ASSPM: A Deep Address Parsing Model Integrating Address Spatial Semantics for Traffic Accident Handling

Kun Li *, Feng Wang , Guangming Ling

Posted Date: 27 June 2024

doi: 10.20944/preprints202406.1885.v1

Keywords: Traffic accident handling; Address parsing; Deep learning model; BERT; Transformer; Address spatial semantics

Article

ASSPM: A Deep Address Parsing Model Integrating Address Spatial Semantics for Traffic Accident Handling

Kun Li [1,*], Feng Wang [1] and Guangming Ling [2]

[1] Intelligent Traffic Police Affairs Laboratory, Henan Police College, China
[2] School of Computer, Henan University of Engineering, China
* Correspondence: lk@hnp.edu.cn; Tel.: +86-18838935650

Abstract: To enhance the accuracy and efficiency of address parsing in traffic accident handling, this study designs a deep address parsing model based on BERT, named ASSPM. This model effectively addresses the limitations of existing deep learning models in Chinese address parsing tasks by integrating address spatial semantics. First, a lexicon-enhancement method based on hybrid representation is employed, transforming the spatial semantics extracted by the Address Spatial Semantics Extractor (ASSE) into a symbolic semantic space that can be integrated with the feature vector space of the deep model. This achieves effective fusion of spatial semantics and the deep model at the model level. Second, a lexicon adapter enhanced with spatial semantics is utilized to integrate spatial semantics into the Transformer layers of the BERT model, achieving semantic fusion at the BERT level. Finally, comparative experiments on a custom address dataset and a public Chinese address dataset validate the proposed method's superior performance over other baseline methods. This research provides robust support for the application of deep address parsing models integrating address spatial semantics in traffic accident handling, promising to improve the efficiency and accuracy of traffic accident management.

Keywords: traffic accident handling; address parsing; deep learning model; bert; transformer; address spatial semantics

1. Introduction

With the acceleration of urbanization and the increasing complexity of traffic systems, the issue of address parsing in traffic accident handling has become increasingly prominent [1]. Accurate and efficient address parsing not only enhances the efficiency of traffic accident management but also provides crucial support for accident analysis and prevention. However, the diversity and complexity of Chinese addresses pose significant challenges to traditional parsing methods [2,3]. In recent years, deep learning technology has achieved remarkable results in the field of natural language processing [4,5], offering new possibilities for addressing this problem.

Deep learning models, particularly pre-trained models such as BERT [6], have demonstrated remarkable performance across various text processing tasks. However, these models still exhibit limitations when handling tasks with specific domain characteristics, such as Chinese address parsing. The primary issue lies in their inability to fully leverage the spatial semantic information embedded in addresses, which is crucial for accurate comprehension and parsing. Therefore, effectively integrating address spatial semantics with deep learning models becomes a key challenge to enhance Chinese address parsing performance.

To achieve effective integration of address spatial semantics with deep learning in the Chinese address parsing task [7], this paper proposes an innovative approach by designing a BERT-based deep address parsing model (ASSPM). This model addresses the limitations of existing deep learning models in Chinese address parsing tasks through the fusion of address spatial semantics. Our approach encompasses two key innovations:

First, we introduce a lexicon-enhancement method based on hybrid representation. This method leverages the robust semantic extraction capabilities of the Address Spatial Semantics Extractor (ASSE), transforming the extracted spatial semantics into a symbolic semantic space. This transformation

 doi:10.20944/preprints202406.1885.v1

not only preserves data distribution characteristics but also standardizes the features from different scales into a unified feature space, providing the deep model with richer and more meaningful inputs. By weighted aggregation and concatenation, these features are embedded into character-level representations, achieving effective fusion of spatial semantics and the deep model at the model level.

Second, we design a spatial semantics-enhanced lexicon adapter. This innovative component is inserted between the Transformer layers of the BERT model, allowing spatial semantic information to be deeply integrated into the BERT model's lower layers. This approach realizes the effective incorporation of prior knowledge into the deep model at the BERT internal level, further enhancing the model's understanding of address spatial semantics.

Through this dual-layer fusion strategy, we construct a deep parsing model that can fully exploit and effectively utilize address spatial semantics. This model not only introduces theoretical innovation but also demonstrates excellent performance in practical applications, providing robust support for improving the efficiency and accuracy of traffic accident handling.

The primary contributions of this research can be summarized as follows:

(1) To address the issue of BERT's inability to effectively utilize address spatial semantics at the embedding layer, we propose a lexicon-enhancement method based on hybrid representation. This method transforms the address spatial semantics extracted by ASSE into a symbolic semantic space, achieving the fusion of address spatial semantics with the BERT-based deep parsing model at the model level.

(2) To tackle the problem of insufficient integration of spatial semantics within the BERT model, we introduce a spatial semantics-enhanced lexicon adapter. This adapter incorporates address spatial semantics into the Transformer layers of the BERT model, realizing effective fusion of spatial semantics within the BERT layers and the BERT-based deep parsing model.

(3) We design a BERT-based deep model that integrates address spatial semantics (Address Spatial Semantics Fusion Deep Parsing Model, ASSPM), demonstrating its outstanding performance in high-resource scenarios. This model excels in Chinese address parsing tasks, promising to enhance the performance of address parsing models in practical applications, especially in scenarios requiring substantial data and hardware resources. This provides a robust solution for address parsing tasks.

The structure of this paper is organized as follows: Section 2 reviews related work, focusing on methods combining deep learning with prior knowledge; Section 3 details the architecture and fusion strategy of the proposed ASSPM model; Section 4 presents the experimental design and results analysis on custom and public Chinese address datasets; and Section 5 concludes the study and discusses future research directions.

2. Related Work

The integration of address spatial semantics is crucial for the application of deep learning models in traffic accident handling. In recent years, researchers have extensively explored methods to combine deep learning with prior knowledge to enhance model performance and application value. The work of Weinzierl et al. [8] demonstrated the effectiveness of embedding Unified Medical Language System knowledge in biomedical text relation extraction. By designing a Knowledge Embedding Encoder (KEE), they developed a Knowledge-Embedded Relation Extraction (REKE) system, providing new perspectives for the development of relation extraction technology.

Methods for incorporating prior knowledge into deep learning models include preprocessing techniques, integration into the network structure, unrolling techniques, and knowledge graphs [9]. Enhancing the representation of the embedding layer is crucial for this integration. Embedding layer representations can be categorized into word-level [10], character-level [11], and hybrid representations [12–15]. For Chinese tasks, hybrid representations are particularly important, including pre-trained methods such as BERT [6] and lattice LSTM methods [14,16] that integrate latent lexical information.

 doi:10.20944/preprints202406.1885.v1

Dynamic frameworks and adaptive encoding are two main methods for integrating lexical information. The dynamic framework, represented by Lattice LSTM [14,16], integrates external lexical information through a meticulously designed LSTM structure. The LR-CNN model proposed by Gui et al. [17] replaces RNN with CNN, addressing lexical conflicts through an attention mechanism. The character-based collaborative graph network proposed by Sui et al. [18] integrates lexical information into each character, enabling interaction between characters and all matched lexical information via a GAN graph attention network. The FLAT model by Li et al. [19] replaces LSTM with Transformer and integrates lexical information through positional encoding, enhancing lexical augmentation effects. Adaptive encoding methods like Simple-Lexicon [15] obtain lexical weights through lexical frequency and combine them with pre-trained models like BERT to improve efficiency and performance.

Recent research focuses on how to integrate lexical information into deep models, including the fusion of dictionary information with pre-trained models, learning Chinese lexical semantics, and understanding sentence structure. The LEBERT model proposed by Liu et al. [20] deeply integrates external dictionary information into the BERT model through a lexicon adapter layer. The BERT-wwm method proposed by Cui et al. [21] employs a whole word masking strategy to better learn Chinese lexical semantics. The BABERT model by Jiang et al. [4] uses unsupervised statistical information to compute boundary-aware representations, capturing sentence structure information and directly injecting it into BERT weights during pre-training, enhancing the model's understanding of sentence structure.

In summary, the application and research of deep learning models integrating address spatial semantics in traffic accident handling involve the embedding of prior knowledge, the fusion of lexical information, and the learning of sentence structure. These research outcomes provide significant theoretical support and practical guidance for this study.

3. Methodology

3.1. Model Framework

This paper proposes a fusion framework of spatial semantics and deep models, thereby achieving integrated deep parsing of Chinese addresses with spatial semantics. This model, named Address Spatial Semantics Fusion Deep Parsing Model (ASSPM), integrates spatial semantic information extracted by the Address Spatial Semantics Extractor (ASSE). The framework is illustrated in Figure 1.

3.2. ASSE

As depicted in Figure 1, the Address Spatial Semantics Extractor (ASSE) constitutes a core component of the ASSPM. ASSE quantifies boundary features using regular matching techniques and utilizes a directed graph to represent spatial dependency relationships among address elements. Furthermore, ASSE employs the XGBoost(eXtreme Gradient Boosting[22]) algorithm to compute confidence scores for each address element, thereby constructing a semantic extraction framework based on a directed computational graph. The output is the Address Element Boundary Matrix (EBM), equipped with multidimensional evaluation mechanisms that leverage spatial semantic information effectively. This enables efficient extraction and quantitative assessment of address semantics.

