side secondary equipment 2
Input side secondary equipment *m*

FIGURE 8
Automatic verification path.

more than one matching result. This is attributed to incomplete or the repetitive construction of certain virtual terminal description texts. To distinguish virtual terminal ports in such cases, reliance on port address data information becomes crucial, as each virtual terminal's port address is unique.

Therefore, this study introduced a verification process based on port address configuration data to address situations where textual descriptions alone cannot establish mapping for the data. Leveraging port address configuration data helps discern matching relationships due to the systematic naming patterns inherent in port address configuration data, with most strings carrying practical significance.

The longest common substring (LCS) method proposed by Amir et al. (2020) was employed to assess the similarity of virtual terminal port addresses. The calculation of the virtual terminal port address similarity involves evaluating the similarity of four attributes constituting the virtual terminal port address: logical device, logical node, data object, and data attribute. The weighted average of these individual similarities is taken as the overall similarity of the virtual terminal port address.

3.4 The automatic verification process for virtual circuit

In intelligent substations, the virtual circuit information of various IEDs is integrated into the station-wide configuration SCD file. Before verification, the SCD file is parsed to obtain the IEDs that need to be checked. The output interface addresses and descriptions are extracted from the SCD file to form the output interface information for the corresponding IED.

During the verification process, the knowledge graph provides the reference model against which current virtual circuit configurations are compared. This comparison helps in identifying discrepancies and potential faults. The verification process begins with the IEDs in the knowledge graph. The matching is conducted along the path indicated by the arrows in Figure 8. The device model information parsed from the SCD file is used to search for the corresponding IED in the knowledge graph. The virtual terminal information extracted from the SCD file for the identified IED are then matched with the virtual terminal information in the knowledge graph, specifically matching them to standard virtual terminal information. This process enables the detection of all virtual circuits for the specified IED, achieving automatic verification of virtual circuits in intelligent substations.

If the corresponding IED cannot be matched in the knowledge graph during verification, a search is conducted based on the device type to find the standard output interface addresses corresponding to that device type in the knowledge graph. Subsequently, the similarity between the virtual terminal information of the IED and the standard virtual terminal information is calculated. Any matches below a predefined threshold undergo manual confirmation to establish the association between the IED's

TABLE 1 Partial data samples.

Order	Input virtual terminal		Output virtual terminal	
	Description	Address	Description	Address
1	A-phase voltage sampling value	MU/UATVTR1. Vol1	Voltage A-phase measurement value	SVLDO1/SVINUATVTR1. Vol2. Inst mag. I
2	B-phase voltage sampling value	MU/UBTVTR1. Vol1	Voltage B-phase measurement value	SVLDO1/SVINUBTVTR1. Vol2. Inst mag. I
3	C-phase voltage sampling value	MU/UCTVTR1. Vol1	Voltage C-phase measurement value	SVLDO1/SVINUCTVTR1. Vol2.inst mag. I
4	Merge unit optical port transmission optical power	PIGO/GOINGGIO17. AnIn10. mag. f	Bus merging unit board 1 optical port 1 optical power	MUGO/SCLI9. LigIntes. mag. f
5	Intelligent terminal network port 1 abnormality	PIGO/GOINGGIO 1. SPCSO6. stVal	Bus intelligent terminal network port 1 is abnormal	RPIT/GOAlmGGI O1. Alm6. stVal

virtual terminal addresses and the standard virtual terminal addresses. This association is then stored in the knowledge graph to facilitate knowledge updates.

