electronics

MDPI

Article

Demonstration-Based and Attention-Enhanced Grid-Tagging Network for Mention Recognition

Haitao Jia [1,†], Jing Huang [2,*,†], Kang Zhao [1], Yousi Mao [1], Huanlai Zhou [3], Li Ren [4], Yuming Jia [2] and Wenbo Xu [1]

[1] School of Resource and Environment, University of Electronic Science and Technology of China, Chengdu 611731, China
[2] School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China
[3] UESTC—Chengdu Quantum Matrix Technology Co., Ltd., Joint Institute of Data Technology, Chengdu 610066, China
[4] University of Electronic Science and Technology Library, University of Electronic Science and Technology of China, Chengdu 611731, China
* Correspondence: 202152011709@std.uestc.edu.cn
† These authors contributed equally to this work.

Abstract: Concepts empower cognitive intelligence. Extracting flat, nested, and discontinuous name entities and concept mentions from natural language texts is significant for downstream tasks such as concept knowledge graphs. Among the algorithms that uniformly detect these types of name entities and concepts, Li et al. proposed a novel architecture by modeling the unified mention recognition as the classification of word–word relations, named W^{2}NER, achieved state-of-the-art (SOTA) results in 2022. However, there is still room for improvement. This paper presents three improvements based on W^{2}NER. We enhanced the grid-tagging network by demonstration learning and tag attention feature extraction, so our modified model is named DTaE. Firstly, addressing the issue of insufficient semantic information in short texts and the lack of annotated data, and inspired by the demonstration learning from GPT-3, a demonstration is searched during the training phase according to a certain strategy to enhance the input features and improve the model's ability for few-shot learning. Secondly, to tackle the problem of W^{2}NER's subpar recognition accuracy problem for discontinuous entities and concepts, a multi-head attention mechanism is employed to capture attention scores for different positions based on grid tagging. Then, the tagging attention features are embedded into the model. Finally, to retain information about the sequence position, rotary position embedding is introduced to ensure robustness. We selected an authoritative Chinese dictionary and adopted a five-person annotation method to annotate multiple types of entities and concepts in the definitions. To validate the effectiveness of our enhanced model, experiments were conducted on the public dataset CADEC and our annotated Chinese dictionary dataset: on the CADEC dataset, with a slight decrease in recall rate, precision is improved by 2.78%, and the comprehensive metric F1 is increased by 0.89%; on the Chinese dictionary dataset, the precision is improved by 2.97%, the recall rate is increased by 2.35%, and the comprehensive metric F1 is improved by 2.66%.

Keywords: discontinuous mention recognition; grid tagging; demonstration learning; tag attention; rotary position embedding

check for updates

Citation: Jia, H.; Huang, J.; Zhao, K.; Mao, Y.; Zhou, H.; Ren, L.; Jia, Y.; Xu, W. Demonstration-Based and Attention-Enhanced Grid-Tagging Network for Mention Recognition. *Electronics* **2024**, *13*, 261. https://doi.org/10.3390/electronics13020261

Academic Editor: Byung-Gyu Kim

Received: 28 November 2023
Revised: 28 December 2023
Accepted: 2 January 2024
Published: 5 January 2024

1. Introduction

In the realm of human cognition, the concept stands as a paramount bridge between concrete elements and abstract understanding. It epitomizes the embodiment of human understanding of the myriad elements in our world, linking past experiences with present interactions and thereby enabling the crucial cognitive functions of recognizing and comprehending novel phenomena [1]. In essence, the human ability to comprehend and understand relies on associating their knowledge with the concrete world through concepts.

For instance, when the word “冻” (freeze) is mentioned, people can associate it with different concepts or entities such as “寒冷” (cold) and “皮冻” (frostbite), allowing them to perceive, understand, and infer in specific contexts. Furthermore, in the era when intelligent upgrading and transformation have become common demands across various fields, understanding, and interpretation have become some of the core missions of artificial intelligence in the post-deep-learning era. Prior knowledge, such as knowledge of concepts, is also essential to endow machines with cognitive intelligence. Both structured and unstructured texts contain rich concepts or entities. Entities refer to distinguishable and independently existing entities, while concepts represent collections of entities with similar characteristics. For a natural, language sequence like “深海中的鲨鱼令大多人恐惧,其缘由是它们是食肉的动物。” (“Sharks in the deep sea frighten most people; the reason is that they are carnivorous animals”.), people need to focus not only on the entity “鲨鱼” (shark) but also on broad concepts like “食肉动物” (carnivorous animal) and “动物” (animal). The comprehensive extraction of both named entities and overarching concepts from textual data is of paramount importance. It serves as a foundational step in the construction of concept knowledge graphs and facilitates a myriad of other downstream tasks.

Entities and concepts in natural language sequences, referred to as mentions in the following text, can be classified into three types: flat, nested, and discontinuous. Flat mentions imply that the words (or characters in Chinese text) in the mention are continuous and that none of its subsequences represent other entities or concepts. Nested mentions mean that the words in the mention are continuous, but they contain subsequences representing other entities or concepts. Discontinuous mentions indicate that the words in the mention are not contiguous. Accurately identifying the boundary of the entity and concept is extremely challenging. Additionally, using a unified model to address these three types of problems and refining the granularity of entity and concept representations has become a research hotspot in recent years.

Moreover, in the past two years, methods based on grid tagging have shown outstanding performance in the unified recognition of entities and concepts [2–6]. Among them, the W^{2}NER [3] model predicts the boundaries of entities and concepts as well as the adjacency word relationships using two grids. It then decodes the entire entity and concept, unifying the recognition problems of flat, nested, and discontinuous cases. Although this method has achieved SOTA results, there is still room for improvement. The SOTA model only focuses on the relationships between words within the grid, overlooking the impact of labels on the classification of word-pair relationships. Especially for discontinuous cases, it is crucial for the model to learn the word positions that it should pay more attention to, which is essential for improving the recognition of discontinuous entities and concepts.

In grid-tagging algorithms, inspired by Liu et al. [7], capturing the attention of grid tagging on words at different positions is valuable. In this way, this paper utilizes a multi-head attention mechanism to embed grid-tagging representations into the model, modeling the relationships between labels and the relationships between tags and words. However, pure attention mechanisms may lose sequence order information, and preserving the positional information of the sequence is essential for the effectiveness of grid-tagging algorithms.

So far, very few publicly available datasets include discontinuous entities and concepts, and there are no Chinese datasets of this kind. Among the limited publicly available datasets, discontinuous cases account for less than 10%. This paper aims to extract flat, nested, and discontinuous entities/concepts from authoritative Chinese dictionary definitions. However, manually annotated data are always limited. Therefore, in situations where the number of samples is small, the model’s learning ability and generalization capability still need improvement.

Additionally, authoritative Chinese dictionaries contain rich conceptual knowledge in the form of vocabulary and phrases. However, most natural language sentences in these dictionaries are short texts due to this nature. Short texts have characteristics such as sparse semantic context, flexible expression, and a lack of annotated data. Traditional algorithms find extracting accurate, complete, and fine-grained conceptual knowledge from these texts challenging. Moreover, as a complex language with limited resources, Chinese has highly flexible expression styles and grammar rules.

In summary, the mining of fine-grained entities and concepts from structured and semi-structured texts face challenges such as significant text noise, abundant missing data, insufficient semantic information, subpar recognition in discontinuous scenarios, and limited annotated data.

The contributions of this paper are as follows:

1. Based on the analysis of practical results obtained from mining mentions of paraphrases in the Chinese dictionary using W^{2}NER, we address the issue of insufficient semantic information in short texts. Furthermore, with the rise of Large Language Models (LLMs), in-context learning theory performs excellently in its performance by feeding a few task-related samples, which is widely applied in the field of Natural Language Processing (NLP) [8]. Concatenating task-related demonstration in the input text as a prompt is also an advantage for normal-scale models. To enhance the learning ability, inspired by task-specific demonstration proposed in GPT-3 [9], we introduce the concept of in-context learning. Following specific selection strategies, it samples inputs and labels from the training data, designs task-specific demonstration templates, constructs corresponding demonstration sequences, concatenates them with input sentences, and provides information enhancement. Simultaneously, it improves the model's ability to learn from limited samples.
2. To address the issue of lower accuracy in recognizing entities or concepts in discontinuous scenarios with the base model, we introduce a multi-head attention mechanism to capture attention scores for words at different positions based on different labels. The attention features of labels are embedded into the model. Additionally, an iterative mechanism is employed to update the representations of label attention, aiming to extract deep fusion features.
3. To tackle the problem of attention mechanisms losing sequence information, this paper introduces rotary position embedding [10], achieving relative encoding effects in an absolute embedding form. Additionally, we then propose an improved algorithm W^{2}NER [3], named the Demonstration and Tag attention Enhanced Grid-Tagging Network (DTaE). The overall architecture of DTaE is illustrated in Figure 1.
4. This study selected an authoritative Chinese dictionary and employed a five-person annotation method to annotate flat, nested, and discontinuous entities and concepts in the definitions. Experiments were conducted on the public dataset CADEC [11] and the Chinese dictionary dataset to validate the effectiveness of the proposed model.

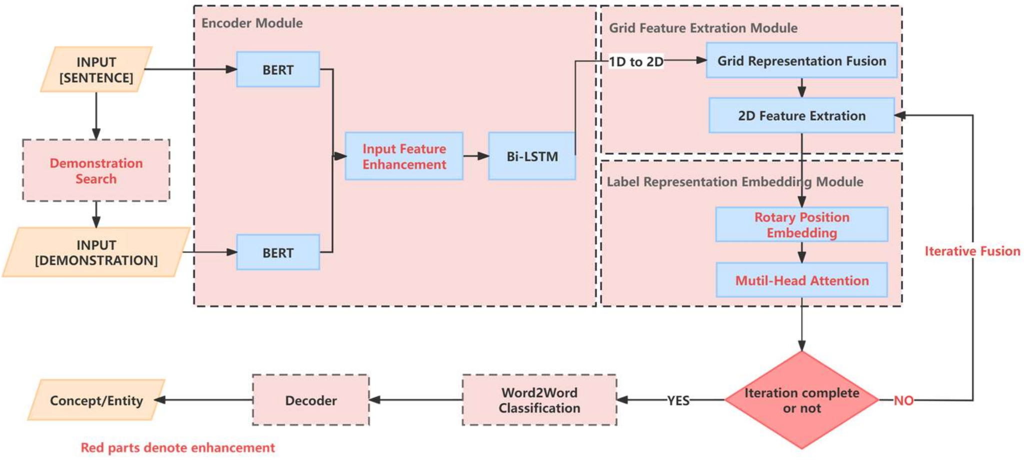


Figure 1. Detailed structure of modules within the enhanced DTaE model.

The subsequent arrangement of this paper is as follows: In Section 2, the current research progress in the relevant field will be introduced. In Section 3, we enhanced the grid-tagging algorithm W^{2}NER [3] by demonstrating learning and multi-head attention mechanism with rotary position embedding. In Section 4, the evaluation of the model's performance will be conducted.

2. Related Work

Entity and concept recognition aims to detect the boundary and semantic label of an entity or a concept mentioned from an input sentence, which, as a fundamental task in NLP, can be divided into three subtasks: flat, nested, and discontinuous named entity and concept recognition [6]. The earliest conventional research focuses on the recognition of flat mentions. In recent years, it has evolved from the conventional flat mention recognition to overlapped [12] and discontinuous mention recognition [13]. Previous methods can be divided into approximately four types: sequence labeling-based methods, span-based methods, generative methods, demonstration-based learning methods, and grid-tagging methods.