To accurately define and identify address elements, this paper rigorously defines 28 types of address elements and constructs an Address Element Boundary Rule Table (AEBRT) based on these types. AEBRT includes a set of regular expressions representing specific boundary features of address elements, along with corresponding indices and weights. This allows effective representation of boundary features for various address elements. Across specific datasets, all AEBRT collectively form the Address Element Boundary Rule Library (AEBRL), providing robust rule support to ASSE and ensuring accuracy and adaptability during the address semantic extraction process.

During semantic extraction, the input Chinese address is first segmented into address token sequencesusing an address segmenter (AS). Subsequently, based on identified address element types, the address graph(AG) predicts the next potential address element type. The address regularizer(AR)

 doi:10.20944/preprints202406.1885.v1

then combines the current token sequence with the predicted address element type to determine the corresponding state sequence, utilizing the XGBoost algorithm to compute confidence scores for each state. The flow control unit in the spatial semantic extraction algorithm determines the extraction process based on these confidence results. Finally, the extracted results are recorded in the EBM.

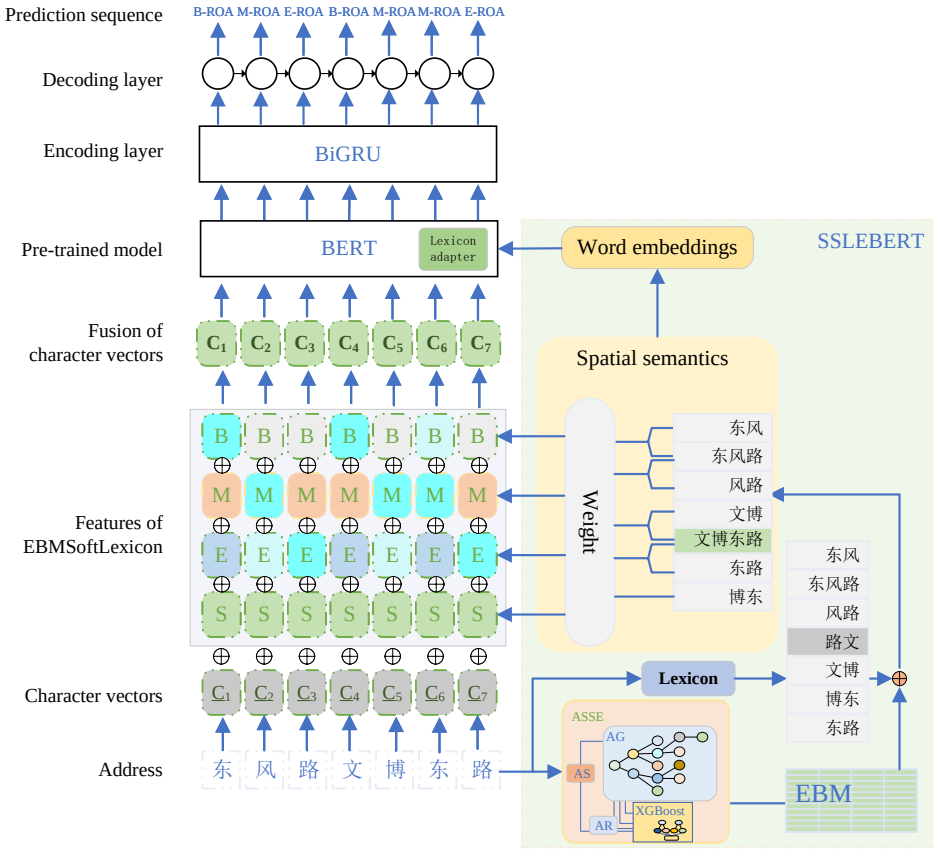

Figure 1. ASSPM. Inputting “东风路文博东路”(Dongfeng Road Wenbo East Road), first, the Address Spatial Semantics Extractor (ASSE) extracts spatial semantic information and generates the Address Element Boundary Matrix (EBM), from which the address elements corresponding to the address text are obtained. Next, using a dictionary (pre-trained word embeddings [23]), all words in the address text are matched to obtain a vocabulary set (including seven elements: “东风”(Dongfeng),“东风路”(Dongfeng Road),“风路”(Feng Road),“路文”(Road Wen),“文博”(Wenbo),“博东”(Bo East) and “东路”(East Road)). The address element set and vocabulary set are then merged, removing words (“路文”(Road Wen)) that do not match in the address element set. Subsequently, two levels of word fusion are performed: one based on EBMSoftLexicon for spatial semantic representation fusion, and the other between spatial semantics and the Transformer layer of the BERT model, thereby constructing the Spatial Semantic Lexicon Enhanced BERT preprocessing model (SSLEBERT). This approach helps integrate and enhance spatial semantic information of address texts. Finally, a Bidirectional Gated Recurrent Unit (BiGRU) is used as the encoding layer to encode the fused spatial semantic representation output by BERT. Simultaneously, a Conditional Random Field (CRF) is employed as the decoding layer to obtain the parsing result of the input address (“东风路文博东路”(Dongfeng Road Wenbo East Road)).

3.3. Spatial Semantic Representation Fusion Module

This section discusses the representation fusion depicted in Figure 1, specifically the fusion of spatial semantics with character-level representations. Essentially, this fusion process involves converting boundary features and spatial distribution patterns into symbolic semantic space through regular expressions and directed computational graphs, thereby effectively integrating with vector feature space at the character level.

 doi:10.20944/preprints202406.1885.v1

(1) Character Representation Layer

For Chinese parsing models based on character-level representations, the input sentence is treated as a sequence of characters, denoted as $s = \{c_1, c_2, \cdots, c_n\} \in \mathcal{V}_c$, where $\mathcal{V}_c$ is the character vocabulary [15]. Each character c_i is represented as a vector, as shown in Equation (1):

$$\boldsymbol{x}_i^c = \boldsymbol{e}^c(c_i), \tag{1}$$

where $\boldsymbol{e}^c$ denotes the character embedding lookup table.

Zhang and Yang [14] demonstrated that character bigrams can effectively enhance character representations. Therefore, character bigrams are constructed and concatenated with the vector representation of character c_i, as shown in Equation (2):

$$\boldsymbol{x}_i^c = \left[\boldsymbol{e}^c(c_i); \boldsymbol{e}^b(c_i, c_{i+1})\right], \tag{2}$$

where $\boldsymbol{e}^b$ represents the bigram character embedding lookup table. This results in the basic character vector representation. Next, methods to enhance it through the incorporation of dictionary information are discussed.

(2) Incorporating lexical information

Below we discuss methods to integrate lexical information into $\boldsymbol{x}_i$. To address the limitation of character-based NER models in utilizing lexical information, Ma et al. [15] proposed two approaches: ExSoftword and SoftLexicon. Before delving into these methods, let's first discuss the representation of words: for any input character sequence $s = \{c_1, c_2, \ldots, c_n\}$, introduce the symbol $w_{i,j}$ to denote the subsequence $\{c_i, c_{i+1}, \ldots, c_j\}$. Next, we will discuss three methods for integrating lexical information: ExSoftword, SoftLexicon, and EBMSoftLexicon.

ExSoftword: As an extension of Softword [24,25], ExSoftword not only retains one segmentation result for each character after segmentation but obtains all possible segmentation results from the lexicon, as shown in Equation (3):

$$\boldsymbol{x}_j^c \leftarrow \left[\boldsymbol{x}_j^c; \boldsymbol{e}^{seg}\left(segs\left(c_j\right)\right)\right], \tag{3}$$

where $segs\left(c_j\right)$ represents all classification labels related to c_j. For example, for the character "江"(River) in Figure 1, which appears as both a middle and an end character, $segs(江) = \{M, E\}$. $\boldsymbol{e}^{seg}\left(segs\left(c_j\right)\right)$ is a 5-dimensional multi-hot vector, with each dimension corresponding to {B, M, E, S, O}.

ExSoftword faces two challenges [15]: 1) It cannot utilize pre-trained word models. 2) It suffers from missing matching information, leading to multiple matching results.

SoftLexicon: To address the aforementioned issues, SoftLexicon [15] offers a solution that involves three primary steps:

Firstly, in the initial step, the matched vocabulary is categorized. To retain segmentation information, each character matched to a word is classified into {B, M, E, S}. For each character c_i, the four sets are defined as follows in Equation (4):

$$\begin{aligned} B(c_i) &= \left\{w_{i,k}, \forall w_{i,k} \in L, i < k \leq n\right\}, \\ M(c_i) &= \left\{w_{j,k}, \forall w_{j,k} \in L, 1 \leq j < i < k \leq n\right\}, \\ E(c_i) &= \left\{w_{j,i}, \forall w_{j,i} \in L, 1 \leq j < i\right\}, \\ S(c_i) &= \left\{c_i, \exists c_i \in L\right\}, \end{aligned} \tag{4}$$

 doi:10.20944/preprints202406.1885.v1

where L denotes the dictionary used in this study. If no vocabulary information matches, it is replaced with "None". This approach embeds words while mitigating information loss.