4 Case study

4.1 Dataset

To validate the effectiveness of the proposed configuration method, a sample set was selected comprising 10 SCD files from 220 kV intelligent substations and 20 SCD files from 110 kV intelligent substations. These samples encompassed 3,869 IEDs and 29,110 records of configured virtual circuits. To conduct experimental analysis, the dataset was divided into training, testing, and validation sets in a ratio of 7:2:1. Since virtual circuit configurations vary significantly across intelligent substations with different voltage levels, data sets were randomly extracted from SCD files of different voltage levels to ensure the generalizability of the proposed method. A subset of the data samples is presented in Table 1. It is evident that the textual descriptions of virtual circuit endpoints share significant similarities, yet there are distinct differences in the address data. In the virtual terminal address data, most strings carry meaningful information; for instance, "MU" signifies a merging unit, and "UATATR" denotes voltage sampling. However, even for the same voltage sampling virtual terminal, the configuration data for virtual terminal addresses can be entirely different. This discrepancy arises due to varying naming conventions among different manufacturers' IEDs, and certain strings such as "mag" and "AnIn" pose challenges in determining their actual significance. Despite such differences in address configuration, the textual descriptions exhibit a high degree of similarity. This observation underscores the rationale behind the main focus of this study on matching information points primarily through text.

4.2 Implementation

Our model incorporated both word and character vectors, utilizing pre-trained Word2Vec (Li et al., 2018) embeddings trained from the Chinese Wikipedia and the electrical vocabulary corpus extracted from the Sogou Input Method, it was then fine-tuned on our dataset for virtual terminal matching with a reduced learning rate of 0.0001 to prevent overfitting on the smaller dataset. The fine-tuning process lasted for 10 epochs. Each vector was set to a dimensionality of 300. To mitigate overfitting and enhance accuracy, Dropout probability was introduced during the experimentation. Following the input layer, sentence vectors are fed into a dual-layer bidirectional GRU with a hidden layer dimension of 128 for each GRU. The attention mechanism comprised eight units, and a rectified linear unit served as the activation function. To achieve optimal experimental results, early stopping was implemented as a training strategy.

In the experiments, the dimension of the word vectors used to initialize embedding vectors was 100, and we fixed the word embedding. The maximum length of the input sequence we chose was 15, and characters that were not in the dictionary were replaced with 0. The model was trained to minimize the cross-entropy of error loss through backpropagation and the Adam optimization algorithm was used with a 0.001 learning rate. The dropout rate was 0.5.

The experimental evaluation drew inspiration from concepts in machine learning, employing precision and recall. The comparative results can be categorized into true positive (TP), false positive (FP), true negative (TN), and false negative (FN). The confusion matrix, as illustrated in Table 2, served as the basis for the performance evaluation using the accuracy and F1-score, as describe by Eqs 22–25:

$$recall\ rate = \frac{TP}{FN + FP} \tag{22}$$

$$precision\ rate = \frac{TP}{FP + TP} \tag{23}$$

$$accuracy = \frac{TN + FP}{FN + TN + FP + TP} \tag{24}$$

$$F1 - score = 2 * \frac{precision\ rate * recall\ rate}{precision\ rate + recall\ rate} \tag{25}$$

4.3 Baseline methods

In the experiment dataset, eight baseline methods were employed, including representation-based text classification

TABLE 2 Confusion matrix.

Reality	Matching result	
	Positive example	Negative example
True	TP	FN
False	FP	TN

models such as Bi-LSTM, CNN, TextCNN (Cao and Zhao, 2018); interactive-based text classification models such as ESIM (Chen et al., 2016), Siamese Bi-LSTM (Li et al., 2021), Attention-Bi-LSTM (Xie et al., 2019); and pre-trained based text classification models such as BERT.

In parameter settings, we strived to ensure consistency across various models as much as possible; in cases where consistency could not be guaranteed, efforts were made to maintain consistency with the original literature. Specifically, for CNN and TextCNN, a two-layer feedforward neural network was employed, with each layer having 128 hidden units. For Bi-LSTM, ESIM, Siamese Bi-LSTM, and Attention-Bi-LSTM, the hidden units for both the Bi-LSTM and feedforward neural network were set to 256. Additionally, consistent with both baseline methods and the proposed approach, Dropout (uniformly set to 0.5) was utilized to mitigate overfitting issues, while training was conducted using the Adam optimizer with a learning rate of 0.001. Regarding BERT, we first separated text pairs in the samples using the SEP token and then inputted them into the BERT-Base model. The vector corresponding to the [CLS] token at the head was extracted as the matching vector for the two sentences, which was then fed into a feedforward neural network to obtain the matching results for the two sentences. Due to convergence issues with a high learning rate, the learning rate for the BERT model was set to 0.0001.