Sequence labeling-based methods treat nested mention-recognition tasks as sequence-labeling tasks, decoding them sequentially based on the order of the sequence. Since a character or word in nested entities can be annotated with multiple distinct labels, conventional sequence labeling models cannot directly recognize nested entities. Most learning models use Bidirectional Encoder Representation from Transformers (BERT) and Bidirectional Long Shot-TERM Memory (Bi-LSTM) to extract word- and character-level features, obtaining the context semantic information of the target word or character [3,7,14–18]. In Tang et al. [15], a multi-task BERT-Bi-LSTM-AM-CRF intelligent processing model was constructed to classify the observation annotation sequence to obtain the named entities in a Chinese text. Methods employing sequence labeling for nested NER attempt to transform multi-label tasks into single-label tasks or modify decoders to accommodate multiple labels. Ju et al. [19] used dynamic stacking of flat entity layers to identify nested entities. The layered-Bi-LSTM-CRF model uses LSTM layers to capture sequential context representations and feeds them back to cascaded CRF layers. The output of the LSTM layers is merged into the current flat NER layer to construct new representations for detected entities, which are then input into the next flat NER layer. This approach fully utilizes the internal information of nested entities, extracting external entities from the inside out. The dynamic stacking of flat NER layers continues until no nested entities are left. Takashi et al. [20] proposed a novel method for learning and decoding nested entities iteratively from the outside in, addressing structural ambiguity problems. They used the Viterbi algorithm to recursively search the scope of each entity to find internal nested entities. This algorithm does not require predefining the maximum length and quantity of entities. Each entity type has its CRF layer to judge whether the decoding meets the constraint conditions. In order to label irregular entities and concepts, Tang et al. [21] proposed an improved schema system named BIOHD containing seven labels to uniformly recognize flat, nested, and discontinuous mentions through the Bi-LSTM-CRF network. However, methods based on sequence labeling have several limitations: they cannot train models in parallel, complex label types lead to sparse label distributions, and there are exposed error issues between layers.

Span-based methods treat nested named entity recognition (NER) as a subsequence-classification problem within a sentence, effectively addressing error-propagation issues [22–27]. The basic idea of these methods is to enumerate all possible subsequences given an input sequence and predict the labels for each subsequence using a classifier. To capture underlying semantic information in the text, approaches like Sohrab et al. [27] classify all possible text fragments as potential entity mentions using deep neural networks. Shared Bi-LSTM networks output representations of text fragments to reduce computational costs and capture contextual semantic information around the fragments. Fisher et al. [26] propose a method where terms or named entities combine into nested structures of named entities. This method recognizes such nested entities in two stages: first, it determines the boundaries of nested entities at different levels, and second, it generates embeddings for each named entity by combining embeddings of smaller terms, predicting the linkage probability between two terms. The representations of each term in the named entity are iteratively updated to recognize nested entities. Span-based methods suffer from high computational costs and lack clear boundary supervision, leading to numerous negative samples, which impact overall algorithm performance. To address these issues, Li F et al. [28] propose a segment-enhanced span-based model for nested NER (SESNER). This model treats the nested NER task as a segment coverage problem. It models entities as line segments, detects segment endpoints, and identifies positional relationships between adjacent endpoints. It detects outermost segments, generating candidate entity fragments for nested entities to classify text fragments. This model enhances boundary supervision in span representations by detecting segment endpoints, preserving long entities while reducing the number of negative samples and improving operational performance. Despite these advancements, span-based methods still face challenges, including excessive consideration of non-named entity regions, leading to numerous negative samples, high computational costs associated with classifying all subsequences in a sentence, and poor performance in boundary detection.

Generative methods aim to generate a named entity or a concept span sequence to fulfill the recognition task [2,29–31]. J. Straková et al. [30] identified flat and nested entities by generating label sequences and then decoding multi-labels. Tan et al. [31] generated span boundary and corresponding labels to detect flat and nested entities. Yan et al. [2] proposed a unified generative framework for various NER subtasks and utilized the unified pre-trained Seq2Seq model Bidirectional and Auto-Regressive Transformers [32] (BARTs) to address three entity recognition tasks. Yan et al. introduced three-pointer mechanisms to precisely locate entities within input sentences—span, byte pair encoding (BPE), and word—to use the BART model effectively. This framework is easy to implement, offering both performance and versatility. However, it still faces challenges related to decoding ambiguities and the inherent exposure bias problem in the Seq2Seq framework.

Demonstration-based learning methods have been researched widely in auto-regressive language models, with the recent introduction of a few training examples in a natural language prompt [33,34]. This kind of prompt is a task-specific demonstration that aims to make the model understand the training task [35]. Related efforts focused on data augmentation by generating synthetic labeled instances, which generate artificial samples but mostly ignore real training data. Thus, Chen et al. [36] proposed a novel description and demonstration-guided data-augmentation (D3A) approach for sequence annotation. Lee et al. [34] designed a demonstration-based approach for NER few-shot learning, which conducted a systematic study of the demonstration strategy, inspiring us to a great extent.

Grid-tagging methods convert a recognition task into a classification task that tags the relationship of each word pair in a grid transformed from a sentence [3,7,37]. In 2022, Li et al. a unified named entity recognition (NER) model called W^{2}NER [3]. They predefined three types of word-pair relationships: THW (Tail-Head-Word), NNW (Next Neighboring Word), and Unknown. First, they utilized pre-trained models for embedding to enhance the semantic representation capability of the model. Secondly, they employed advanced span annotation to convert the input into a two-dimensional grid, effectively alleviating decoding ambiguity issues. Next, they used a two-dimensional convolutional network to extract deep two-dimensional features. Finally, they employed joint biaffine and linear classifiers for word-pair relationship classification, obtaining named entities through decoding. This method addressed the challenges of entity boundary recognition difficulties and exposure bias. This approach is highly advanced by unifying the modeling of three types of entities and breaking through the core bottleneck through word-pair relationship classification. Furthermore, Liu et al. [7] improved recognition accuracy by expanding tag definition and the multi-head attention mechanism.

3. Methods

When addressing the challenges of flat, nested, and discontinuous entity types, it is essential to analyze their commonalities and transform them into a unified training task to construct an effective unified model. For entity-recognition problems, it is crucial to accurately identify the boundary words, internal words, and categories of entities. In this context, the state-of-the-art model in the field as of 2022, W^{2}NER [3], transformed entity recognition problems into word-pair relationship classification problems. In English, this corresponds to words, and in Chinese, it corresponds to characters (hereinafter referred to as "words" without specific mention of "characters"). The algorithmic process is as follows: First, words' semantic features are extracted via BERT-Bi-LSTM from input text. BERT and its variants are heavily used to extract semantic word features in Nature Language Processing and the proposed approach. Some studies [38] recently showed that these tools can often extract semantic meaning and sentiment in a way that matches human-generated text. Then, matrix-form position embedding and upper-triangular region embedding are introduced, combined with word features to construct features. Next, two-dimensional grid features are constructed through Conditional Layer Normalization (CLN), which is easier to extract using convolutional networks. Hybrid Dilated Convolution (HDC) generates a deeply interactive feature of two-dimensional object as well as W^{2}NER, which is extremely appropriate for handling gird feature extraction [3,39,40]. Subsequently, a joint classifier using Multilayer Perceptron (MLP) and Biaffine is employed to classify word-pair relationships within the grid. Finally, a decoding algorithm is designed to decode entities and concepts in the input text based on word-pair relationships.

Although W^{2}NER [3] has achieved significant state-of-the-art (SOTA) results in the unified recognition domain, its recognition capability in discontinuous scenarios still needs improvement. The network only focuses on the relationships between words and words, neglecting the influence of tagging on the classification of word-pair relationships. This paper proposes three improvements to address these issues: enhancing input features through demonstrations, strengthening the representation ability of tagging features using multi-head attention mechanisms, and preserving sequence position information through rotary position embedding.

In summary, this paper proposes an improved algorithm based on the state-of-the-art model W^{2}NER [3], named Demonstration and Tag-aware Enhanced Grid-Tagging Network (DTaE). The specific steps of the algorithm are as follows:

1. Searching for demonstration sentences: During the training phase, a certain search strategy is employed to sample input sequences and corresponding annotated data from the training set. Task-specific demonstration templates are designed, and the obtained input sequences and annotated data are filled into these templates to construct demonstrations. Relevant demonstration sequences are concatenated with input sequences, playing a role in task demonstration and information enhancement.
2. Word feature extraction and sentence feature enhancement: pre-trained models are used to embed input sequences, obtaining word features and sentence features. The demonstration information is also embedded using the same pre-trained model, resulting in a sentence vector indicated by the "[CLS]" token. This vector is weighted using learnable weights and fused with the input sequence's sentence vector to enhance the features. Furthermore, Bi-LSTM networks are utilized to embed the representation vectors of input sequences, introducing more semantic context information.
3. Construction of two-dimensional grid features: The vector representations of the input text are concatenated with position embedding and upper-triangular matrices to create BERT-style input. Conditional Layer Normalization is applied to merge grid representations.
4. Hybrid Dilation Convolution (HDC) for extracting two-dimensional features: Convolutional networks are naturally suited for extracting two-dimensional features. Although single fine-grained dilated convolutions can enlarge the receptive field, they face the grid effect: as the dilation rate increases, the input sampling becomes sparse,

potentially leading to the loss of local information. Multiple fine-grained mixed dilated convolutions are used to extract grid features, enlarging the receptive field while avoiding the grid effect and effectively extracting two-dimensional features.

5. Obtaining tag attention features: Tagging features are constructed, and multi-head attention mechanisms capture the attention weights of different labels at different positions in the input sequence. These attention features are iteratively updated to enhance the model's recognition ability. To address the issue of attention mechanisms losing sequence position information, rotary position embedding is introduced in a relative embedding form, achieving results similar to absolute embedding. This embedding advantage is fully utilized by incorporating rotary position embedding before the aforementioned multi-head attention mechanism.
6. Deepening features: Word-pair grid features and tag attention features are iteratively fused through multiple rounds of the steps mentioned in (4) and (5).
7. Joint classifier: A joint classifier consisting of a multilayer perceptron and biaffine predictors predicts the word-pair relationships for various grid indicators.
8. Decoding: Based on the predicted word-pair relationships within the grid, the decoding algorithm is applied to obtain flat, nested, and discontinuous entities and concepts in the input sequence.

According to the aforementioned process, the algorithm design diagram for our improved grid-tagging model is illustrated in Figure 2.

3.1. Demonstration Search

Demonstration-based learning is a simple and effective auxiliary supervised learning method originating from GPT-3. The core idea is to provide the model with task demonstrations, allowing the input to undergo context learning through these demonstrations in advance. This enables the model to understand better and predict the tasks.

In the context of unified recognition of flat, nested, and discontinuous entities and concepts in short texts, especially in enhancing recognition in discontinuous scenarios to obtain more accurate and granular concepts, this study introduces task-related demonstrations into entity- and concept-recognition algorithms as a means of information enhancement. The goal is to compensate for the lack of rich contextual information and the scarcity of data samples in short texts. The challenge lies in determining reliable search strategies and designing effective demonstration templates.