Secondly, the sets {B, M, E, S} of vocabulary vectors are compressed into fixed-dimensional vectors. After obtaining the sets B, M, E, S for each character, they are compressed into fixed-dimensional vectors in two ways. The first approach is to directly average them, as shown in Equation (5):

$$v^u(U) = \frac{1}{|U|} \sum_{w \in U} e^w(w), \tag{5}$$

where U represents the set of vocabulary, i.e., the set of vocabulary, and $e^w(w)$ represents the word embedding.

At the same time, in order to maintain computational efficiency, dynamic weighted algorithms such as attention mechanisms have not been adopted. Ma et al. [15] suggested using frequency to represent the weight of each word. Since the frequency of a word can be obtained offline as a static value, this significantly improves the efficiency of calculating the weight of each word. The literature confirmed that the compression method implemented by Equation (5) is not ideal.

Next, the second approach is discussed. Specifically, let $z(w)$ represent the frequency at which the word w appears in the dictionary. Thus, the weighted representation of the set U can be obtained, as shown in Equation (6):

$$v^u(U) = \frac{4}{Z} z(w) e^w(w), \tag{6}$$

where $Z = \sum_{w \in B \cup M \cup E \cup S} z(w)$. If w is covered by another subsequence matching the dictionary, the frequency of w will not increase. This prevents the problem where the frequency of shorter words is always smaller than that of longer words covering them.

Finally, the third step involves merging character representations. Once the fixed-dimensional representations of the four vocabulary sets are obtained, they can be combined into a fixed-dimensional feature, as shown in Equation (7):

$$e^u(B, M, E, S) = [v^u(B); v^u(M); v^u(E); v^u(S)], \tag{7}$$

where v^u is the weighted function provided by Equation (6). Thus, Equation (7) yields the SoftLexicon character representation, as shown in Equation (8):

$$x^c \leftarrow [x^c; e^u(B, M, E, S)], \tag{8}$$

EBMSoftLexicon: While SoftLexicon has demonstrated significant performance, particularly showcasing its potential in low-resource scenarios, for Chinese address parsing tasks, this approach introduces some unnecessary lexical information. For instance, in the context of “东风路”(Dongfeng Road) depicted in Figure 1, the lexical information for “东风”(Dongfeng) could be filtered out entirely due to the prominent tail feature of the character “路”(road). Similarly, “文博”(Wenbo) and “博东”(Bo East) serve as examples of irrelevant “interference” information that does not need consideration. To address the challenges identified, an enhanced method called EBMSoftLexicon is proposed to integrate lexical information more effectively, particularly for Chinese address parsing tasks. This method combines the SoftLexicon approach with Address Element Boundary Matrix (EBM) information derived from ASSE. This integration aims to refine or enhance SoftLexicon by incorporating boundary information, confidence scores, and evaluation metrics from EBM.

 doi:10.20944/preprints202406.1885.v1

Specifically, the approach involves calculating a weighted vector representation $E(w)$ for each vocabulary word w, as defined in Equation (9):

$$E(w) = \sum_{w \in \mathbf{ADD}} score^w \cdot C^w, \tag{9}$$

where $score^w$ and C^w are determined by Equations (10) and (11), respectively.

$$score = \frac{\log(score)}{\log(MAX_{score})}, \tag{10}$$

$$C = XGBoostCC(\vec{x}; P) = xgb.predict(D(\vec{x}), xgb(P)), \tag{11}$$

In equation (11), $\vec{x}$ represents an 8-dimensional input vector, specifically the address state, composed of address element type, transition probability, length information, matching start and end positions, regular expression score, matching length, and a numeric indicator (1 for Chinese and Arabic numerals, 0 otherwise). P denotes the parameters of the XGBoost, with the experimental section detailing the configuration of these parameters to obtain P. $D(\vec{x})$ denotes the function that converts the input vector $\vec{x}$ into a DMatrix data structure. $xgb(P)$ refers to the XGBoost trained using the parameters P, and $xgb.predict$ is the function used to make predictions with the trained model.

ADD denotes the annotated dataset. Correspondingly, let $E = \sum_{w \in B \cup M \cup E \cup S} E(w)$, which allows us to obtain the weighted representation of the word set U in EBMSoftLexicon, as shown in equation (12):

$$V^u(U) = \begin{cases} \frac{4}{E} E(w) e^w(w), & \text{if } \exists(EBM^w), \\ v^u(U), & \text{else.} \end{cases} \tag{12}$$

Here, $\exists(EBM^w)$ indicates that the EBM score surpasses the threshold, allowing for enhanced embedding using $E(w)$ and $e^w(w)$. If the condition is not met (else case), or for non-Chinese address data where EBM information isn't applicable, a fallback to SoftLexicon embedding $v^u(U)$ occurs.

EBMSoftLexicon is obtained offline through ASSE. This method first uses ASSE to extract all Chinese addresses' EBM from the annotated dataset. This extraction differs slightly from the original data extraction because basic information can be obtained directly from the annotation information, such as the matching targets and address element boundary types in equation (13). This significantly reduces the complexity of solving and allows evaluation information to be obtained through the evaluation mechanism in the extractor. Then, equation (9) is used to obtain the weights of all vocabulary in the current dataset. It is important to note that vocabulary exists in two forms: complete address elements represented by EBM column vectors, and boundary information contained in EBM. In cases where both forms exist, preference is given to the former due to its inclusion of the latter.

$$E_i = P(\Psi(add, T_j, AEBRL)), \tag{13}$$

Here, add represents the address text, T_j denotes a specific address element type, and Ψ extracts boundary information using T_j and $AEBRL$. The function P combines the results of Ψ into E_i, which populates the first seven dimensions of $E^k_{i,j} = (i, j, k, [k], W_k, a, b, C)^T$. The Ψ function is detailed in Algorithm 1:

 doi:10.20944/preprints202406.1885.v1

Algorithm 1 Algorithm for Regular Matching of Address Element Boundaries

Require: add, AEBRL, T_j
Ensure: E_i
$rstart, rend \leftarrow$ INITIALIZATION(T_j) ▷ Initialize boundary information based on type
$CurAEBRT \leftarrow$ GETAEBRL$(AEBRL, T_j)$ ▷ Retrieve the Address Element Boundary Rule Table ($AEBRT$) corresponding to T_j in $AEBRL$
for each $key \in CurAEBRT$ **do**
    $tstart \leftarrow 0$
    while $tstart \geq 0$ **do**
        $tstart, tend, c \leftarrow$ GETSEBYRE(add, key) ▷ Regular expression matching and record current weight
        if $tend \geq 0$ **then**
            $rstart, tend, add, c \leftarrow$ BETTERRECORD$(tstart, tend, add, c)$ ▷ Update best result based on weight
        end if
    end while
end for
if $rend ==$ best **then**
    $start, end \leftarrow -1, -1$
else
    $start, end \leftarrow rstart, rend$
end if
$E_i \leftarrow$ CONSTRUCT$(start, end, add, AEBRL, T_j, c)$ ▷ Retrieve E_i information

Finally, substituting equation (12) into equations (7) and (8), the EBMSoftLexicon character representation is obtained as shown in equation (14):

$$x^c \leftarrow [x^c; [V^u(B); V^u(M); V^u(E); V^u(S)]], \tag{14}$$

In summary, EBMSoftLexicon not only avoids introducing unnecessary or "noise" information directly from dictionary lookups, but also leverages boundary and spatial distribution information effectively. By obtaining vocabulary information from ASSE, which includes evaluation scores and confidence levels, when EBM is not obtained or the obtained EBM has insufficient scores (corresponding to the *else* case in equation (12)), the solution follows equation (6). Therefore, the EBMSoftLexicon represented by equation (14) is compatible with SoftLexicon, contributing to enhanced model stability. This approach achieves a blended representation of spatial semantics and deep parsing models through enhanced embedding layers.

The proposed model, compared to traditional approaches, significantly reduces the reliance on specialized data and domain knowledge. It also addresses the fundamental cost issues associated with manual annotation in deep learning models tackling this problem. By integrating spatial semantics and deep models in a novel manner, the model enhances the performance of deep learning models. Importantly, through modifications to the Address Element Boundary Rules Library (AEBRL), it enables corrections to anomalous results, thereby partially mitigating the interpretability issues inherent in deep models at the current research stage.

3.4. Spatial Semantic Lexicon Enhanced BERT Module

In the previous section on representation fusion, spatial semantics were integrated into the embedding layer. To leverage the expressive power of BERT more effectively, this section proposes a Spatial Semantic Lexicon Enhanced BERT (SSLEBERT) preprocessing model that integrates spatial semantics directly into the BERT model, achieving superior fusion results.

Liu et al. [20] introduced Lexicon Enhanced BERT (LEBERT), which integrates lexicon features into BERT between the k-th and (k+1)-th Transformer layers using a lexicon adapter. LEBERT differs

 doi:10.20944/preprints202406.1885.v1

significantly from BERT in two key aspects: 1) It combines character and lexicon features as inputs, transforming Chinese sentences into sequences of character-word pairs; 2) LEBERT introduces a lexicon adapter between Transformer layers to effectively integrate lexicon knowledge into BERT.