4.4 Results and discussion

4.4.1 Comparative experiment

In order to substantiate the superiority of the proposed text matching model, which integrates multi-head attention and Siamese neural networks, experiments were conducted. The results of the comparisons are presented in Table 3.

The experimental results showcased in Table 3 highlight the effectiveness of our proposed text-matching model compared to traditional text classification approaches. Our proposed text matching model outperformed all baseline models, achieving an impressive accuracy of 97.52% and an F1-score of 97.89%. The integration of multi-head attention and Siamese neural networks enables our model to effectively capture semantic similarities between sentences. The substantial performance improvement of our model over traditional approaches underscores the importance of integrating multi-head attention and Siamese neural networks for text-matching tasks. The superior accuracy and F1-score achieved by our model signify its robustness and effectiveness in capturing semantic relationships between sentences.

Table 3 displayed the training times for nine experimental models on training and test dataset. Bi-LSTM, CNN,Attention-Bi-LSTM and TextCNN demonstrated a clear advantage in training time, while ESIM and Siamese Bi-LSTM required longer training times. Our model integrated Siamese network, Bi-GRU, and muti-head attention, resulting in a complex structure, hence its training time was only surpassed by the structurally complex BERT. In practical applications, since text matching model training generally occurs offline, the model's time complexity requirement is not high, with more emphasis placed on the accuracy of similarity judgment. Additionally, the training time of our model is essentially the same as that of the baseline Siamese Bi-LSTM model. This indicates that in the task of virtual terminal matching, the extraction of interaction features from text pairs based on interaction units has minimal impact on the model's time complexity.

4.4.2 Ablation experiment

In order to comprehensively understand the contribution of different aspects of our proposed model, we conducted an ablation study. We explored various granularities, pooling strategies (average, max), multi-head attention mechanisms, and the impact of incorporating address string validation on experimental results. The findings are summarized in Table 4.

The experimental results indicate that utilizing both character and word embeddings as input can capture more textual information. Employing both max-pooling and average-pooling facilitates effective interaction with semantic information in sentence pairs. The incorporation of attention mechanisms enables the model to capture diverse semantic relationships, thereby enhancing its performance. Additionally, integrating port address validation improves the model's accuracy and F1-score, ensuring its robustness in text matching tasks.

4.4.3 Parameter sensitivity experiment

In the experiment, the variation in the number of heads in the multi-head attention mechanism and the layers in the GRU has a certain impact on the model. Therefore, this study employs sensitivity analysis to investigate and analyze the parameters. Sensitivity analysis primarily involves analyzing the effect of changing a specified piece of information under the assumption of a certain state, designating it as the independent variable, and examining how this designated independent variable affects changes in other variables. In this experiment, we set the independent variables as the number of heads in the multi-head attention mechanism and the layers in the GRU, exploring their effects on the trend of virtual terminal matching results.

(1) Impact of the number of heads in multi-head attention

Multi-head attention enables the aggregation of information from multiple dimensions, facilitating a better understanding of semantic information from different spatial perspectives and preventing overfitting. Leveraging this characteristic, this study tests the number of heads in the multi-head attention mechanism on the validation set, sequentially setting the number of heads as [2, 4, 6, 8, 10] for experimentation. The most suitable number of heads is selected to configure parameters for the pseudo-anchor matching model, as shown in Figure 9 with the experimental results.

According to the experimental results in Figure 9, it is evident that on the validation set, when the number of heads in the multi-head attention mechanism reaches 8, the F1-score and accuracy

TABLE 3 Comparative experiment results.

Model	Accuracy (%)	F1-score (%)	Training time (s)
Bi-LSTM	82.64	83.15	2,310
CNN	83.69	84.27	983
TextCNN	93.77	93.89	1280
ESIM	94.13	94.49	3379
Siamese Bi-LSTM	95.56	95.67	3529
Attention-Bi-LSTM	94.02	94.06	2,581
BERT	96.17	96.28	62,788
Our model	97.52	97.89	3752

TABLE 4 Ablation experiment results.