Search strategies aim to find instances in the training set as demonstrations for the task. This study adopts two demonstration strategies: random search and semantic search. A random seed is set in the random search strategy, and instances of different target types in the training set are counted. Instances of different types are randomly selected, and multiple demonstrations are concatenated to form demonstration sentences. The remaining sentences are used as training data. In the semantic search strategy, highly relevant sentences are searched for each instance of training data to serve as its demonstration sentence. High relevance means semantic relevance at the sentence level, with semantic differences at the word level. Semantic search requires a universal sentence vector representation. Searching for many highly related sentence pairs using BERT incurs significant time costs, and the obtained sentence representations are unsuitable for unsupervised semantic search. SentenceBERT networks can address these issues. This pre-trained model is based on a twin network derived from BERT, which can obtain semantically meaningful chapter vectors. The basic principle is to obtain sentence vectors for two sentences through the twin network, and the pre-training task involves calculating the similarity between the two sentences. We use SentenceBERT(SBERT) [41] to search semantically relevant sentences for input as a demonstration. Constructing highly relevant sentence pairs in the training set, with sentence 1 as the demonstration and sentence 2 as the training data point.

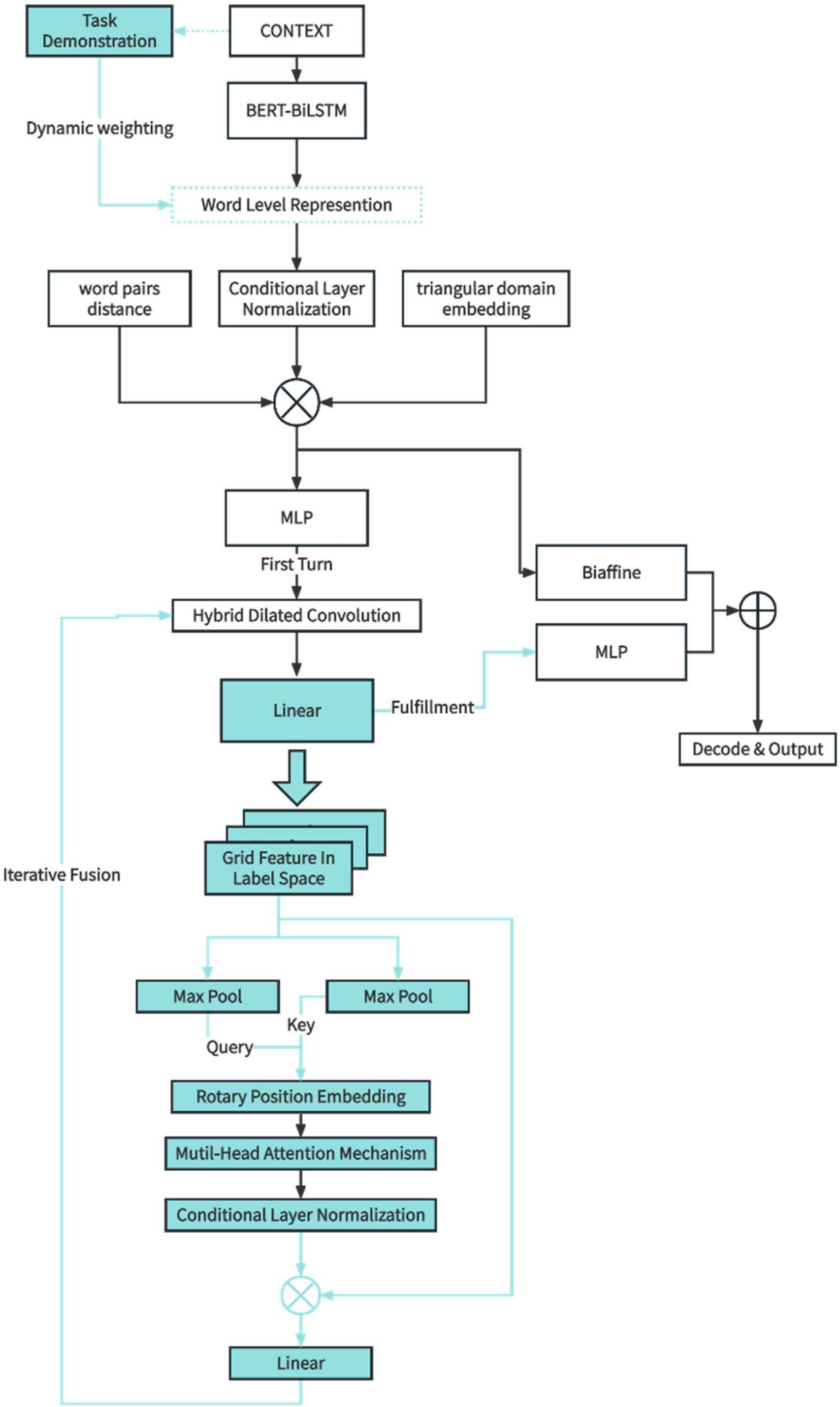

Task Demonstration
CONTEXT
BERT-BiLSTM
Dynamic weighting
Word Level Represention
word pairs distance
Conditional Layer Normalization
triangular domain embedding
MLP
First Turn
Biaffine
MLP
Hybrid Dilated Convolution
Linear
Fulfillment
Decode & Output
Grid Feature In Label Space
Iterative Fusion
Max Pool
Max Pool
Key
Query
Rotary Position Embedding
Mutil-Head Attention Mechanism
Conditional Layer Normalization
Linear

Figure 2. The overall architecture of our model.

Based on the search strategy, instances are searched in the training set to construct demonstrations. Specifically, a task demonstration can be represented as an input sequence $\widetilde{x} = [[SEP], \hat{x}_1, \cdots, \hat{x}_l]$, where l is the length of the sequence. Using instance-oriented demonstration templates, each instance includes an input sentence s and a set of annotated data S_e. If S_e is an empty set, there are no entities or concepts of the target type in the input sentence. Let the target type be $Type$. In this case, the demonstration sentence $\widetilde{x}$ is constructed as s + "There is no entity or concept named Type"., For Chinese demonstration sentences, it would be s + "上句中不存在Type实体或概念。". If S_e contains k entities or concepts, for each entity or concept e_i in S_e, an answer sentence ans_i is constructured: "e_i is entity or concept named $Type$". "or" "e_i 是一个$Type$实体或概念。" "The demonstration sentence $\widetilde{x}$ is then constructed as $s + ans_1 + \cdots + ans_k$".

Based on the above discussion, the two demonstration search algorithms used in this paper are illustrated in Figures 3 and 4.

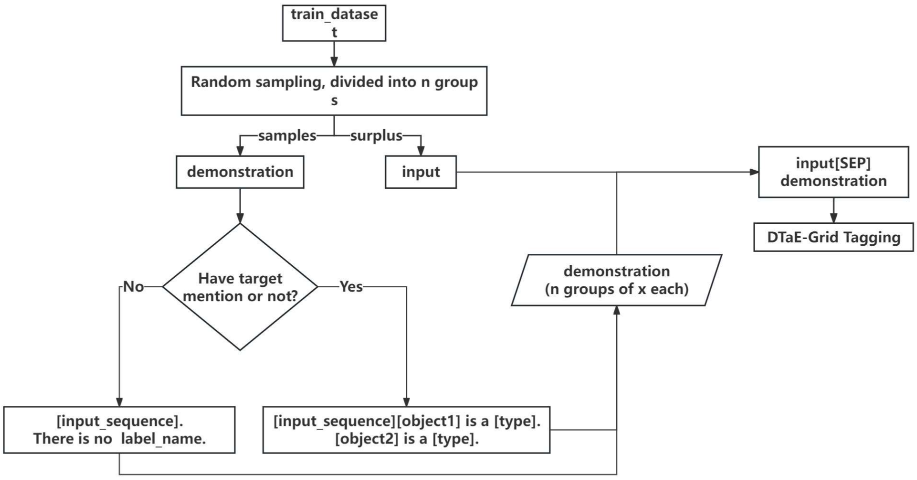

train_dataset
Random sampling, divided into n groups
samples
surplus
demonstration
input
input[SEP] demonstration
DTaE-Grid Tagging
demonstration (n groups of x each)
Have target mention or not?
No
Yes
[input_sequence]. There is no label_name.
[input_sequence][object1] is a [type]. [object2] is a [type].

Figure 3. Random search strategy of our model for a demonstration of the search task.

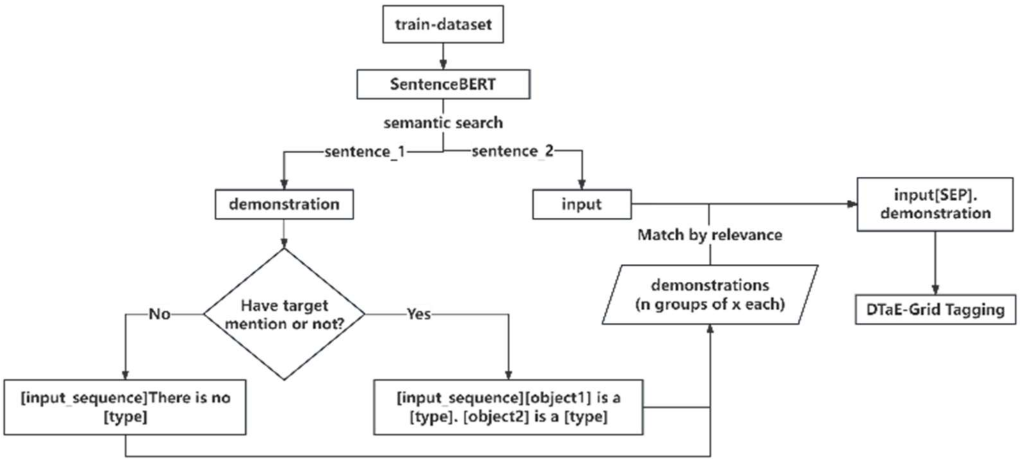

train-dataset
SentenceBERT
semantic search
sentence_1
sentence_2
demonstration
input
input[SEP]. demonstration
Match by relevance
demonstrations (n groups of x each)
DTaE-Grid Tagging
Have target mention or not?
No
Yes
[input_sequence]There is no [type]
[input_sequence][object1] is a [type]. [object2] is a [type]

Figure 4. Semantic search strategy for our model for the demonstration of a search task.

3.2. Sequence Embedding

Once the natural language sequence $X = \left[x; \widetilde{x}\right]$ is obtained, it needs to be embedded in three stages: pre-trained model embedding, demonstration enhancement, and context embedding. These stages are essential for obtaining enhanced input representations incorporating information from the pre-trained model and the demonstration data. Specifically, leveraging pre-trained models like BERT, the input sequence X is embedded to obtain semantic representations H', as shown in Equation (1) below, where n represents the length of training data x, l represents the length of the demonstration $\widetilde{x}$, h_x represents the semantic representation vector of x, and $h_{\widetilde{x}}$ represents the representation vector of $\widetilde{x}$.

$$H' = BERT([x_1, \cdots, x_n; [SEP], \hat{x}_1, \cdots, \hat{x}_l]) = [h_x, h_{\widetilde{x}}] \quad (1)$$

Next, the representation vector of the demonstration sentence is utilized to enhance the training data, introducing more task-related semantic information into the training dataset, thereby enhancing the semantic representation ability of short texts. Through a dynamic weighting network, $h_{\tilde{x}}$ is integrated into the semantic representation vector of x, obtaining the word-level representation vector H_x, as shown in Equation (2).

$$H_x = WH' + b \tag{2}$$

Here, $W \in \mathbb{R}^{(n+l)\times n}$ represents the learnable weight matrix, and $b \in \mathbb{R}^{n\times 1}$ is the learnable bias. Additionally, Xavier initialization is employed for the weight network to prevent gradient vanishing during backpropagation, ensuring consistency in the variances of inputs and outputs.