This design allows LEBERT to fully exploit BERT's expressive capabilities by integrating external features at a lower level, thereby enhancing model performance. ASSE extracts spatial semantics from address texts, including lexicon features with evaluation mechanisms. These lexicon features can be integrated into BERT via a lexicon adapter between Transformer layers. This paper terms this enhanced BERT model with address spatial semantics as the Spatial Semantic Lexicon Enhanced BERT (SSLEBERT) model, depicted in Figure 2.

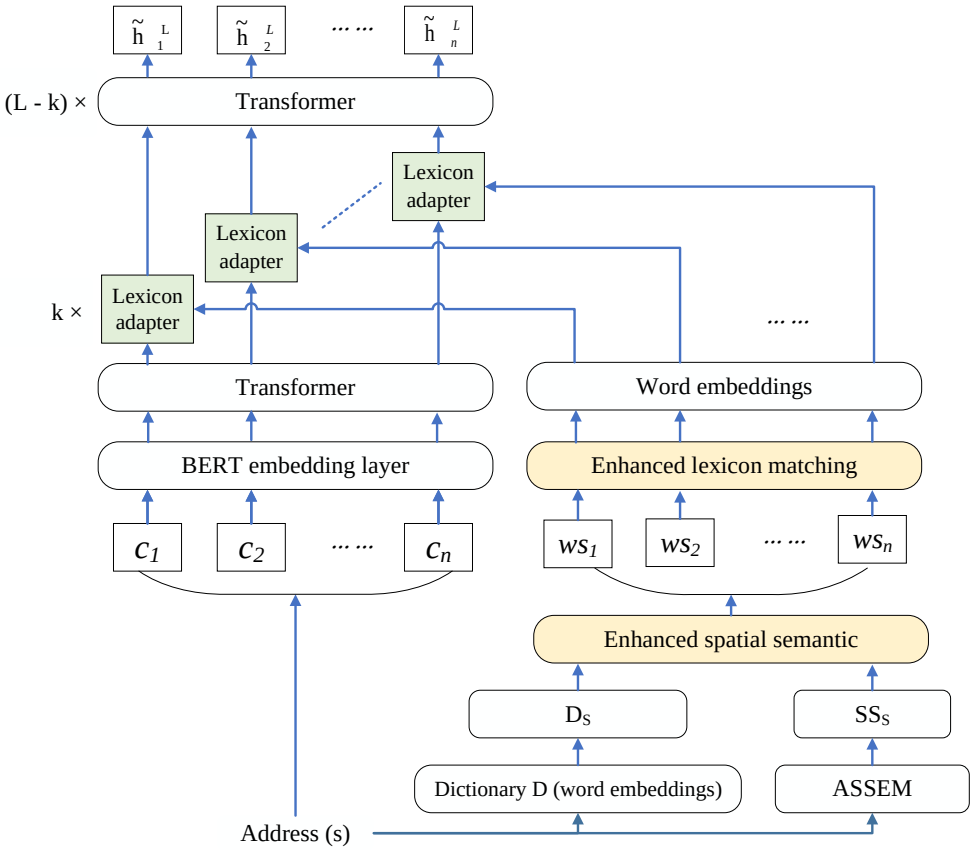

Figure 2. Architecture of SSLEBERT. Here, D_s represents the collection of all vocabulary obtained after dictionary matching, SS_s represents the vocabulary (i.e., address elements) obtained through the ASSE, c_i denotes the i-th Chinese character in the sentence, and ws_i denotes the matching word assigned to character c_i.

Next, we will discuss the enhanced spatial semantics of character-word pairs, lexicon adapter, and lexicon-enhanced BERT.

(1) Enhanced Spatial Semantics of Character-Word Pairs

A Chinese sentence is typically represented as a sequence of characters, which only captures character-level features. To incorporate lexical information, the character sequence is expanded into a sequence of character-word pairs. Given a Chinese dictionary $\mathbf{D}$ and a sentence $\mathbf{s_c} = \{c_1, c_2, ..., c_n\}$ consisting of n characters, potential words in the sentence, denoted as WS, are identified by matching the character sequence with $\mathbf{D}$. Specifically, a Trie tree is constructed based on $\mathbf{D}$, and all possible character subsequences in the sentence are matched against this Trie tree to obtain all potential words, denoted as D_s. To enhance spatial semantics representation, lexical elements from the dictionary and address features extracted by the ASSE are utilized. $\mathbf{SS_s}$ denotes the vocabulary sequence obtained by ASSE, referred to as spatial vocabulary sequence. For

 doi:10.20944/preprints202406.1885.v1

each address, each corresponding WS is sequentially searched within $\mathbf{SS_s}$ for substring matches. Words in WS that cannot be found in $\mathbf{SS_s}$ are removed, and finally, the union operation is applied between WS and $\mathbf{SS_s}$. The spatial semantics-enhanced character-word pairs corresponding to the addresses in Figure 1 are illustrated in Figure 3.

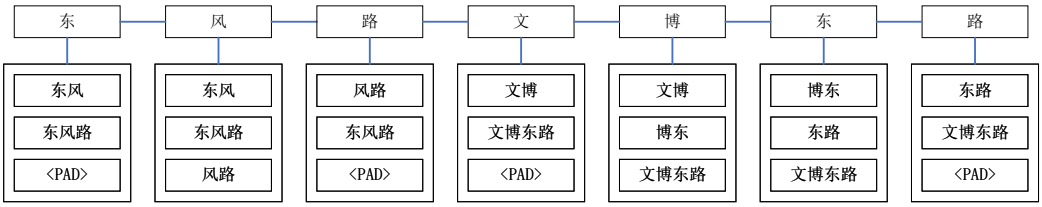

Figure 3. Illustration of Character-Word Pair Sequence. The sequence "东风路文博东路"(Dongfeng Road Wenbo East Road) can match seven different words through **D**: "东风"(Dongfeng),"东风路"(Dongfeng Road),"风路"(Feng Road),"路文"(Road Wen),"文博"(Wenbo),"博东"(Bo East) and "东路"(East Road). After ASSE processing, $\mathbf{SS_s}$ is {"东风路"(Dongfeng Road), "文博东路"(Wenbo East Road)}. Enhanced processing filters out vocabulary not found in address features (e.g., "路文"(Road Wen)), resulting in D_s containing seven words: "东风"(Dongfeng),"东风路"(Dongfeng Road),"风路"(Feng Road),"文博"(Wenbo),"博东"(Bo East),"东路"(East Road) and "文博东路"(Wenbo East Road). Subsequently, each word in D_s is mapped to its corresponding characters, transforming the Chinese sentence into a sequence of character-word pairs, denoted as $\mathbf{s_{cw}} = \{(c_1, ws_1), (c_2, ws_2), ..., (c_n, ws_n)\}$, where c_i represents the i-th character in the sentence and ws_i denotes the matched word assigned to c_i. <PAD> denotes padding, with each word corresponding to its included characters.

(2) Lexicon Adapter

Each position in every sentence contains two distinct types of information: character-level and word-level features. Consistent with existing hybrid model approaches, this study aims to effectively integrate lexicon features with BERT. The Lexicon Adapter (LA) [20], depicted in Figure 4, illustrates this integration process.

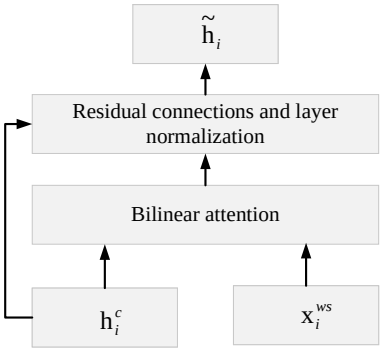

Figure 4. Structure diagram of the Lexicon Adapter. It accepts character vectors and matched word features as input. Through bilinear attention mechanism, it simultaneously focuses on characters and words, weighting lexicon features to combine them into a vector. This vector is then fused with character-level vectors and undergoes layer normalization, thereby embedding lexicon information directly into BERT.

The Lexicon Adapter accepts two inputs: character vectors and their corresponding word embeddings. At the i-th position in the character-word pair sequence, inputs are represented as (h_i^c, x_i^{ws}), where h_i^c is the character vector obtained from a Transformer layer of BERT, and $x_i^{ws} = \{x_{i1}^w, x_{i2}^w, ..., x_{im}^w\}$ is a set of word embeddings. The j-th word in x_i^{ws} is represented as shown in Equation (15):

$$x_{ij}^w = \mathbf{e}^w(w_{ij}), \tag{15}$$

 doi:10.20944/preprints202406.1885.v1

where $\mathbf{e}^w$ is a pre-trained word embedding lookup table and w_{ij} is the j-th word in ws_i. It's important to note that the Address Semantic Extractor (ASSE) may yield words in $\mathbf{SS_s}$ that are not in the vocabulary. In such cases, these words are split into multiple components and their corresponding word vectors are looked up individually using Equation (12).