Model	Accuracy (%)	F1-score (%)
Only word	93.78	93.89
Only character	94.16	94.53
Average pooling	95.23	95.43
Max pooling	95.31	95.39
Without multi-head attention	95.87	96.18
Without port address validation	94.41	94.93
Our model	97.52	97.89

attain their maximum values at 85.01. At this point, the model demonstrates optimal performance. As the number of heads gradually increases or decreases from 8, the F1-score and accuracy of the model progressively decreases. When a single attention head is applied to a sentence, although each word embedding contains embeddings from other words, it is predominantly influenced by the embedding of the word itself. However, utilizing multiple heads enables attention to be concentrated at different positions, aggregating multi-layered information. The advantage of the multi-head attention mechanism lies in its ability to balance the model, providing an additional space for parameter adjustment. This indicates that neither an excessive nor insufficient number of heads is optimal; rather, a balance is required. Both an excess or a deficiency in the number of heads will affect the virtual terminal matching results. Through testing, it is observed that in this model, the optimal performance in virtual terminal matching is achieved when the number of heads is set to 8.

(2) Impact of GRU layers

GRU strengthens the connection between vocabulary and context when processing information, enriching the semantic information of features and alleviating the problem of differences between sentences. The different layers of GRU affect the complexity of the model and have a certain impact on data fusion. In this study, GRU is utilized to process information. To investigate the influence of GRU layers on the model, experiments are set up, selecting GRU

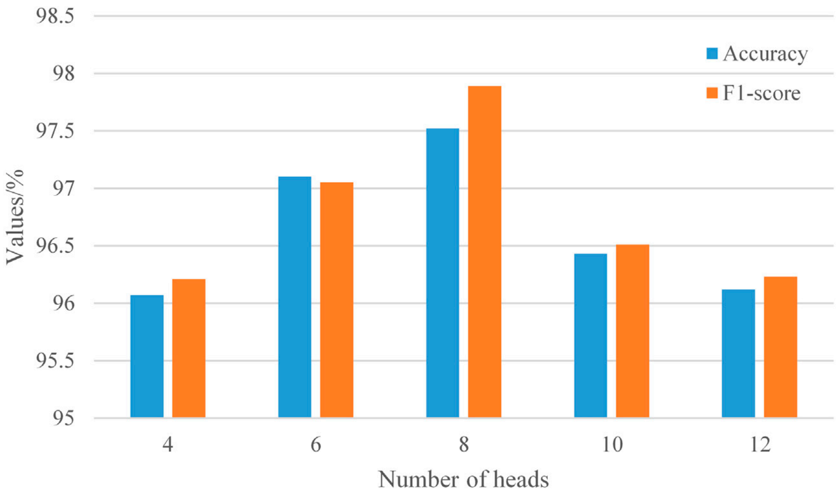


FIGURE 9
Accuracy and F1-score with different number of heads.