Finally, to capture the contextual information of the sequence and enhance the semantic feature representation H_x of sequence x, bidirectional Long Short-Term Memory networks (Bi-LSTMs) are employed. This step further constructs the input features $H \in \mathbb{R}^{n\times d_h}$, where d^h represents the dimensionality of a word representation, as shown in Equation (3).

$$H = BiLSTM(H_x) = [h_1, \cdots, h_n] \tag{3}$$

3.3. Grid Feature Extraction

The algorithm based on grid tagging requires the construction of grids and the extraction of grid features. This part includes grid representation fusion and two-dimensional feature extraction. Inspired by BERT's input structure, which includes position embedding, sentence tokenization, and the sentence sequence, three inputs are constructed to extract grid features: relative distance embedding for word pairs $E^d \in \mathbb{R}^{n\times n\times d_{E^d}}$, upper-lower triangular area embedding $E^t \in \mathbb{R}^{n\times n\times d_{E^t}}$, and grid representation for word pairs $V \in \mathbb{R}^{n\times n\times d_h}$.

Specifically, for a grid $grid \notin \mathbb{R}^{n\times n}$, where each cell represents a word pair (w_i, w_j), their relative distance embedding is proportional to the distance. The embedding values increase with distance, with a distance of 1 embedded as 0, and for distances in the range [4, 7], the embedding is 2, and so on. Following this rule, the relative distance embedding E^d corresponding to the grid can be obtained, with a dimensionality of d_{E^d} for the distance embedding of a single grid.

As mentioned earlier, this algorithm's designed word-pair relationships include the Next Neighboring Word (NNW) and Tail-Head Word (THW) tags. Within a concept or entity, based on tagging characteristics, NNW tags only appear in the upper triangular area or on the diagonal of the grid, while THW tags only appear in the lower triangular area or on the diagonal. Therefore, distinguishing between the grid's upper and lower triangular areas is necessary. To achieve this, the upper-lower triangular area embedding feature E^t is designed, with a dimensionality of d_{E^t} for the embedding of a single grid's upper–lower triangular area.

After the aforementioned embedding process, a one-dimensional sequence of deeply fused semantic features $H \in \mathbb{R}^{n\times d_h}$ is obtained. It is necessary to transform this feature into word-pair representations $V \in \mathbb{R}^{n\times n\times d_h}$. That is, for an n-length sequence, the features of any word pair (w_i, w_j) need to be computed. The relationships between word pairs in the grid have directionality. In other words, one of the necessary conditions for the relationship (w_i, w_j) to be NNW is $1 \leq i \leq j \leq n$. To obtain high-quality grid features while considering the directionality between word pairs, Conditional Layer Normalization (CLN) networks are employed to model the representation $V_{ij} \in \mathbb{R}^{d_v}$ of the word pair (w_i, w_j), as shown in Equation (4).

$$V_{ij} = \gamma_{ij} \odot (\frac{h_j - \mu}{\sigma}) + \lambda_{ij} = CLN(h_i, h_j) \tag{4}$$

where $\gamma_{ij} = W_\alpha h_i + b_\alpha$ represents the gain parameters, and $\lambda_{ij} = W_\beta h_i + b_\beta$ represents the parameters for layer normalization. μ and σ are the mean and standard deviation of vector h_j, calculated as shown in Equations (5) and (6).

$$\mu = \frac{1}{d_h} \sum_{k=1}^{d_h} h_{jk} \tag{5}$$

$$\sigma = \sqrt{\frac{1}{d_h} \sum_{k=1}^{d_k} (h_{jk} - \mu)^2} \tag{6}$$

Concatenating the aforementioned three types of representational features, the fusion of grid representation features is achieved through a multilayer perceptron, resulting in the preliminary fused grid representation feature $C \in \mathbb{R}^{n \times n \times d_c}$, where d_c represents the feature dimension of a single grid, as shown in Equation (7).

$$C = MLP_1([V; E^d; E^t]) \tag{7}$$

To capture the interaction between word pairs at different distances and expand the receptive field, dilated convolution is employed to extract two-dimensional grid features. However, using a single dilated convolution faces the issue of the grid effect. To avoid losing local information and extracting irrelevant long-distance information, a hybrid dilated convolution $DConv_L$ with different dilation rates is employed, where L represents the dilation rate. The fused grid representation feature C is input into the $DConv_L$ network to extract two-dimensional features $Q^L \in \mathbb{R}^{n \times n \times d_q}$, as shown in Equation (8).

$$Q^L = GELU(DConv_L(C)) \tag{8}$$

By cascading different dilated convolutional layers with various dilation rates, we obtain preliminary two-dimensional grid features Q_0, where m represents the number of dilated convolutional networks. The subscript "0" indicates that these features have not yet been iteratively fused with the tag attention features. This process is represented in Equation (9), where d_{wp} represents the embedding dimension of word-pair grid representations.

$$Q_0 = [Q^{L_1}, \cdots, Q^{L_m}] \in \mathbb{R}^{n \times n \times d_{wp}} \tag{9}$$

3.4. Label Representation Embedding

Currently, the correct identification of the three types of mentions only occurs when all grid tagging within the mention are predicted. Therefore, embedding tag representations into the model, modeling relationships between tags, and modeling relationships between tags and words can enhance the model's recognition ability in discontinuous scenarios.

In other words, tag representation embedding requires the model to learn the attention of different tags on different word positions, perceiving the roles of different tags and organizing these roles as implicit knowledge. Introducing attention mechanisms to perceive the attention of each tag independently allows the model to learn relevant information in different representation subspaces, enriching the perceptual dimensions.

However, although the multi-head attention mechanism can perceive the roles of various labels from multiple perspectives, it cannot learn the sequence of the sequence. As mentioned above, the sequence order is inevitable for grid tags. Therefore, introducing Rotary Position Embedding [10] (RoPE) into the multi-head attention mechanism allows the sequence information to be retained. RoPE is an advanced form of position embedding that achieves excellent results by implementing relative position embedding in the form of absolute position embedding.

Specifically, the process of tag representation embedding modules is as follows: extracting tag features, introducing rotary position embedding to retain sequence information, obtaining tag attention features through a multi-head attention mechanism, iteratively integrating word-pair features and tag attention features, thus embedding tag representations into the model.

By using a linear network, the word-pair grid features Q_0 obtained from the above steps are mapped to the label space, extracting features for different labels T_p, where $1 \leq p \leq P$, and P represents the number of labels. The extracted features $T_p \in \mathbb{R}^{n \times n \times d_{tag}}$ are represented as follows in Equation (10).

$$T_p = W_p Q_0 + b_p \tag{10}$$

Here, $W_p \in \mathbb{R}^{d_{tag} \times d_q}$ and $b_p \in \mathbb{R}^{d_{tag}}$ are learnable parameters where d_{tag} represents the dimensionality of individual tag features. Concatenating these individual tag features results in the word-pair grid $T = [T_1, \cdots, T_P] \in \mathbb{R}^{n \times n \times (P \cdot d_{tag})}$ mapped to the tag space.

Through the MaxPooling layers $MaxPool_1 \in \mathbb{R}^{n \times (P \cdot d_{tag})}$ and $MaxPool_2 \in \mathbb{R}^{n \times (P \cdot d_{tag})}$, word-pair grid features in both horizontal and vertical directions are obtained: $T_{sub} \in \mathbb{R}^{n \times (P \cdot d_{tag})}$, $T_{obj} \in \mathbb{R}^{n \times (P \cdot d_{tag})}$. These serve as query, key, and value vectors for the multi-head attention mechanism.

Rotary Position Embedding (RoPE) is introduced into the query vector q and key-value vector v. The core idea is to perform the $f(\cdot)$ operation on them separately, ensuring that the equation $< f(q,a), f(k,b) >= g(q,k,a-b)$ holds in the form of absolute positions implementing relative position embedding. The two-dimensional form of $f(\ )$ is as follows in Equation (11), where $\theta = 10,000^{-\frac{2i}{d}}$, and this can be extended to multiple dimensions accordingly, where i corresponds to the index.

$$f(q,a) = \begin{pmatrix} \cos a\theta & -\sin a\theta \\ \sin a\theta & \cos a\theta \end{pmatrix} \begin{pmatrix} q_0 \\ q_1 \end{pmatrix} \tag{11}$$

Rotary Position Embedding (RoPE) is introduced separately in T_{sub} and T_{obj} to obtain ${T_{sub}}'$ and ${T_{obj}}'$ There are three optional methods, as shown in Equations (11)–(13).

$$f(q,a)_{add} = \left(\begin{pmatrix} \cos a\theta & -\sin a\theta \\ \sin a\theta & \cos a\theta \end{pmatrix} + q \right) \begin{pmatrix} q_0 \\ q_1 \end{pmatrix} \tag{12}$$

$$f(q,a)_{mul} = \left(\begin{pmatrix} \cos a\theta & -\sin a\theta \\ \sin a\theta & \cos a\theta \end{pmatrix} \times q \right) \begin{pmatrix} q_0 \\ q_1 \end{pmatrix} \tag{13}$$

$$f(q,a)_{zero} = \begin{pmatrix} \cos a\theta & -\sin a\theta \\ \sin a\theta & \cos a\theta \end{pmatrix} \begin{pmatrix} q_0 \\ q_1 \end{pmatrix} \tag{14}$$

Subsequently, the tag attention features are obtained using the multi-head attention mechanism, where ${T_{sub}}'$ serves as the query vector and ${T_{obj}}'$ serves as the key and value vectors. Let the embedding dimension of each attention head be d_{head}, and there are hn heads in total. The specific calculation process is as follows:

First, for each attention head $head_{hi}, hi \in [1, hn]$, calculate the corresponding attention weights. Map the query vector and the key-value vector to their respective linear spaces using the corresponding linear networks, as shown in Equation (15), where $W_Q \in \mathbb{R}^{(Pd_{tag}) \times d_Q}$, $W_K \in \mathbb{R}^{(Pd_{tag}) \times d_K}$, and $W_V \in \mathbb{R}^{(Pd_{tag}) \times d_V}$ are learnable parameter matrices.

$$\begin{aligned} Q_{head_{hi}} &= {T_{sub}}' W_Q \\ K_{head_{hi}} &= {T_{obj}}' W_K \\ V_{head_{hi}} &= {T_{obj}}' W_V \end{aligned} \tag{15}$$

Then, calculate attention cores using function $s(\cdot)$, and finally, and obtain the attention $Att_{head_{hi}} \in \mathbb{R}^{n\times(Pd_{tag})\times d_{head}}$ for $head_{hi}$ through weighted summation, as shown in Equation (16).

$$Att_{head_{hi}} = \sum_{j=1}^{n} \alpha_{ij} v_{head_{hi},j} = \sum_{j=1}^{n} soft\max(s(k_{head_{hi},j}, q_{head_{hi},i}))v_{head_{hi},j} \tag{16}$$

The attention score function $s(\ )$, as shown in Equation (17), employs a scaled dot-product model.

$$s(k_{head_{hi}}, q_{head_{hi}}) = {k_{head_{hi}}}^{T} q_{head_{hi}} / \sqrt{(P \bullet d_{tag})} \tag{17}$$

Next, the attention weights from hn heads are concatenated and undergo another linear transformation, resulting in the tag-aware representation vector $T_{awared} \in \mathbb{R}^{n\times(Pd_{tag})\times d_{att}}$, as shown in Equation (18), where $W_o \in \mathbb{R}^{(hnd_{head})\times d_{att}}$ is a learnable parameter matrix.

$$T_{awared} \in \mathbb{R}^{n\times(P\bullet d_{tag})\times d_{att}} = concat(Att_{head_{h1}}, \cdots, Att_{head_{hn}})W_o \tag{18}$$

Finally, to embed the tag attention vectors into the model and integrate them deeply with the word-pair grid features, the two-dimensional tag attention features $T^{2D}_{awared} \in \mathbb{R}^{n\times n\times(Pd_{tag})}$ are reconstructed through the Conditional Layer Normalization network CLN_1, as shown in Equation (19), where the calculation method of the Conditional Layer Normalization network has been described earlier.

$$T^{2D}_{awared} = CLN_1(T_{awared}, T_{awared}) \tag{19}$$

The final tag-embedded word-pair grid features $Q_1 \in \mathbb{R}^{n\times n\times d_{wp}}$ are obtained using element-wise summation of the word-pair grid features $T \in \mathbb{R}^{n\times n\times(P\cdot d_{tag})}$ and the tag attention features, followed by a linear network, as shown in Equation (20), where $W_{TT} \in \mathbb{R}^{(Pd_{tag})\times d_{wp}}$ represents the learnable parameters.

$$Q_1 = (T^{2D}_{awared} \oplus T)W_{TT} \tag{20}$$

Furthermore, to deeply extract features between words, between words and tags, and between tags, the deep features are iteratively updated as shown in Equation (21).

$$Q_0 = Q_1 \tag{21}$$

Repeating multiple rounds of iterative calculations from Equations (9)–(21) enables the extraction of deeply embedded tag grid word-pair features Q_1.