To ensure compatibility between these two representations, a nonlinear transformation is applied to the word vectors, as shown in Equation (16):

$$v_{ij}^w = \mathbf{W}_2(\tanh(\mathbf{W}_1 x_{ij}^w + \mathbf{b}_1)) + \mathbf{b}_2, \tag{16}$$

where $\mathbf{W}_1$ is a $d_c \times d_w$ matrix, $\mathbf{W}_2$ is a $d_c \times d_c$ matrix, and $\mathbf{b}_1$ and $\mathbf{b}_2$ are scalar biases. d_w and d_c represent the dimensions of word embeddings and BERT's hidden layers, respectively.

As depicted in Figure 3, each Chinese character corresponds to multiple words. However, each word contributes differently to each task. To select the most relevant words from all matched words, a character-to-word attention mechanism is introduced. All v_{ij}^w assigned to the i-th character are represented as $V_i = (v_{i1}^w, ..., v_{im}^w)$, where m is the total number of assigned words. The relevance of each word can be calculated using Equation (17):

$$\mathbf{a}_i = \text{softmax}(h_i^c \mathbf{W}_{attn} V_i^T), \tag{17}$$

where $\mathbf{W}_{attn}$ is the weight matrix for bilinear attention. Thus, the weighted sum of all words is obtained using Equation (18):

$$z_i^w = \sum_{j=1}^{m} a_{ij} v_{ij}^w, \tag{18}$$

Finally, the weighted lexicon information is injected into the character vector using Equation (19):

$$\tilde{h}_i = h_i^c + z_i^w, \tag{19}$$

(3) Lexicon-Enhanced BERT

Lexicon-Enhanced BERT (LEBERT) combines the Lexicon Adapter (LA) with BERT, where LA is applied to a specific layer of BERT to incorporate external lexicon knowledge into the model. Given a Chinese sentence $\mathbf{s}_c = \{c_1, c_2, ..., c_n\}$, we construct the corresponding character-word pair sequence $\mathbf{s}_{cw} = \{(c_1, ws_1), (c_2, ws_2), ..., (c_n, ws_n)\}$.

Firstly, the characters $\{c_1, c_2, ..., c_n\}$ are input into the input embeddings, which include token, segment, and positional embeddings, resulting in $E = \{e_1, e_2, ..., e_n\}$. These embeddings E are then passed through the transformer encoder, where each transformer layer operation is defined as in Equation (20):

$$\begin{aligned} G &= \text{LN}(H^{l-1} + \text{MultiHeadAttention}(H^{l-1})) \\ H^l &= \text{LN}(G + \text{FFN}(G)) \end{aligned}, \tag{20}$$

where $H^l = \{h_1^l, h_2^l, ..., h_n^l\}$ represents the output of the l-th layer, $H^0 = E$; LN denotes layer normalization; MultiHeadAttention refers to the multi-head attention mechanism; FFN denotes a two-layer feed-forward network with ReLU as the hidden activation function.

To inject lexicon information between the k-th and $(k+1)$-th Transformer layers, we first obtain the output $H^k = \{h_1^k, h_2^k, ..., h_n^k\}$ after passing through k consecutive Transformer layers. Then, each pair (h_i^k, x_i^{ws}) is processed through the Lexicon Adapter, transforming the i-th pair into $\tilde{h}_i^k$, as shown in Equation (21):

$$\tilde{h}_i^k = \text{LA}(h_i^k, x_i^{ws}), \tag{21}$$

 doi:10.20944/preprints202406.1885.v1

Given that BERT consists of $L = 12$ Transformer layers, $\widetilde{H}^k = \{\tilde{h}_1^k, \tilde{h}_2^k, ..., \tilde{h}_n^k\}$ is then input into the remaining $(L - k)$ Transformers, resulting in the output H^L of the L-th Transformer for sequence labeling tasks.

3.5. Encoding-Decoding and Loss Function

This section discusses the encoding layer, decoding layer, and loss function of the Lexicon-Enhanced Deep Parsing Model (ASSPM) that integrates spatial semantics. To better capture local dependencies between words and enhance spatial semantics, the output of enhanced BERT serves as input for subsequent steps, employing a Bidirectional Gated Recurrent Unit (BiGRU) as the encoder. The BiGRU reads sequences in both forward and backward directions, capturing lexical context information while preserving the order of words. Each BiGRU layer's output includes features learned from the previous layer, providing higher-level abstractions for subsequent layers, which facilitates learning structural properties of addresses from different perspectives.

In the decoding layer, considering the dependencies between labels in sequence labeling tasks, the model adopts a Conditional Random Field (CRF) as the final decoding layer. CRF effectively captures dependencies between labels and optimizes tasks such as named entity recognition based on input sequence features. Given the output $H^L = \{h_1^L, h_2^L, ..., h_n^L\}$ from the last layer, the score matrix O is computed as shown in Equation (22):

$$O = \mathbf{W}_o H^L + \mathbf{b}_o, \tag{22}$$

where O is an $n \times K$ matrix, n is the length of the input sequence, and K is the number of label categories. Each element $O_{i,j}$ in O represents the score for the j-th label category at the i-th position in the input sequence. These scores are used to compute the conditional probability of a given label sequence $\mathbf{y} = \{y_1, y_2, ..., y_n\}$ as shown in Equation (23):

$$p(\mathbf{y}|\mathbf{s}) = \frac{\exp(\sum_i (O_{i,y_i} + T_{y_{i-1},y_i}))}{\sum_{\tilde{\mathbf{y}}} \exp(\sum_i (O_{i,\tilde{y}_i} + T_{\tilde{y}_{i-1},\tilde{y}_i}))}, \tag{23}$$

where T is the transition score matrix, and $\tilde{\mathbf{y}}$ denotes all possible label sequences.

The loss function for N labeled data $\{\mathbf{s}_j, \mathbf{y}_j\}|_{j=1}^N$ is defined to minimize the negative log-likelihood at the sentence level, as shown in Equation (24):

$$\mathcal{L} = -\sum_j \log(p(\mathbf{y}|\mathbf{s})), \tag{24}$$

This section provides a detailed construction of the encoding layer, decoding layer, and loss function. By employing a Bidirectional GRU encoder, the model effectively learns local dependencies between words and explores structural properties of addresses at different levels. Furthermore, using Conditional Random Field (CRF) as the decoding layer optimizes sequence labeling tasks such as named entity recognition. Finally, minimizing the sentence-level negative log-likelihood serves to enhance model performance.

4. Experiment

4.1. Dataset Processing

To thoroughly validate the proposed deep parsing model in this chapter, two datasets with different annotation standards were utilized: 1) ChineseAddress (CADD) dataset [2]; 2) DOOR dataset, developed under the National Key R&D Program — Urban Daily Management Data Integration and Event Dynamic Simulation (2018YFB2100603), based on door address data from the legal and administrative address database.

 doi:10.20944/preprints202406.1885.v1

Initially, access to the legal and administrative address database was secured through a signed user privacy confidentiality agreement. The DOOR dataset comprises 94,078 addresses, with lengths ranging from 16 to 41 characters and an average length of 22 characters. Address components include Province (PROVINCE), City (CITY), County (COUNTY), Community (COMMUNITY), Point of Interest (POI), Town (TOWN), Village (VILLAGE), Road (ROAD), and Door Number (DOOR). An example address format is “浙江省温州市苍南县矾山镇杨子山村圆盘**-*号”(Zhejiang Province, Wenzhou City, Cangnan County, Fanshan Town, Yangzishan Village, Yuanpan **-*). For privacy protection, numeric information in addresses is masked with "*".

Then, 10,000 addresses were randomly sampled from the DOOR dataset. Utilizing the address component information provided by the legal and administrative address database facilitated straightforward data labeling. Finally, the dataset was split into training, validation, and test sets in a 6:2:2 ratio, resulting in the creation of the DOOR dataset. This dataset comprises 6,000 training samples, 2,000 validation samples, and 2,000 test samples. The format of the data is structured as follows: "Zhejiang Province (PROVINCE), Wenzhou City (CITY), Cangnan County (COUNTY), Fanshan Town (TOWN), Yangzishan Village (COMMUNITY), Yuanpan **-* (DOOR)".

During data preprocessing, the use of a Visual Assisted Labeling Module (ALM) ensured careful inspection of data quality, removing records with missing or anomalous values. Additionally, the dataset was balanced to mitigate potential biases during model training. The annotation process prioritized accuracy for each address component and addressed noise issues present in the dataset.

In summary, this paper provides a detailed description of dataset acquisition, preprocessing, partitioning, and annotation processes, while maintaining a focus on data quality and respecting user privacy. Next, these two datasets will be used to train and validate the proposed deep parsing model.

4.2. Experimental Setup and Evaluation Metrics

The hardware configuration and software environment required for the experiments in this study are detailed in Table 1.

Table 1. Experimental Setup Configuration.