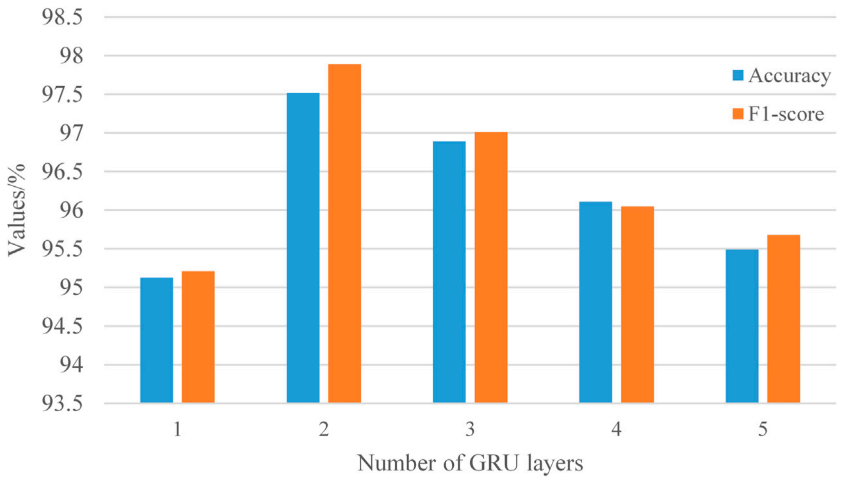
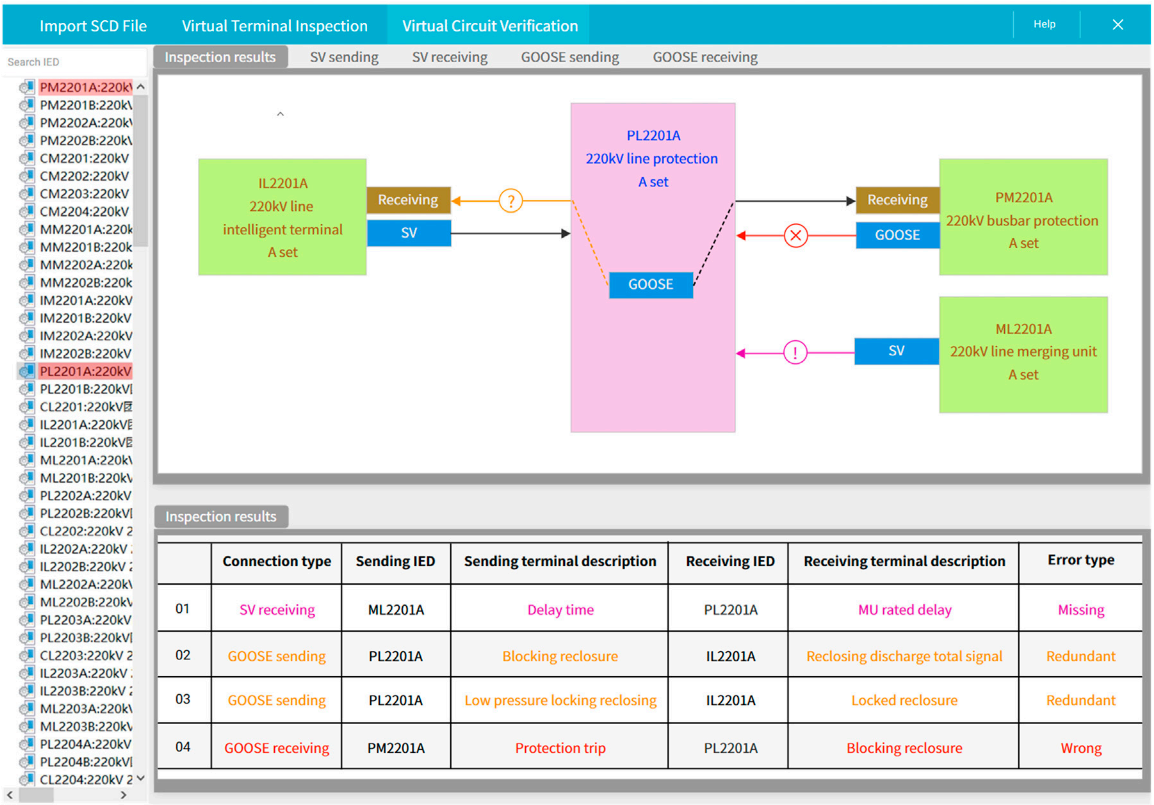


FIGURE 10
Accuracy and F1-score with different number of GRU layers.



	Connection type	Sending IED	Sending terminal description	Receiving IED	Receiving terminal description	Error type
01	SV receiving	ML2201A	Delay time	PL2201A	MU rated delay	Missing
02	GOOSE sending	PL2201A	Blocking reclosure	IL2201A	Reclosing discharge total signal	Redundant
03	GOOSE sending	PL2201A	Low pressure locking reclosing	IL2201A	Locked reclosure	Redundant
04	GOOSE receiving	PM2201A	Protection trip	PL2201A	Blocking reclosure	Wrong

FIGURE 11
Verification results of virtual circuit verification system.