3.5. Joint Prediction and Decoding

On top of predicting word-pair relationships using a multilayer perceptron, we incorporate a double affine predictor to enhance the classification capability. The calculation process of the double affine predictor is shown in Equation (22), where W, U, and b are learnable hyperparameters.

$$Biaffine(h_i, h_j) = (MLP_2(h_i))U(MLP_3(h_j)) + W[MLP_2(h_i); MLP_3(h_j)] + b \tag{22}$$

Combining the multilayer perceptron and the biaffine predictor, we obtain the relationship probability y_{ij} between word pairs $\left(w_i, w_j\right)$ as shown in Equation (23).

$$y_{ij} = soft\max(Biaffine(x_i, x_j) + MLP(Q_{1,ij})) \tag{23}$$

After obtaining the relationships between word pairs $\left(w_i, w_j\right)$ in the input sequence x, the flat, nested, and discontinuous concepts or entities are decoded in the grid. The decoding algorithm is illustrated in Figure 5.

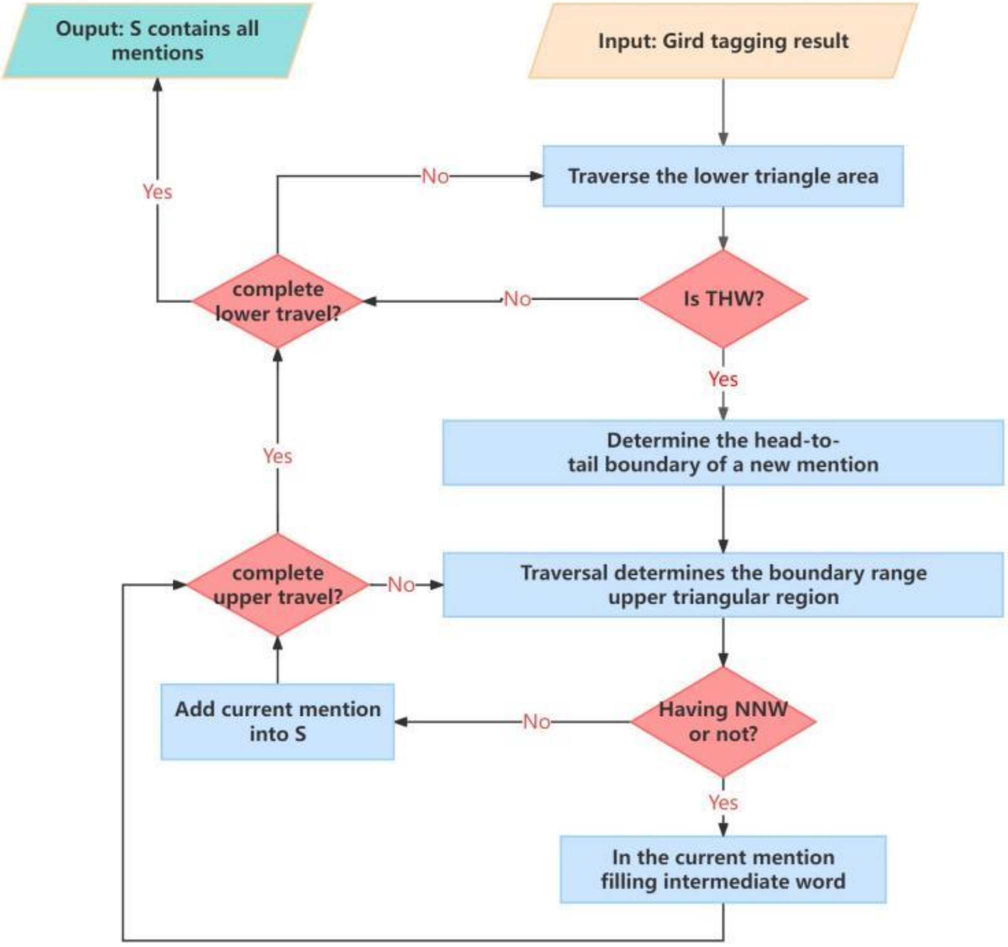

Ouput: S contains all mentions
Input: Gird tagging result
Traverse the lower triangle area
No
Yes
complete lower travel?
No
Is THW?
Yes
Determine the head-to-tail boundary of a new mention
Yes
complete upper travel?
No
Traversal determines the boundary range upper triangular region
Add current mention into S
No
Having NNW or not?
Yes
In the current mention filling intermediate word

Figure 5. The process of decoding the mentions in the tagging grid.

4. Experiment

4.1. Dataset

Datasets containing discontinuous entities and concepts were selected to evaluate the effectiveness of the model proposed in this paper. Currently, the only free and publicly available dataset meeting the requirements is CADEC. This English dataset comes from the biomedical field and includes only one entity type, Adverse Drug Reactions (ADRs). Discontinuous entities and concepts in this dataset constitute only 10% of the data. Additionally, to test the improved model's performance in discontinuous scenarios and its learning ability in situations with limited samples, all discontinuous data were extracted from the CADEC dataset to create the CADEC-Discontinuous dataset.

Furthermore, for the Chinese lexical resources mentioned in this paper, the Modern Chinese Dictionary compiled by the Institute of Linguistics, the Chinese Academy of Social Sciences, and the Chinese Dictionary jointly compiled by experts from multiple provinces were chosen. Five individuals annotated entities and concepts mentioned in the definitions of these Chinese dictionaries, and the corresponding annotations were determined through voting. To ensure the quality and reliability of annotation, we specifically invited five annotators with backgrounds in linguistics and related fields to participate in this task. These five annotators are all from the linguistics departments of well-known universities, and among them are scholars who have performed in-depth research in Chinese lexicology and semantics. They possess rich experience in areas such as compiling Chinese dictionaries and vocabulary teaching and providing strong assurance for the professionalism and accuracy of this annotation task. In detail, we collected all entities and concepts annotated by these five annotators and had them rate entities and concepts based on five options:

"Completely Agree", "Agree", "Don't Know", "Disagree", and "Completely Disagree". If any annotator was unfamiliar with any term in the target vocabulary, they could choose "Don't Know". Only entities and concepts that received at least three "Agree" (or "Completely Agree") votes were considered in the dataset.

The statistical data for the datasets used in this paper are shown in Table 1 below.

Table 1. The statistical data of the datasets.

Dataset	**CADEC**				**CADEC-Discontinuous**				**Chinese Dictionary**			
	All	**Train**	**Dev**	**Test**	**All**	**Train**	**Dev**	**Test**	**All**	**Train**	**Dev**	**Test**
#Sentences	7597	5340	1097	1160	439	306	59	74	1496	1036	230	230
#Entities	6318	4430	898	990	679	491	94	94	4268	2916	639	713
#Discontinuous	679	491	94	94	679	491	94	94	253	172	46	35
%Discontinuous	10.7	11.1	10.5	9.5	100	100	100	100	16.9	16.6	20.0	15.2

The "#" represents the number of statistical data. The "%" represents the percentage of the total statistics.

4.2. Evaluation Metrics

The typical evaluation metrics for assessing model performance include precision (P), which evaluates the ratio of correctly predicted positive observations to the total predicted positive observations; recall (R), which assesses the proportion of actual positive cases that were correctly predicted; and the composite metric F1 score (F1). Precision and recall are conflicting metrics. The confusion matrix divides the different outcomes of classification, as shown in Table 2.

Table 2. The confusion matrix of classification results.

Predict **True**	**Positive Samples**	**Negative Samples**
Positive samples	*TP*	*FN*
Negative samples	*FP*	*TN*

The formulas for the above-mentioned metrics are as follows:

$$\begin{aligned} P &= \tfrac{TP}{TP+FN} \\ R &= \tfrac{TP}{TP+FP} \\ F1 &= \tfrac{2 \bullet P \bullet R}{P+R} \end{aligned} \tag{24}$$

4.3. Experimental Results and Analysis

In recent years, significant methods for the unified recognition of three types of entities and concepts include methods based on sequence labeling, methods based on hypergraphs, methods based on sequence-to-sequence, and methods based on grid tagging. The sequence labeling methods assign a tag to each token, such as the LSTM-CRF model based on the BIOHD tagging method proposed by Tang et al. [21] in 2018. The hypergraph-based methods use hypergraphs to represent and infer mentions of different types in the text, as exemplified by the model proposed by Wang et al. [42] in 2019. Sequence-to-sequence methods directly generate word sequences of mentions at the decoder side, with typical examples being the model proposed by Fei et al. [43] in 2021. Grid-tagging methods, the core approach adopted in this study, construct the input sequence into a grid, assign tags to each pair of words in the grid, and then decode mentions (entities or concepts) in the input text based on these tags. Representative models include those proposed by Wang et al. [37] in 2021 and W^{2}NER proposed by Li et al. [3] in 2022.

Here, a total of four sets of experiments were conducted. The results of the first and second sets of experiments are shown in Table 3, the results of the third set of experiments are shown in Table 4, and the results of the fourth set of experiments are shown in Table 5:

1. On the public dataset CADEC, this study compares various typical models in recent years and validates the improvement effect of the proposed DTaE grid-tagging algorithm for unified entity and concept mining tasks.
2. On the Chinese Dictionary dataset, a comparison is made with the baseline model W^{2}NER.
3. The study validates the significant improvement of the proposed DTaE model in unified entity and concept recognition tasks.
4. On the public dataset CADEC, ablation experiments are conducted to verify the effectiveness of the introduced demonstration search module, multi-head attention mechanism, and rotary position embedding in the improved DTaE model.
5. On the non-sequential data from the public dataset CADEC, a comparison is made with the base model W^{2}NER to validate the effectiveness of the proposed DTaE grid-tagging algorithm in non-sequential scenarios and its capability to learn from fewer samples.

Table 3. Comparative experiments on CADEC and Chinese dictionary datasets.

Model		**CADEC**			**Chinese Dictionary**		
		P (%)	**R (%)**	**F1 (%)**	**P (%)**	**R (%)**	**F1 (%)**
Methods based on sequence labeling	Tang et al. 2018 [21]	67.80	64.99	66.36	/	/	/
Methods based on hypergraph	Wang and Lu 2019 [42]	72.10	48.40	58.00	/	/	/
Methods based on sequence-to-sequence	Fei et al. 2021 [43]	75.50	71.80	72.40	/	/	/
Methods based on grid tagging	Wang et al. 2021 [37]	70.50	**72.50**	71.50	67.59	75.97	71.53
	Li et al. 2022 [3]	74.09 (81.84)	72.35 **(83.80)**	73.21 (82.77)	86.75	86.38	86.56
	Ours Model	**76.87** **(83.24)**	71.52 (82.94)	**74.10** **(83.55)**	**89.72**	**88.73**	**89.22**

The bold number represents the highest result in each column. The bracket represents the token level. A script for calculating mention-level evaluation metrics on the CADEC dataset can be found in Supplementary Materials.