Configuration Item	Value
Framework	PyTorch
Python Version	3.8.5
PyTorch Version	1.12.1+cu113
Transformers Version	4.33.2
Experiment Device	Precision 3930 Rack Workstation
CPU	3.3 GHz Intel(R) Xeon(R) E-2124
GPU	NVIDIA RTX A4000 × 2
Memory	64GB
Operating System	64-bit Windows 10 Professional Workstation

We utilized 200-dimensional pre-trained word embeddings [23], obtained by Song et al. through training a directional skip-gram model on news and web text [1]. The dictionary D used in this study corresponds to the vocabulary of these pre-trained word embeddings. To ensure fairness, we incorporated all address elements from the Address dataset used in the Address Spatial Semantic Extractor (ASSE), as well as the CADD and DOOR datasets' training sets. Corresponding word vectors were obtained by splitting address elements into multiple words and retrieving their vectors individually, calculated as per Equation (12). Similar to the approach by Liu et al. [20], we employed a lexicon adapter between the first and second Transformers of BERT, with a maximum token length set to 4, filtering out tokens exceeding this length. Since the Deep Parsing Model converts parsing tasks

[1] Pre-trained embeddings link: https://ai.tencent.com/ailab/nlp/en/data/tencent-ailab-embedding-en-d200-v0.1.0-s.tar.gz

 doi:10.20944/preprints202406.1885.v1

into sequence labeling tasks, specifically Named Entity Recognition (NER), evaluation metrics include Precision (P), Recall (R), and F1-score (F1). These metrics are calculated on a per-address element basis, defined as:

$$P = \frac{TP}{TP + FP}, \tag{25}$$

$$R = \frac{TP}{TP + FN}, \tag{26}$$

$$F1 = \frac{2 \cdot P \cdot R}{P + R}, \tag{27}$$

where TP denotes True Positives (correctly retrieved address tokens, irrespective of element type), FP indicates False Positives (address tokens in the standard segmentation that were not correctly retrieved), and FN represents False Negatives (address tokens missing from the output that should have been retrieved).

The hyperparameter configuration is presented in Table 2:

Table 2. Hyperparameter Configuration Table.

Parameter Name in English	Value	Parameter Name and Description in Chinese
Attention Heads	4	Number of attention heads: This refers to the number of heads in multi-head self-attention mechanisms, allowing the model to focus on different parts of input sequences at different positions, enhancing its ability to capture contextual information.
Batch Size	32	Batch size: The number of data samples used in each iteration to update the model weights. Larger batch sizes can accelerate training but may require more memory.
Dropout Ratio	0.2	Dropout rate: The probability of randomly dropping neurons during training, used to prevent overfitting and enhance model generalization capability.
Embedding Size d	200	Embedding dimension d: Dimensionality of embedding vectors used to represent input data, mapping discrete features into continuous vector spaces, commonly employed in natural language processing and recommendation systems.
Hidden State Size d_{model}	32	Hidden state dimension d_{model}: Dimension of hidden states in Transformer models, representing abstract representations of inputs at each position, influencing model representational capacity.
Learning Rate	5e-5	Learning rate: Controls the rate at which model parameters are updated during training, crucial for balancing training speed and performance, often requiring adjustment.
Max Seq Length	128	Maximum sequence length: Maximum allowable length of input sequences. Input sequences longer than this may need to be truncated or padded to fit the model.
Transformer Encoder Layers	6	Transformer encoder layers: Number of encoder layers in the Transformer model, determining model complexity and hierarchical performance.
Weight Decay	0.01	Weight decay: Regularization technique to mitigate model overfitting and improve generalization by penalizing large weights.
Warmup Ratio	0.1	Warmup ratio: Proportion of total training epochs dedicated to gradually increasing the learning rate, typically used to stabilize training in the initial phase.

 doi:10.20944/preprints202406.1885.v1

4.3. Comparison with Baseline Methods

To further validate the effectiveness of the proposed ASSRM model, we conducted comparisons with several baseline methods. These baseline methods represent current mainstream deep learning approaches, including Lattice-LSTM, SoftLexicon-BERT, ALBERT-BiGRU-CRF, BiGRU-CRF, LEBERT, GeoBERT, BERT-BiGRU-CRF, BABERT, RoBERTa-BiLSTM-CRF, and MGeo, totaling 10 different methods. Here are the details of Lattice-LSTM, ALBERT-BiGRU-CRF, SoftLexicon, BiGRU-CRF, LEBERT, GeoBERT, and BERT-BiGRU-CRF:

(1) Lattice-LSTM: Zhang et al. [14] introduced the Lattice-LSTM model at the ACL conference in 2018. This model aims to address issues faced by character-based and word-based methods. It jointly encodes characters of input sequences and their potential words from a dictionary into a lattice structure. Compared to character-based methods, Lattice-LSTM explicitly utilizes lexical and sequential information. Unlike word-based methods, it avoids segmentation errors since it does not require segmentation operations.
(2) ALBERT-BiGRU-CRF: Introduced by Lan et al. [26] in 2020, ALBERT-BiGRU-CRF is based on the BERT pre-trained language representation model. This model enhances performance by incorporating fixed-length context into input sequences and employs bidirectional encoder representations within a Transformer architecture. Compared to traditional BERT models, ALBERT has fewer parameters and achieves excellent results across various NLP tasks, thereby advancing research and applications in the NLP field.
(3) SoftLexicon-BERT: Ma et al. [15] proposed SoftLexicon in 2020, an innovative approach combining dictionaries with neural networks for Chinese Named Entity Recognition (NER). This method introduces SoftLexicon to share lexical information between encoder and decoder, thus improving overall performance.
(4) BiGRU-CRF: Liu et al. [27] introduced this model in 2021, which integrates deep learning with traditional methods for Chinese address parsing. BiGRU-CRF utilizes the strong contextual memory of Bidirectional Gated Recurrent Units (Bi-GRU) to effectively handle unknown words and ambiguities without relying on dictionaries or manual features. The Conditional Random Field (CRF) layer ensures reasonable output labeling sequences. It excels in address parsing by handling patterns like intersection addresses and spatially vague offsets. Using a five-word tagging method enhances granularity for subsequent matching and positioning operations.
(5) LEBERT: Liu et al. [20] proposed LEBERT in 2021, which explores the integration of lexicon information with pre-trained models like BERT. Unlike existing methods that shallowly fuse lexicon features through a single shallow and randomly initialized sequence layer, LEBERT directly integrates external lexical knowledge into the BERT layer through a lexicon adapter layer, achieving deep fusion of lexicon knowledge and BERT representation for Chinese sequence labeling tasks.
(6) GeoBERT: Liu et al. [28] introduced GeoBERT in 2021, a geographical information pre-training model. By constructing multi-granularity geographical feature maps, GeoBERT maps Points of Interest (POI) and various administrative regions into a unified graph representation space, embedding geographical information into text semantic space through predicting masked geographical information. This allows the model to learn geographical distribution characteristics for distinguishing different city POIs.
(7) BERT-BiGRU-CRF: Lv et al. [29] proposed this model in 2022 for geographic named entity recognition based on deep learning. It combines BERT's bidirectional encoder representations, Bidirectional GRU (BiGRU), and Conditional Random Field (CRF). The model utilizes pre-trained language models to construct integrated deep learning models, enriching semantic information in character vectors to improve the extraction capabilities of complex geographical entities.
(8) BABERT: Jiang et al. [4] introduced BABERT in 2022, a pre-trained language model that enhances performance by introducing fixed-length context into input sequences and employing an unsupervised statistical boundary information method. Experimental results demonstrate BABERT's

 doi:10.20944/preprints202406.1885.v1

stable performance improvement across various Chinese sequence labeling tasks. Additionally, BABERT serves as a complement to supervised dictionary exploration, potentially achieving better performance improvements by further integrating external dictionary information. During pre-training, the model learns rich geographical information and semantic knowledge, enabling accurate identification of key entities and assigning correct labels. Compared to traditional dictionary-based methods, BABERT better captures boundary information in addresses, thereby improving address parsing accuracy.

(9) RoBERTa-BiLSTM-CRF: Zhang Hongwei et al. [30] proposed this method in 2022, combining deep learning for Chinese address parsing. It aims to solve problems with existing address matching methods that overly rely on segmentation dictionaries and fail to effectively identify address elements and their types. The model standardizes and analyzes address components post-parsing to enhance address parsing performance.

(10) MGeo: Ding et al. [5] introduced MGeo in 2023, a multi-modal geographic language model containing a geographic encoder and a multi-modal interaction module. It views geographical context as a new modality and fully extracts multi-modal correlations to achieve precise query and matching functionalities.

These baseline methods are compared to demonstrate that the proposed Spatial Semantic Integration in Deep Parsing Model (ASSRM) has superior performance in address parsing. Table 3 provides specific experimental results.

Table 3. Comparative performance results on the CADD and DOOR datasets against baseline methods.