TABLE 5 Comparison of verification results.

Verification method	Number of virtual circuits	Accuracy (%)	Time (min)
Manual verification	6000	92	160
Our model	6000	99	3

layers as [1, 2, 3, 4, 5], and testing and analyzing them on the validation set. The specific results are shown in Figure 10.

From Figure 10, it can be observed that when the number of GRU layers is set to 2, the model performs better compared to other values. This is primarily because when the number of GRU layers is 1, the model can only learn information in one direction, resulting in poor fitting to the dataset and inferior performance in extracting information compared to multi-layer GRU. However, when the number of layers increases to 2, GRU can learn information in two directions, enabling better learning of forward and backward contextual information. This facilitates capturing deeper relationships among hidden states, enriching the textual representation of vocabulary, and obtaining better text representation. However, due to the existence of GRU, both memory and time overheads increase. Therefore, as the number of GRU layers continues to increase, the time overhead increases with the increase in the number of neurons in GRU, and memory overhead also becomes significant. Moreover, the problem of vanishing gradients between layers becomes more apparent, leading to issues with data generalization and a higher likelihood of overfitting. Hence, this study sets the number of GRU layers to two to better learn information from various aspects and achieve better virtual terminal matching results.

4.4.4 Engineering application

The SCD editing tool modifies the SCD file in the sample to be verified. This process manually sets the error virtual circuits. A virtual circuit verification system based on the established knowledge graph and virtual circuit verification process, the verification results are shown in Figure 11. When a virtual circuit does not exist in the verification template based on the SCD file that passed the verification, the program automatically identifies the newly added virtual circuit and marks it as "!" as a reminder. When a virtual circuit is missing, the program automatically identifies the missing virtual circuit and marks it "?" as a reminder. The standard terminal library and virtual circuit verification template file based on the proposed method automatically verify the virtual circuit of the SCD file. The verification results were correct, showing the effectiveness of the proposed method.

In order to demonstrate the feasibility of system application, the efficiency and accuracy of intelligent verification and verification designed in the article were compared with manual verification by selecting the same number of intelligent substation configuration file verification tasks. The comparison results are shown in Table 5. It can be intuitively observed that compared with manual verification, the application of automatic verification technology for virtual circuit verification can help improve the efficiency and accuracy of intelligent substation virtual circuit verification.

5 Conclusion

In addressing the complexities associated with the verification of virtual circuits in intelligent substations, this study introduces a novel method that synergizes the strengths of knowledge graphs and deep learning. Through this fusion, we not only enhance the accuracy of virtual circuit verification but also set a new benchmark that surpasses the capabilities of traditional manual inspections and existing automated solutions. Our approach, characterized by its innovative integration of a Siamese neural network with a multi-head attention mechanism, demonstrates robust performance in the context of virtual terminal matching. Additionally, the inclusion of virtual terminal address string verification further enhances the accuracy of virtual circuit verification, presenting a new method for the verification of virtual loops in intelligent substations.

Future research will explore the method's adaptability to real-time configuration changes and potential integration with existing substation management systems, aiming to provide a more cohesive and efficient operational framework for intelligent substations and the broader power system. In addition, as intelligent substations continue to evolve, incorporating advancements such as IoT devices and advanced communication protocols, future research will focus on adapting our verification method to these new technologies. This includes exploring how to effectively process and integrate real-time data from a variety of sources to continuously update and refine the verification process.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

HC: Conceptualization, Writing–original draft. YZ: Methodology, Writing–original draft. YG: Methodology, Writing–original draft. JS: Methodology, Software, Writing–original draft. CT: Supervision, Validation, Writing–review and editing. XR: Supervision, Validation, Writing–review and editing. HC: Data curation, Writing–review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This work was supported by the State Grid Jiangsu Electric Power Co., Ltd. Incubation Project (JF2023011).