Table 4. Ablation experiments on public datasets.

Model	**Number of Iterations**	**Attention**	**RoPE**	**Demonstration**	**CADEC**		
					P (%)	**R (%)**	**F1 (%)**
W^{2}NER	1	/	1	/	74.09	**72.35**	73.21
Ours	6	Y	/	/	74.76	69.38	71.96
Ours	6	Y	Zero	/	76.39	70.91	73.55
Ours	6	Y	Add	/	75.70	68.24	71.14
Ours	6	Y	Mul	/	73.33	68.60	70.89
Ours	6	Y	Zero	Random	76.69	71.00	73.60
Ours	6	Y	Zero	SentenceBERT	**76.87**	71.52	**74.10**

The bold number represents the highest result in each column.

Table 5. Comparative experiments on discontinuous scenarios of public datasets.

Model		CADEC-Discontinuous		
		P (%)	**R (%)**	**F1 (%)**
Methods based on grid tagging.	W^{2}NER	48.68	63.79	39.36
	Ours Model	51.66	68.42	41.49

To validate the effectiveness of the proposed improved model for unified recognition tasks, comparisons were made with various types of typical algorithms on the public dataset CADEC. As shown in Table 3, the proposed improved model achieved the highest precision and F1 score among all typical models at the cost of a slightly lower recall rate. Compared to all typical models, the improved model in this study showed the highest precision and F1 score. In comparison to the base model W^{2}NER, the precision improved by 2.78%, and the overall F1 score increased by 0.89%. Furthermore, the performance of the improved model in recognizing the required data for the application scenario in this study was verified using the Chinese Dictionary dataset. Compared to W^{2}NER, this study's proposed DTaE grid-tagging model demonstrated improvements in three metrics: a 2.97% increase in precision, a 2.35% increase in recall, and a 2.66% increase in the overall F1 score. To further demonstrate the effectiveness of our improvement, Table 3 presents the token-level evaluation metrics on the CADEC dataset, as shown in the brackets. The core of the grid-tagging algorithm is to recognize the word-pair relationships on the grid and then obtain the textual mentions from input sentences through decoding. Our work has achieved significant improvements in recognizing word-pair relationships, which helps to enhance the effect of obtaining mentions through decoding in the entire sentence. However, local errors in recognizing word-pair relationships will affect the overall decoding effect to a certain extent. Therefore, the final performance shows a slighter improvement in mention-level metrics, which does not mean that our improvement is affected by data noise.

To comprehensively assess the trade-off between precision, recall, and F1 score, we conducted tests on the classification using the Receiver Operating Characteristic (ROC) curve. The Area Under the Receiver Operating Curve (AUC-ROC) is used to measure the performance of the classification. The ROC curve has a valuable characteristic: it remains unchanged when there is a change in the distribution of positive and negative samples in the test set. Lable imbalance, where there is a significant difference in the number of negative samples compared to positive samples (or vice versa), is common in real-world datasets. Additionally, the distribution of positive and negative samples in test data may vary over time.

The ROC-AUC curve for token-level classification of word-pair relationships is presented in Figure 6, a script for Figure 6 can be found in Supplementary Materials, where NNW represents the Next Neighboring Word, and THW-ADR stands for the tail headword of the ADR (Adverse Drug Reaction) entity. More explanations of labels can be found in the aforementioned text. We could find out that both NNW and THW-ADR reach a higher enough area (0.94 and 0.96). And the black dashed line on the diagonal represents the fiducial. It is suggested that our work does improve the token-level performance instead of noise in the evaluation data. This aligns with the characteristics of the grid-tagging method, where both public datasets and our private dictionary exhibit high accuracy at the token level. In summary, our work also maintains this excellent feature.

To validate the effectiveness of the three innovations introduced in this study, ablation experiments were conducted on the public dataset CADEC. As shown in Table 4, embedding label attention features into the model and iteratively merging word-pair grid features led to a 0.67% increase in precision (P), but it resulted in a significant impact on recall due to the loss of sequential information. Furthermore, the introduction of rotary position embedding improved precision (P) by 2.3% and increased F1 score by 0.34%. Additionally, utilizing SBERT semantic search demonstration sentences as input for information enhancement further improved the precision (P) and F1 score. The proposed model achieved the best metrics, as indicated in the last row of the table.

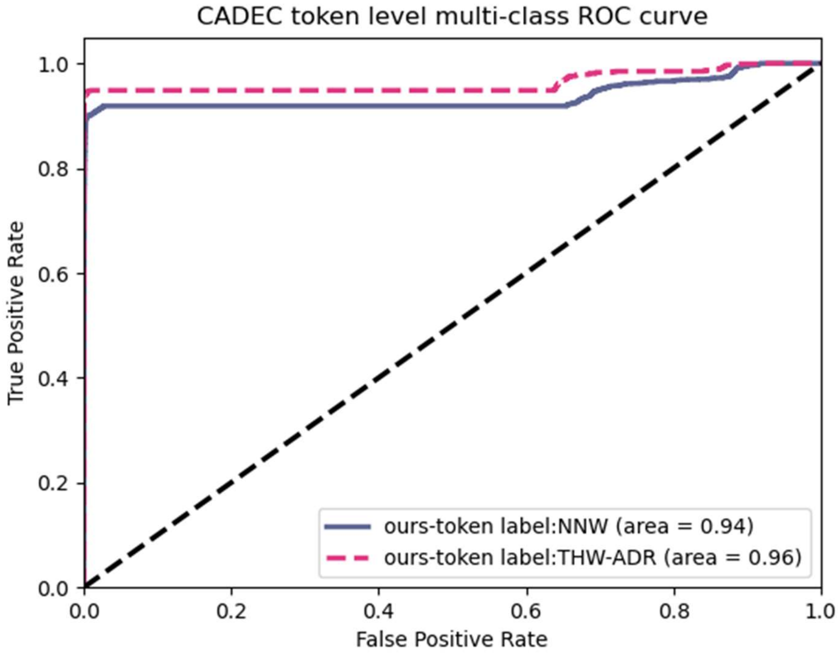


Figure 6. ROC-AUC curve for predicting grid labels on the CADEC dataset.

To rigorously evaluate the proposed model's ability to learn from few-shot and its improvement in recognizing discontinuous scenarios, the flat and nested parts of the data in the public dataset CADEC were removed, retaining only the discontinuous data. The statistics for the CADEC-Discontinuous dataset are shown in Table 1. Based on the experimental results in Table 5, it is evident that the proposed model has achieved the improvement objectives. Compared to the base model, the proposed model demonstrated a 2.98% increase in precision (P), a 4.63% increase in recall (R), and a 2.13% increase in F1 score. The differences in F1 scores among different models on the CADEC-Discontinuous dataset are illustrated in Figure 7.

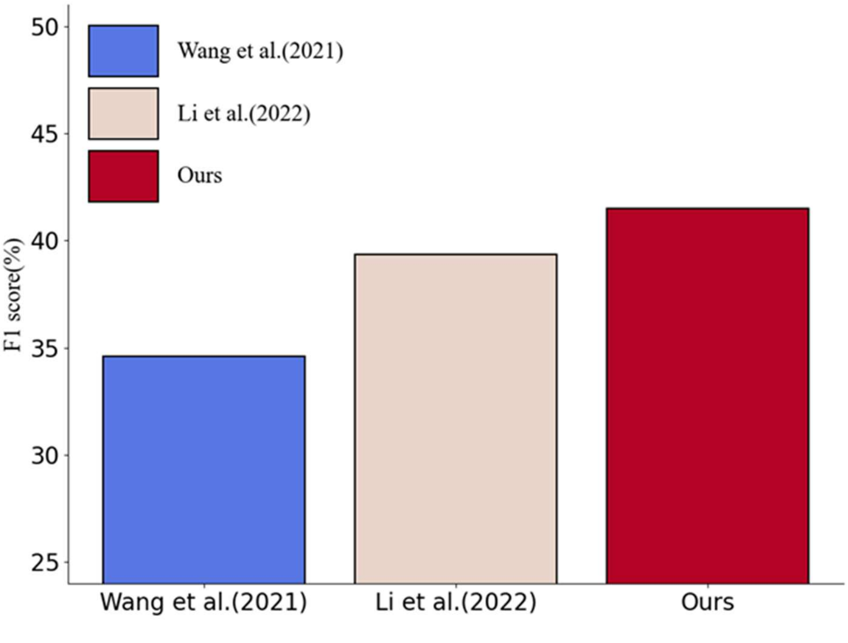


Figure 7. The differences in F1 scores among different models on the CADEC-Discontinuous [3,37].

Additionally, concerning the attention mechanism introduced in this study, attention heatmaps were generated on the CADEC dataset to analyze their effectiveness. For instance, in the sentence "I developed a pain that seems to originate at the base of my neck, travel down to my shoulder blades and then run to my shoulders, elbow, and wrist", there are discontinuous mentions of ADR types such as "pain neck", "pain shoulders", "pain elbow", and "pain wrist". The average attention scores of the three heads for these mentions are shown in Figure 8. Average attention scores between each word pair in the two-dimensional grid formed by the example sentence are shown in Figure 9. It is clear that the colors of grids around each mention are warm, and the attention scores are also high, with an average of around 0.75, shown in the yellow highlighted part of Figure 9.