Method	CADD			DOOR		
	P	R	F	P	R	F
Lattice-LSTM[14] (2018)	89.47	89.11	89.29	90.55	90.19	90.37
SoftLexicon-BERT [15] (2020)	90.65	90.47	90.56	91.88	91.72	91.80
ALBERT-BiGRU-CRF[26](2020)	90.63	90.55	90.59	92.07	91.80	91.93
BiGRU-CRF[27](2021)	92.46	92.68	92.57	93.84	93.78	93.81
LEBERT[20](2021)	93.25	92.43	92.84	94.08	93.63	93.85
GeoBERT[28] (2021)	94.46	91.29	92.85	95.43	92.45	93.92
BERT-BiGRU-CRF[29](2022)	**94.90**	91.79	93.32	95.28	93.56	94.41
BABERT[4] (2022)	94.53	91.67	93.08	94.40	93.58	93.99
RoBERTa-BiLSTM-CRF[30](2022)	93.57	92.65	93.11	94.50	94.13	94.31
MGeo[5] (2023)	93.52	92.82	93.17	94.48	94.28	94.38
ASSPM (Our)	94.33	**93.65**	**93.99**	**95.52**	**94.31**	**94.91**

From the experimental results in Table 3, it is evident that the ASSPM model performs significantly better than all comparative methods on both datasets. Specifically, on the CADD dataset, its F1 score surpasses the second-ranked (BERT-BiGRU-CRF) by 0.67%, and on the DOOR dataset by 0.5%. This strongly validates the effectiveness of incorporating spatial semantic features in enhancing model performance. Meanwhile, methods like SoftLexicon-BERT and LEBERT, which utilize dictionary information, show less pronounced improvements compared to other methods, indicating limited efficacy from solely introducing dictionary features. In contrast, ASSPM demonstrates substantial performance gains after integrating spatial features, highlighting the importance of supplementary information provided by spatial features.

ASSPM shows particularly notable superiority in recall (R) metrics on both datasets, surpassing the second-ranked by 1.86% on CADD and 1.07% on DOOR, indicating that incorporating spatial features helps reduce model omission rates. Overall, ASSPM achieves an approximately 1% absolute gain on both datasets, a significant improvement in parsing tasks that fully demonstrates the method's effectiveness. In conclusion, the ASSPM model, which integrates spatial semantic features, exhibits outstanding performance in experiments, offering new perspectives and insights for Chinese address parsing in high-resource environments.

 doi:10.20944/preprints202406.1885.v1

To comprehensively evaluate the proposed model, in-depth efficiency analyses were conducted. During experiments, two GPUs were used for accelerated model training, with various batch sizes tested. It was found that a batch size of 32 achieved optimal training speed and balanced performance. Throughout the training process, multiple experiments were conducted to analyze key training parameters such as learning rate, number of epochs, and regularization coefficients. The choice of learning rate was found to significantly impact model performance, with rates that were too high or too low leading to unstable training. After optimization, an appropriate learning rate was selected to ensure rapid convergence and good generalization capability of the model.

4.4. Ablation Experiments

In this section, all experimental configurations and baseline setups are identical. Ablation experiments were conducted by removing different modules to assess their respective impacts within the model. Table 4 presents the results of these ablation experiments.

Table 4. The results of the ablation experiments.

Ablation method	Dateset					
	CADD			DOOR		
	P	R	F	P	R	F
Baseline (ASSPM)	94.33	93.65	93.99	95.52	94.31	94.91
A-ESL w/o EBMSoftLexicon	1.17$^{\downarrow}$	1.53$^{\downarrow}$	1.35$^{\downarrow}$	1.16$^{\downarrow}$	2.05$^{\downarrow}$	1.61$^{\downarrow}$
A-SSLEBERT w/o SSLEBERT	0.87$^{\downarrow}$	1.23$^{\downarrow}$	1.05$^{\downarrow}$	0.87$^{\downarrow}$	1.77$^{\downarrow}$	1.33$^{\downarrow}$
A-S w/o S	0.77$^{\downarrow}$	1.13$^{\downarrow}$	0.95$^{\downarrow}$	0.71$^{\downarrow}$	1.61$^{\downarrow}$	1.17$^{\downarrow}$
A-BiGRU w/o BiGRU	0.63$^{\downarrow}$	0.99$^{\downarrow}$	0.81$^{\downarrow}$	0.55$^{\downarrow}$	1.45$^{\downarrow}$	1.01$^{\downarrow}$

Among them, "w/o" is an abbreviation for "without," indicating the absence of something. First, the first line sets ASSE's parsing results as the baseline. Next, the second line's A-ESL model removes the proposed character fusion module (EBMSoftLexicon), i.e., inputs the original character vectors into BERT. The A-SSLEBERT model in the third line removes the spatial semantic lexicon enhanced BERT preprocessing model (SSLEBERT), i.e., makes no modifications to BERT. The A-S model in the fourth line replaces SSLEBERT with LEBERT, further clarifying the effectiveness of the proposed method. The A-BiGRU model in the fifth line removes the BiGRU encoder, directly using the output of SSLEBERT with a CRF decoder to obtain the final results.

Comparing the results of the ablation experiments, the following conclusions can be drawn:

(1) The effectiveness of EBMSoftLexicon in improving model performance is validated. The A-ESL model, which removes the character fusion module (EBMSoftLexicon), shows a decrease in F1 score by nearly 1.5%. This indicates that lexical information plays an important role in address text, and the character fusion module is crucial for enhancing model performance. By integrating lexical information, the character fusion module provides more accurate context understanding for the model, especially in handling complex address contexts.

(2) A-SSLEBERT and A-S jointly demonstrate the effectiveness of SSLEBERT in improving model performance, emphasizing the significant research significance of the proposed enhanced spatial fusion method in Chinese address parsing tasks. The A-SSLEBERT model, by removing the spatial semantic lexicon enhanced BERT preprocessing model (SSLEBERT), shows a decrease in F1 score by approximately 1.2%. This underscores the effectiveness of SSLEBERT in enhancing model performance. This module enhances BERT's representation by leveraging spatial semantic information, which is crucial for addressing complex contexts in Chinese address parsing tasks.

(3) In different scenarios, the applicability of the spatial semantic lexicon to improving model performance is demonstrated. The A-S model, which replaces SSLEBERT with LEBERT, results in a decrease in F1 score by about 1%. The introduction of LEBERT further clarifies the effectiveness of the proposed method, especially in handling address information in different contexts.

 doi:10.20944/preprints202406.1885.v1

(4) In the last line's A-BiGRU model, BiGRU plays a significant role. Removing this encoder results in a decrease in F1 score by nearly 1%, indicating that BiGRU significantly contributes to performance in address parsing tasks. Particularly in handling address information in different scenarios, BiGRU makes a notable contribution to performance improvement.

4.5. Results Discussion

This section provides a deeper analysis of the experimental results from Tables 3 and 4, aiming to further understand the importance and contributions of ASSPM.

From the results in Table 3, it is evident that the proposed Spatial Semantic Fusion Deep Parsing Model (ASSPM) exhibits outstanding performance on the CADD and DOOR datasets. Particularly noteworthy is ASSPM achieving an F1 score of 93.99 on the CADD dataset and 94.91 on the DOOR dataset. These results clearly demonstrate the superior performance of ASSPM in Chinese address parsing tasks, surpassing previous baseline methods and recent related research.

Comparisons with other models further emphasize the superiority of ASSPM. For instance, compared to the SoftLexicon-BERT model, ASSPM shows better performance on the DOOR dataset, indicating its stronger capability in integrating spatial semantic information. Additionally, compared to the state-of-the-art RoBERTa-BiLSTM-CRF and BABERT models, ASSPM performs better on the CADD dataset. These comparative results highlight the uniqueness and performance improvement of the proposed method.

A detailed analysis of the results from ablation experiments shown in Table 4 helps understand the roles of internal components within the ASSPM model, emphasizing their critical contributions to enhancing model performance.

Firstly, the effectiveness of the EBMSoftLexicon module is clearly validated. Removing this module from ASSPM results in a decrease in F1 score by nearly 1.5%. This underscores the crucial role of integrating lexical information for improving model performance in address texts. The introduction of EBMSoftLexicon enhances the model's understanding and processing of lexical information in Chinese address texts, thereby boosting performance.

Secondly, the comparison between SSLEBERT and LEBERT modules highlights the effectiveness of SSLEBERT in enhancing model performance. Comparing SSLEBERT with LEBERT in the A-SSLEBERT and A-S models clarifies the pivotal role of SSLEBERT. In the A-S model, replacing SSLEBERT with LEBERT shows that SSLEBERT significantly enhances model performance. This further underscores the importance of enhancing spatial semantic information and model integration.

Lastly, the A-BiGRU model in the ablation experiments demonstrates the importance of the BiGRU encoder. Removing the BiGRU encoder from the A-BiGRU model results in a decrease in F1 score by nearly 1%, highlighting the critical role of BiGRU in address parsing tasks, especially in encoding and understanding address texts.

In summary, the ASSPM model proposed in this study not only achieves effective integration of spatial semantic and deep learning for address parsing but also provides a practical solution to enhance accuracy and efficiency in addressing traffic accident processing. These findings are not only theoretically significant for Chinese address parsing but also offer robust technical support for practical applications such as traffic accident handling, promising to significantly improve work efficiency and service quality in related fields.