Conflict of interest

Authors HC, YG, CT, and XR were employed by State Grid Jiangsu Electrical Power Company. Authors YZ and HC were employed by State Grid Nanjing Power Supply Company. Author JS was employed by State Grid Suzhou Power Supply Company.

The authors declare that this study received funding from State Grid Jiangsu Electric Power Co., Ltd. The funder had the following involvement in the study: during the manuscript writing phase, they provided suggestions on the organization of the article. After the manuscript was written, they provided constructive comments on the way the images were presented. They were also involved in the revision of the manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Amir, A., Charalampopoulos, P., Pissis, S. P., and Radoszewski, J. (2020). Dynamic and internal longest common substring. *Algorithmica* 82, 3707–3743. doi:10.1007/s00453-020-00744-0

Bieder, F., Sandkühler, R., and Cattin, P. C. (2021) *Comparison of methods generalizing max-and average-pooling*.

Cao, Y. K., and Zhao, T. (2018). "Text classification based on TextCNN for power grid user fault repairing information," in *5th international conference on systems and informatics*, 1182–1187. doi:10.1109/ICSAI.2018.8599486

Chen, Q., Zhu, X. D., Ling, Z. H., Wei, S., Jiang, H., and Inkpen, D. (2016) *Enhanced LSTM for natural language inference*.

Chen, X., Zhang, C., Liu, Q. K., Peng, Y., Zhou, D. M., and Zhen, J. M. (2021). Automatic configuration method of intelligent recorder based on deep semantic learning. *Power Syst. Prot. Control* 49 (02), 179–187. doi:10.19783/j.cnki.pspc.200367

Chen, X. J., Jia, S. B., and Xiang, Y. (2020). A review: knowledge reasoning over knowledge graph. *Expert Syst. Appl.* 141, 112948. doi:10.1016/j.eswa.2019.112948

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014) *Learning phrase representations using RNN encoder-decoder for statistical machine translation*.

Cui, Y. G., Zhang, F., Liu, H. Y., Ji, J., Liu, X. G., and Zhao, J. (2018). "Research on secondary circuit identification technology and condition-based maintenance mode of intelligent substation," in *2018 international conference on power system technology*, 3716–3722. doi:10.1109/POWERCON.2018.8602073

Fan, W. D., Feng, X. W., Dong, J. X., Hu, Y., Wang, W. Q., and Gao, X. (2020). Automatic matching method of a virtual terminal in intelligent substation based on semantic similarity of historical data. *Power Syst. Prot. Control* 48, 179–186. doi:10.19783/j.cnki.pspc.191272

Guan, N. N., Song, D. D., and Liao, L. J. (2019). Knowledge graph embedding with concepts. *Knowledge-Based Syst.* 164, 38–44. doi:10.1016/j.knosys.2018.10.008

Guo, Q., Huang, J., Xiong, N., and Wang, P. (2019). MS-pointer network: abstractive text summary based on multi-head self-attention. *IEEE Access* 7, 138603–138613. doi:10.1109/ACCESS.2019.2941964

Hao, X. G., Zhao, Y. H., Yin, X. G., Luo, P., Yang, J. C., and Mao, Y. R. (2020). Intelligent substation virtual circuit check based on the intermediate model file. *J. Electr. Power Sci. Technol.* 35, 132–137. doi:10.19781/j.issn.16739140.2020.05.018

Huang, Q., Jing, S., Li, J., Li, J., Cai, D. S., Wu, J., et al. (2017). Smart substation: state of the art and future development. *IEEE Trans. Power Deliv.* 32, 1098–1105. doi:10.1109/TPWRD.2016.2598572

Li, C., and Wang, B. (2023). A knowledge graph method towards power system fault diagnosis and classification. *Electronics* 12, 4808. doi:10.3390/electronics12234808

Li, S., Zhao, Z., Hu, R. R., Li, W. S., Liu, T., and Du, X. Y. (2018) *Analogical reasoning on Chinese morphological and semantic relations*.