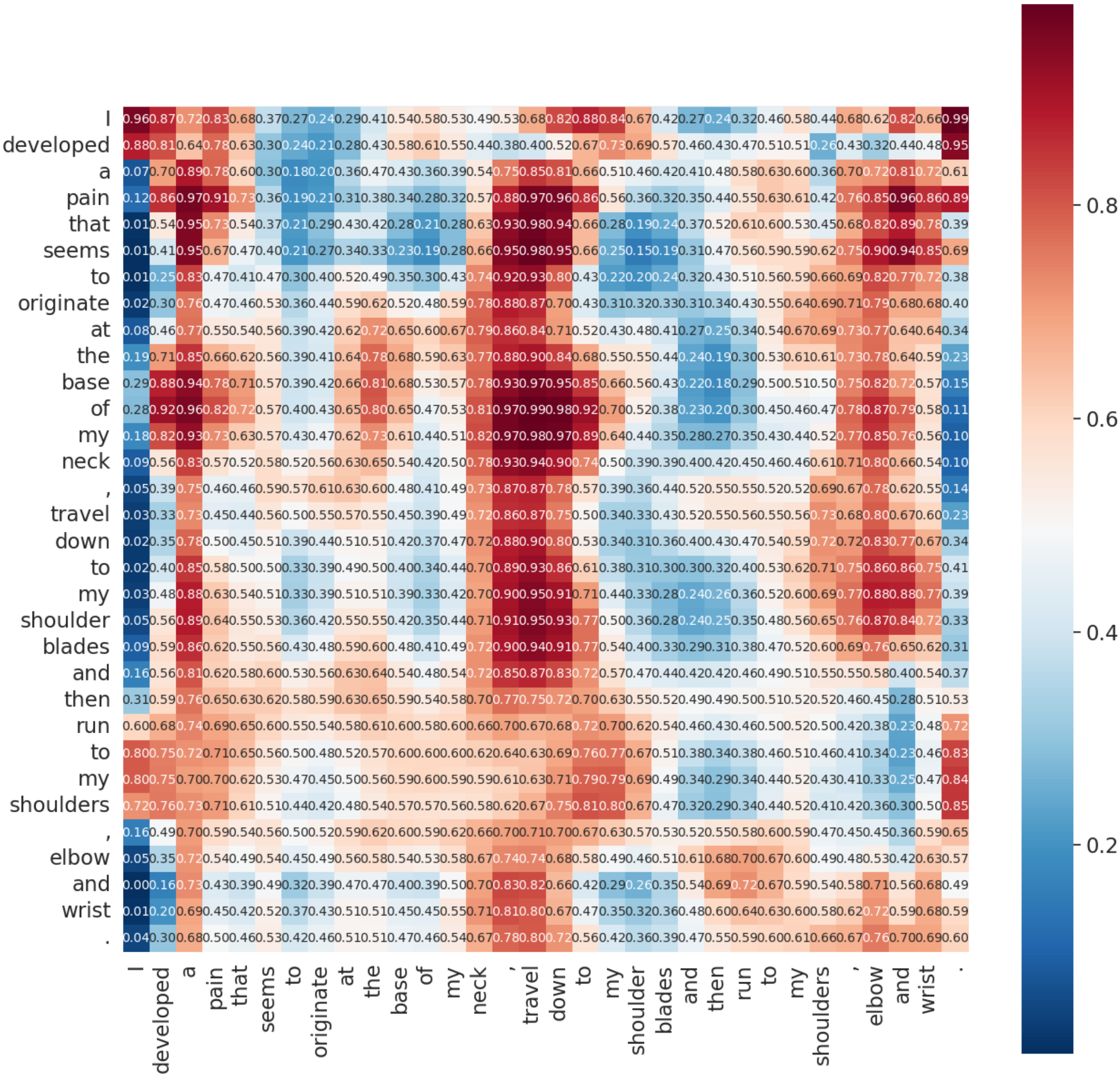
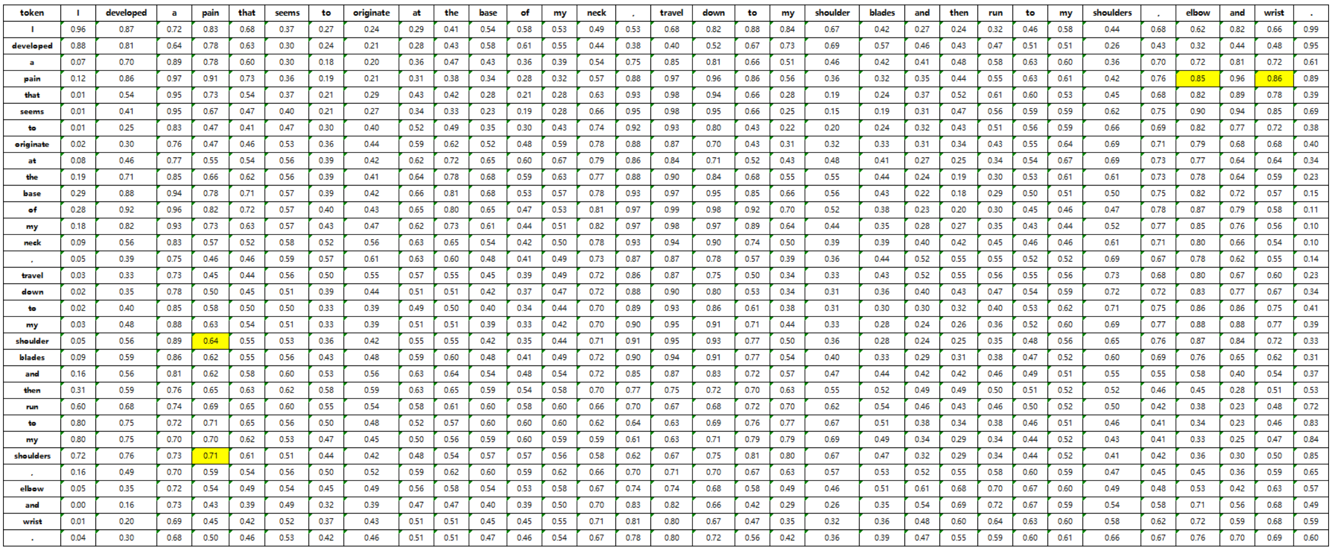


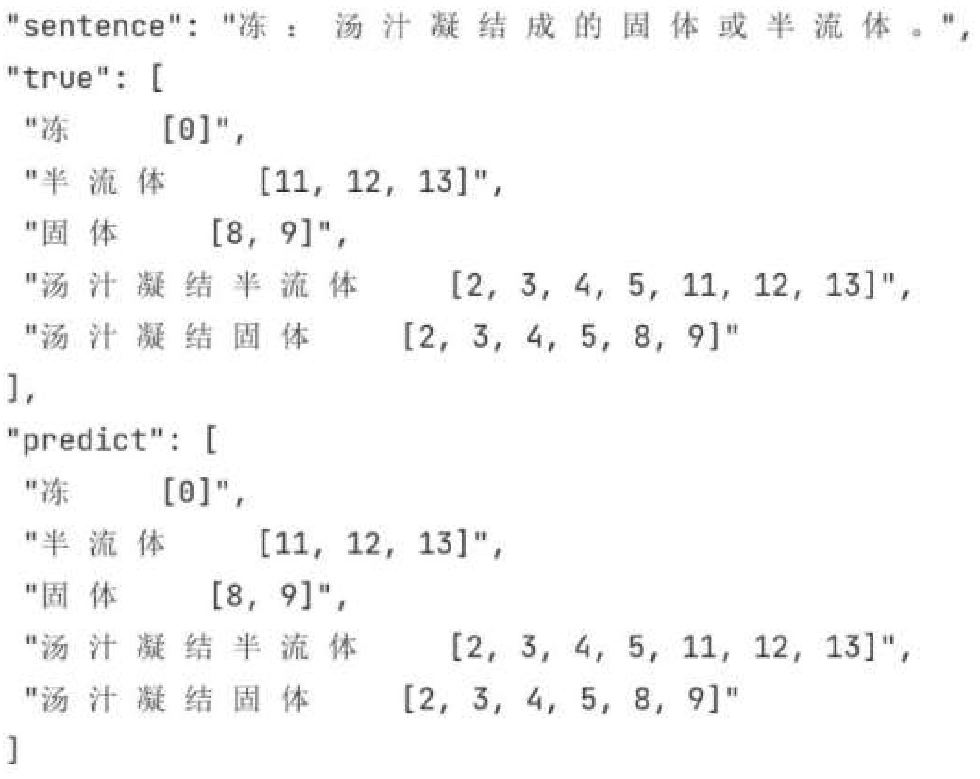
Figure 8. The heatmap of attention weight from an instance in CADEC, where warmer color means higher weight and vice versa for the cool color.