5. Conclusion and Future Work

This study addresses the accuracy and efficiency issues in address parsing for traffic accident handling by proposing a Deep Address Parsing Model with Spatial Semantic Fusion (ASSPM). The model effectively addresses the limitations of existing deep learning models in fully utilizing spatial semantic information in Chinese address parsing tasks. Initially, the model employs a hybrid representation-based lexical enhancement method to transform spatial semantics into a symbolic semantic space that can be integrated with the feature vector space of deep models. This integration achieves spatial seman-

tic fusion with BERT-based deep parsing models at the model level. Furthermore, an enhanced spatial semantic dictionary adapter is introduced to embed address spatial semantics into the Transformer layers of the BERT model, enabling spatial semantic fusion with BERT-based deep parsing models at the BERT layer level. Experimental results demonstrate that ASSPM performs well in high-resource scenarios. This study provides strong support for the application of deep address parsing models with spatial semantic fusion in traffic accident handling. Future work will focus on further enhancing model performance, expanding its application scope, and exploring more innovative technological integrations to address various challenges in practical applications. Ultimately, this aims to achieve a more efficient and accurate intelligent traffic accident handling system.

Author Contributions: Conceptualization, Kun Li, Feng Wang, and Guangming Ling; methodology, Kun Li, Feng Wang, and Guangming Ling; software, Guangming Ling and Feng Wang; validation, Kun Li and Feng Wang; resources, Kun Li and Feng Wang; data curation, Kun Li; writing—original draft preparation, Kun Li and Guangming Ling; writing—review and editing, Kun Li and Feng Wang; funding acquisition, Feng Wang.

Funding: This research was funded by the Science and Technology Breakthrough Project of Henan Province, China, with grant number 232102240019. The authors express their sincere gratitude to the Department of Science and Technology of Henan Province for their support, which ensured the smooth implementation of the research on deep learning-based traffic control methods. The resources provided by this project were crucial for the completion of this study.

Data Availability Statement: Not applicable.

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

References

1. Luo, S.; Li, Q.; Wei, F.; Zhou, X. Multi-source information fusion accident processing system based on knowledge graph. *Journal of Shandong University of Technology (Natural Science Edition)* **2024**, *38*, 28–34. doi:10.13367/j.cnki.sdgc.2024.03.005.
2. Li, H.; Lu, W.; Xie, P.; Li, L. Neural Chinese Address Parsing. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies(NAACL-HLT). Association for Computational Linguistics, 2019, pp. 3421–3431.
3. Ling, G.; Mu, X.; Wang, C.; Xu, A. Enhancing Chinese Address Parsing in Low-Resource Scenarios through In-Context Learning. *ISPRS International Journal of Geo-Information* **2023**, *12*, 296–316. doi:10.3390/ijgi12070296.
4. Jiang, P.; Long, D.; Zhang, Y.; Xie, P.; Zhang, M.; Zhang, M. Unsupervised Boundary-Aware Language Model Pretraining for Chinese Sequence Labeling. Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing. Association for Computational Linguistics, 2022, pp. 526–537. doi:10.18653/v1/2022.emnlp-main.34.
5. Ding, R.; Chen, B.; Xie, P.; Huang, F.; Li, X.; Zhang, Q.; Xu, Y. MGeo: Multi-Modal Geographic Pre-Training Method. Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval. ACM, 2023, pp. 185–194. doi:10.1145/3539618.3591728.
6. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies(NAACL-HLT). Association for Computational Linguistics, 2019, pp. 4171–4186.
7. Kang, M.; He, X.; Liu, C.; Wang, M.; Gao, Y. An Integrated Processing Strategy Considering Vocabulary, Structure and Semantic Representation for Address Matching. *Journal of Earth Information Science* **2023**, *25*, 1378–1385.
8. Weinzierl, M.A.; Maldonado, R.; Harabagiu, S.M. The Impact of Learning Unified Medical Language System Knowledge Embeddings in Relation Extraction from Biomedical Texts. *Journal of the American Medical Informatics Association* **2020**, *27*, 1556–1567.
9. Diao, S.; Bai, J.; Song, Y.; Zhang, T.; Wang, Y. ZEN: Pre-training Chinese Text Encoder Enhanced by N-gram Representations. Findings of the Association for Computational Linguistics(EMNLP). Association for Computational Linguistics, 2020, pp. 4729–4740. doi:10.18653/v1/2020.findings-emnlp.425.

10. Zheng, S.; Wang, F.; Bao, H.; Hao, Y.; Zhou, P.; Xu, B. Joint Extraction of Entities and Relations Based on a Novel Tagging Scheme. Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics(ACL); Association for Computational Linguistics: Vancouver, Canada, 2017; pp. 1227–1236. doi:10.18653/v1/P17-1113.
11. Kuru, O.; Can, O.A.; Yuret, D. CharNER: Character-Level Named Entity Recognition. Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers. The COLING 2016 Organizing Committee, 2016, pp. 911–921.
12. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF Models for Sequence Tagging. *Computer Science - Computation and Language* **2015**.
13. Tran, Q.; MacKinlay, A.; Yepes, A.J. Named Entity Recognition with Stack Residual LSTM and Trainable Bias Decoding. *IJCNLP* **2017**, pp. 566–575.
14. Zhang, Y.; Yang, J. Chinese NER Using Lattice LSTM. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL). Association for Computational Linguistics, 2018, pp. 1554–1564.
15. Ma, R.; Peng, M.; Zhang, Q.; Wei, Z.; Huang, X.J. Simplify the Usage of Lexicon in Chinese NER. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 5951–5960.
16. Li, Z.; Ding, N.; Liu, Z.; Zheng, H.; Shen, Y. Chinese Relation Extraction with Multi-Grained Information and External Linguistic Knowledge. Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics. Association for Computational Linguistics, 2019, pp. 4377–4386.
17. Gui, T.; Ma, R.; Zhang, Q.; Zhao, L.; Jiang, Y.G.; Huang, X. CNN-Based Chinese NER with Lexicon Rethinking. Twenty-Eighth International Joint Conference on Artificial Intelligence(IJCAI), 2019, pp. 4982–4988.
18. Sui, D.; Chen, Y.; Liu, K.; Zhao, J.; Liu, S. Leverage Lexical Knowledge for Chinese Named Entity Recognition via Collaborative Graph Network. Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). Association for Computational Linguistics, 2019, pp. 3830–3840.
19. Li, X.; Yan, H.; Qiu, X.; Huang, X.J. FLAT: Chinese NER Using Flat-Lattice Transformer. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, pp. 6836–6842.
20. Liu, W.; Fu, X.; Zhang, Y.; Xiao, W. Lexicon Enhanced Chinese Sequence Labeling Using BERT Adapter. 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). Association for Computational Linguistics, 2021, pp. 5847–5858. doi:10.18653/v1/2021.acl-long.454.
21. Cui, Y.; Che, W.; Liu, T.; Qin, B.; Yang, Z. Pre-Training With Whole Word Masking for Chinese BERT. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *29*, 3504–3514. doi:10.1109/TASLP.2021.3124365.
22. Chen, T.; Guestrin, C. XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Association for Computing Machinery, 2016, KDD '16, pp. 785–794. doi:10.1145/2939672.2939785.
23. Song, Y.; Shi, S.; Li, J.; Zhang, H. Directional Skip-Gram: Explicitly Distinguishing Left and Right Context for Word Embeddings. Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies(NAACL-HLT). Association for Computational Linguistics, 2018, pp. 175–180. doi:10.18653/v1/N18-2028.
24. Peng, N.; Dredze, M. Improving Named Entity Recognition for Chinese Social Media with Word Segmentation Representation Learning. Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics(ACL). Association for Computational Linguistics, 2016, pp. 149–155. doi:10.18653/v1/P16-2025.
25. Zhao, H.; Kit, C. Unsupervised Segmentation Helps Supervised Learning of Character Tagging for Word Segmentation and Named Entity Recognition. Proceedings of the Sixth SIGHAN Workshop on Chinese Language Processing, 2008.
26. Lan, Z.; Chen, M.; Goodman, S.; Gimpel, K.; Sharma, P.; Soricut, R. ALBERT: A Lite BERT for Self-supervised Learning of Language Representations. International Conference on Learning Representations, 2019.
27. Liu, X.y.; Li, Y.l.; Yin, B.; Tian, X. Chinese Address Understanding by Integrating Neural Network and Spatial Relationship. *Science of Surveying and Mapping* **2021**, *v.46;No.212*, 165–171.
28. Liu, X.; Hu, J.; Shen, Q.; Chen, H. Geo-BERT Pre-training Model for Query Rewriting in POI Search. Findings of the Association for Computational Linguistics(EMNLP). Association for Computational Linguistics, 2021, pp. 2209–2214. doi:10.18653/v1/2021.findings-emnlp.190.

 doi:10.20944/preprints202406.1885.v1

29. Lv, X.; Xie, Z.; Xu, D.; Jin, X.; Ma, K.; Tao, L.; Qiu, Q.; Pan, Y. Chinese Named Entity Recognition in the Geoscience Domain Based on BERT. *Earth and Space Science* **2022**, *9*, e2021EA002166. doi:10.1029/2021EA002166.
30. Hongwei, Z.; Qingyun, D.; Zhangjian, C.; Chen, Z. A Chinese Address Parsing Method Using RoBERTa-BiLSTM-CRF. *Wuhan University Journal of Natural Sciences: Information Science Edition* **2022**.