Li, Z. G., Chen, H., and Chen, H. Y. (2021). Biomedical text similarity evaluation using attention mechanism and Siamese neural network. *IEEE Access* 9, 105002–105011. doi:10.1109/ACCESS.2021.3099021

Liang, G., On, B. W., Jeong, D., Kim, H. C., and Choi, G. S. (2018). Automated essay scoring: a siamese bidirectional LSTM neural network architecture. *Symmetry* 10 (12), 682. doi:10.3390/sym10120682

Liu, R., Fu, R., Xu, K., Shi, X. Z., and Ren, X. N. (2023). A review of knowledge graph-based reasoning technology in the operation of power systems. *Appl. Sci.* 13, 4357. doi:10.3390/app13074357

Oliveira, B. A. S., De Faria Neto, A. P., Fernandino, R. M. A., Carvalho, R. F., Fernandes, A. L., and Guimarães, F. G. (2021). Automated monitoring of construction sites of electric power substations using deep learning. *IEEE Access* 9, 19195–19207. doi:10.1109/ACCESS.2021.3054468

Ren, J. B., Li, T. C., Gen, S. B., Liu, Q. Q., He, Y. K., Wang, Z. H., et al. (2020). An automatic mapping method of intelligent recorder configuration datasets based on Chinese semantic deep learning. *IEEE Access* 8, 168186–168195. doi:10.1109/ACCESS.2020.3024060

Reyad, M., Sarhan, A. M., and Arafa, M. (2023). A modified Adam algorithm for deep neural network optimization. *Neural Comput. Appl.* 35 (23), 17095–17112. doi:10.1007/s00521-023-08568-z

Selim Ustun, T., and Suhail Hussain, M. (2020). An improved security scheme for IEC 61850 mms messages in intelligent substation communication networks. *J. Mod. Power Syst. Clean* 8, 591–595. doi:10.35833/MPCE.2019.000104

Song, Q. P., Sheng, W. X., Kou, L. F., Zhao, D. B., Wu, Z. P., Fang, H. F., et al. (2016). Smart substation integration technology and its application in distribution power grid. *CSEE J. Power Energy* 2, 31–36. doi:10.17775/CSEEJPES.2016.00046

Tian, J. P., Song, H., Sheng, G. H., and Jiang, X. C. (2022). An event knowledge graph system for the operation and maintenance of power equipment. *IET Generation, Transm. Distribution* 16, 4291–4303. doi:10.1049/gtd2.12598

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., et al. (2017). Attention is all you need. *Adv. neural Inf. Process. Syst.* 30, 5999–6009. doi:10.48550/arXiv.1706.03762

Wang, Q., Mao, Z. D., Wang, B., and Guo, L. (2017). Knowledge graph embedding: a survey of approaches and applications. *IEEE Trans. Knowl. Data En.* 29 (12), 2724–2743. doi:10.1109/TKDE.2017.2754499

Wang, W. Q., Hu, Y., Zhao, N., and Wu, C. Y. (2018). Automatic connection method of virtual terminators based on optimization model of distance weight vectors. *Power Syst. Technol.* 42 (01), 346–352. doi:10.13335/j.1000-3673.pst.2017.1746

Xie, J., Chen, B., Gu, X. L., Liang, F. M., and Xu, X. Y. (2019). Self-attention-based BiLSTM model for short text fine-grained sentiment classification. *IEEE Access* 7, 180558–180570. doi:10.1109/ACCESS.2019.2957510

Zhang, X. L., Liu, H. H., Li, J. Q., Ai, J. Y., Tang, Y., and Wang, J. Y. (2015). Automatic test platform in smart substation for relay protection. *Automation Electr. Power Syst.* 39 (18), 91–96. doi:10.7500/AEPS20141103006

Zhao, Z. C., Lu, D. H., Liu, S. J., Chen, S., and Ding, J. W. (2023). Research on automatic correlation technology of virtual circuit in secondary system of intelligent substation. *J. Phys. Conf. Ser.* 2479, 012068. doi:10.1088/1742-6596/2479/1/012068