token	I	developed	a	pain	that	seems	to	originate	at	the	base	of	my	neck	,	travel	down	to	my	shoulder	blades	and	then	run	to	my	shoulders	,	elbow	and	wrist	.
I	0.96	0.87	0.72	0.83	0.68	0.37	0.27	0.24	0.29	0.41	0.54	0.58	0.53	0.49	0.53	0.68	0.82	0.88	0.84	0.67	0.42	0.27	0.24	0.32	0.46	0.58	0.44	0.68	0.62	0.82	0.66	0.99
developed	0.88	0.81	0.64	0.78	0.63	0.30	0.24	0.21	0.28	0.43	0.58	0.61	0.55	0.44	0.38	0.40	0.52	0.67	0.73	0.69	0.57	0.46	0.43	0.47	0.51	0.51	0.26	0.43	0.32	0.44	0.48	0.95
a	0.07	0.70	0.89	0.78	0.60	0.30	0.18	0.20	0.36	0.47	0.43	0.36	0.39	0.54	0.75	0.85	0.81	0.66	0.51	0.46	0.42	0.41	0.48	0.58	0.63	0.60	0.36	0.70	0.72	0.81	0.72	0.61
pain	0.12	0.86	0.97	0.91	0.73	0.36	0.19	0.21	0.31	0.38	0.34	0.28	0.32	0.57	0.88	0.97	0.96	0.86	0.56	0.36	0.32	0.35	0.44	0.55	0.63	0.61	0.42	0.76	0.85	0.96	0.86	0.89
that	0.01	0.54	0.95	0.73	0.54	0.37	0.21	0.29	0.43	0.42	0.28	0.21	0.28	0.63	0.93	0.98	0.94	0.66	0.28	0.19	0.24	0.37	0.52	0.61	0.60	0.53	0.45	0.68	0.82	0.89	0.78	0.39
seems	0.01	0.41	0.95	0.67	0.47	0.40	0.21	0.27	0.34	0.33	0.23	0.19	0.28	0.66	0.95	0.98	0.95	0.66	0.25	0.15	0.19	0.31	0.47	0.56	0.59	0.59	0.62	0.75	0.90	0.94	0.85	0.69
to	0.01	0.25	0.83	0.47	0.41	0.47	0.30	0.40	0.52	0.49	0.35	0.30	0.43	0.74	0.92	0.93	0.80	0.43	0.22	0.20	0.24	0.32	0.43	0.51	0.56	0.59	0.66	0.69	0.82	0.77	0.72	0.38
originate	0.02	0.30	0.76	0.47	0.46	0.53	0.36	0.44	0.59	0.62	0.52	0.48	0.59	0.78	0.88	0.87	0.70	0.43	0.31	0.32	0.33	0.31	0.34	0.43	0.55	0.64	0.69	0.71	0.79	0.68	0.68	0.40
at	0.08	0.46	0.77	0.55	0.54	0.56	0.39	0.42	0.62	0.72	0.65	0.60	0.67	0.79	0.86	0.84	0.71	0.52	0.43	0.48	0.41	0.27	0.25	0.34	0.54	0.67	0.69	0.73	0.77	0.64	0.64	0.34
the	0.19	0.71	0.85	0.66	0.62	0.56	0.39	0.41	0.64	0.78	0.68	0.59	0.63	0.77	0.88	0.90	0.84	0.68	0.55	0.55	0.44	0.24	0.19	0.30	0.53	0.61	0.61	0.73	0.78	0.64	0.59	0.23
base	0.29	0.88	0.94	0.78	0.71	0.57	0.39	0.42	0.66	0.81	0.68	0.53	0.57	0.78	0.93	0.97	0.95	0.85	0.66	0.56	0.43	0.22	0.18	0.29	0.50	0.51	0.50	0.75	0.82	0.72	0.57	0.15
of	0.28	0.92	0.96	0.82	0.72	0.57	0.40	0.43	0.65	0.80	0.65	0.47	0.53	0.81	0.97	0.99	0.98	0.92	0.70	0.52	0.38	0.23	0.20	0.30	0.45	0.46	0.47	0.78	0.87	0.79	0.58	0.11
my	0.18	0.82	0.93	0.73	0.63	0.57	0.43	0.47	0.62	0.73	0.61	0.44	0.51	0.82	0.97	0.98	0.97	0.89	0.64	0.44	0.35	0.28	0.27	0.35	0.43	0.44	0.52	0.77	0.85	0.76	0.56	0.10
neck	0.09	0.56	0.83	0.57	0.52	0.58	0.52	0.56	0.63	0.65	0.54	0.42	0.50	0.78	0.93	0.94	0.90	0.74	0.50	0.39	0.39	0.40	0.42	0.45	0.46	0.46	0.61	0.71	0.80	0.66	0.54	0.10
,	0.05	0.39	0.75	0.46	0.46	0.59	0.57	0.61	0.63	0.60	0.48	0.41	0.49	0.73	0.87	0.87	0.78	0.57	0.39	0.36	0.44	0.52	0.55	0.55	0.52	0.52	0.69	0.67	0.78	0.62	0.55	0.14
travel	0.03	0.33	0.73	0.45	0.44	0.56	0.50	0.55	0.57	0.55	0.45	0.39	0.49	0.72	0.86	0.87	0.75	0.50	0.34	0.33	0.43	0.52	0.55	0.56	0.55	0.56	0.73	0.68	0.80	0.67	0.60	0.23
down	0.02	0.35	0.78	0.50	0.45	0.51	0.39	0.44	0.51	0.51	0.42	0.37	0.47	0.72	0.88	0.90	0.80	0.53	0.34	0.31	0.36	0.40	0.43	0.47	0.54	0.59	0.72	0.72	0.83	0.77	0.67	0.34
to	0.02	0.40	0.85	0.58	0.50	0.50	0.33	0.39	0.49	0.50	0.40	0.34	0.44	0.70	0.89	0.93	0.86	0.61	0.38	0.31	0.30	0.30	0.32	0.40	0.53	0.62	0.71	0.75	0.86	0.86	0.75	0.41
my	0.03	0.48	0.88	0.63	0.54	0.51	0.33	0.39	0.51	0.51	0.39	0.33	0.42	0.70	0.90	0.95	0.91	0.71	0.44	0.33	0.28	0.24	0.26	0.36	0.52	0.60	0.69	0.77	0.88	0.88	0.77	0.39
shoulder	0.05	0.56	0.89	0.64	0.55	0.53	0.36	0.42	0.55	0.55	0.42	0.35	0.44	0.71	0.91	0.95	0.93	0.77	0.50	0.36	0.28	0.24	0.25	0.35	0.48	0.56	0.65	0.76	0.87	0.84	0.72	0.33
blades	0.09	0.59	0.86	0.62	0.55	0.56	0.43	0.48	0.59	0.60	0.48	0.41	0.49	0.72	0.90	0.94	0.91	0.77	0.54	0.40	0.33	0.29	0.31	0.38	0.47	0.52	0.60	0.69	0.76	0.65	0.62	0.31
and	0.16	0.56	0.81	0.62	0.58	0.60	0.53	0.56	0.63	0.64	0.54	0.48	0.54	0.72	0.85	0.87	0.83	0.72	0.57	0.47	0.44	0.42	0.42	0.46	0.49	0.51	0.55	0.55	0.58	0.40	0.54	0.37
then	0.31	0.59	0.76	0.65	0.63	0.62	0.58	0.59	0.63	0.65	0.59	0.54	0.58	0.70	0.77	0.75	0.72	0.70	0.63	0.55	0.52	0.49	0.49	0.50	0.51	0.52	0.52	0.46	0.45	0.28	0.51	0.53
run	0.60	0.68	0.74	0.69	0.65	0.60	0.55	0.54	0.58	0.61	0.60	0.58	0.60	0.66	0.70	0.67	0.68	0.72	0.70	0.62	0.54	0.46	0.43	0.46	0.50	0.52	0.50	0.42	0.38	0.23	0.48	0.72
to	0.80	0.75	0.72	0.71	0.65	0.56	0.50	0.48	0.52	0.57	0.60	0.60	0.60	0.62	0.64	0.63	0.69	0.76	0.77	0.67	0.51	0.38	0.34	0.38	0.46	0.51	0.46	0.41	0.34	0.23	0.46	0.83
my	0.80	0.75	0.70	0.70	0.62	0.53	0.47	0.45	0.50	0.56	0.59	0.60	0.59	0.59	0.61	0.63	0.71	0.79	0.79	0.69	0.49	0.34	0.29	0.34	0.44	0.52	0.43	0.41	0.33	0.25	0.47	0.84
shoulders	0.72	0.76	0.73	0.71	0.61	0.51	0.44	0.42	0.48	0.54	0.57	0.57	0.56	0.58	0.62	0.67	0.75	0.81	0.80	0.67	0.47	0.32	0.29	0.34	0.44	0.52	0.41	0.42	0.36	0.30	0.50	0.85
,	0.16	0.49	0.70	0.59	0.54	0.56	0.50	0.52	0.59	0.62	0.60	0.59	0.62	0.66	0.70	0.71	0.70	0.67	0.63	0.57	0.53	0.52	0.55	0.58	0.60	0.59	0.47	0.45	0.45	0.36	0.59	0.65
elbow	0.05	0.35	0.72	0.54	0.49	0.54	0.45	0.49	0.56	0.58	0.54	0.53	0.58	0.67	0.74	0.74	0.68	0.58	0.49	0.46	0.51	0.61	0.68	0.70	0.67	0.60	0.49	0.48	0.53	0.42	0.63	0.57
and	0.00	0.16	0.73	0.43	0.39	0.49	0.32	0.39	0.47	0.47	0.40	0.39	0.50	0.70	0.83	0.82	0.66	0.42	0.29	0.26	0.35	0.54	0.69	0.72	0.67	0.59	0.54	0.58	0.71	0.56	0.68	0.49
wrist	0.01	0.20	0.69	0.45	0.42	0.52	0.37	0.43	0.51	0.51	0.45	0.45	0.55	0.71	0.81	0.80	0.67	0.47	0.35	0.32	0.36	0.48	0.60	0.64	0.63	0.60	0.58	0.62	0.72	0.59	0.68	0.59
.	0.04	0.30	0.68	0.50	0.46	0.53	0.42	0.46	0.51	0.51	0.47	0.46	0.54	0.67	0.78	0.80	0.72	0.56	0.42	0.36	0.39	0.47	0.55	0.59	0.60	0.61	0.66	0.67	0.76	0.70	0.69	0.60

Figure 9. The table of attention weight from an instance in CADEC.

In summary, the proposed improved model DTaE grid-tagging algorithm, without compromising the unified entity/concept recognition effectiveness, introduced the multi-head attention mechanism and rotary position embedding, further enhancing the recognition performance in discontinuous scenarios. Simultaneously, inspired by the in-context learning concept in large language models, the introduction of task demonstrations enhanced the model's ability to learn from a few shots. In the Chinese Dictionary dataset, out of 230 test instances, the model accurately predicted fine-grained entities/concepts in 196 glosses, as depicted in Figure 10, demonstrating the practical predictive effectiveness of the improved model on the Chinese Dictionary data.



```
"sentence": "冻 ： 汤 汁 凝 结 成 的 固 体 或 半 流 体 。",
"true": [
 "冻     [0]",
 "半 流 体     [11, 12, 13]",
 "固 体     [8, 9]",
 "汤 汁 凝 结 半 流 体     [2, 3, 4, 5, 11, 12, 13]",
 "汤 汁 凝 结 固 体     [2, 3, 4, 5, 8, 9]"
],
"predict": [
 "冻     [0]",
 "半 流 体     [11, 12, 13]",
 "固 体     [8, 9]",
 "汤 汁 凝 结 半 流 体     [2, 3, 4, 5, 11, 12, 13]",
 "汤 汁 凝 结 固 体     [2, 3, 4, 5, 8, 9]"
]
```

Figure 10. Example of our model recognizing mentions on Chinese Dictionary data.

5. Conclusions

This paper aims to enhance the model's ability to learn from few shots and improve its recognition capabilities in non-sequential scenarios based on the state-of-the-art (SOTA) model W^{2}NER for the unified recognition of flat, nested, and discontinuous mentions. Three improvements were made to the grid-tagging algorithm W^{2}NER, leading to the proposed Demonstration and Tag-aware Enhanced Grid-Tagging Network (DTaE) model. Specifically, this study introduced task demonstrations as a form of information enhancement during the training phase. The model incorporated tag attention features obtained through the multi-head attention mechanism and preserved sequence position information using rotary position embedding. The model iteratively merged word-pair grid features and tag attention features, embedding tag attention into the model. Through multiple rounds of experiments, the effectiveness of the proposed model was validated on the public dataset CADEC and the Chinese Dictionary dataset annotated in this study.

6. Limitation and Future Work

Our current research is constrained by hardware computational capabilities, confined to relatively smaller pre-training models such as BERT and its variants. However, the emergence of large language models with hundreds of billions of parameters has showcased powerful capabilities not present in traditional pre-training models. More and more efforts are being made to incorporate these large language models. Future work may consider employing language models with a greater number of parameters as our word-embedding tools to better capture semantic information. Additionally, with the development of large language models, prompt methods akin to In-Context Learning have seen significant progress. In the future, exploring In-Context Learning methods and Chain-of-Thought approaches more tailored to large language models as replacements for demonstrations, aiming to achieve enhanced performance, may be worthwhile. Additionally, there is typically a precision gap between the token level and the mention level for grid-tagging methods and similar approaches. Some experiments have shown that the precision of the token level tends to be higher than that of the mention level. Currently, this method lacks a systematic explanation of its characteristics and effective improvement methods. In future work, we will explore ways to enhance mention-level precision on the decoding side to reduce the gap with token-level precision.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/electronics13020261/s1, We provide two main scripts: a script for calculating mention-level evaluation metrics on the CADEC dataset, and a script for constructing ROC-AUC curves at the token level for multi-class scenarios on the CADEC dataset.

Author Contributions: Conceptualization, H.J., J.H. and H.Z.; Methodology, J.H.; Software, J.H.; Validation, J.H.; Formal analysis, K.Z. and Y.M.; Investigation, H.J.; Resources, H.J.; Data curation, K.Z. and Y.M.; Writing—original draft, H.J. and J.H.; Writing—review & editing, K.Z. and Y.M.; Visualization, K.Z. and Y.M.; Supervision, H.Z., L.R., Y.J. and W.X.; Project administration, H.Z., L.R., Y.J. and W.X.; Funding acquisition, H.J., H.Z. and W.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Sichuan Science and Technology Program (under Grants No. 2022YFQ0052) and the Technology Development Project of China Railway Eryuan Engineering Group Co., Ltd. (under Grant No. KYY2019003(19-22)).

Data Availability Statement: Publicly available datasets were analyzed in this study. The CADEC dataset can be found here: https://pubmed.ncbi.nlm.nih.gov/25817970/. Our research is supported by the aforementioned funding. Due to the confidentiality requirements of the institution, we are unable to provide Chinese dictionary data.

Conflicts of Interest: Author Huanlai Zhou was employed by the company UESTC—Chengdu Quantum Matrix Technology Co., Ltd. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Murphy, G. *The Big Book of Concepts*; MIT Press: Cambridge, MA, USA, 2004.
2. Yan, H.; Gui, T.; Dai, J.; Guo, Q.; Zhang, Z.; Qiu, X. A unified generative framework for various NER subtasks. *arXiv* **2021**, arXiv:2106.01223.
3. Li, J.; Fei, H.; Liu, J.; Wu, S.; Zhang, M.; Teng, C.; Ji, D.; Li, F. Unified named entity recognition as word-word relation classification. In Proceedings of the AAAI Conference on Artificial Intelligence, Virtual, 22 February–1 March 2022.
4. Hu, Y.; He, H.; Chen, Z.; Zhu, Q.; Zheng, C. A unified model using distantly supervised data and cross-domain data in NER. *Comput. Intell. Neurosci.* **2022**, *2022*, 1987829. [CrossRef] [PubMed]
5. Lu, J.; Zhao, R.; Mac Namee, B.; Tan, F. Punifiedner: A prompting-based unified ner system for diverse datasets. In Proceedings of the AAAI Conference on Artificial Intelligence, Washington, DC, USA, 7–14 February 2023.
6. Zhang, S.; Shen, Y.; Tan, Z.; Wu, Y.; Lu, W. De-bias for generative extraction in unified NER task. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Long Papers. Volume 1, pp. 808–818.
7. Liu, J.; Ji, D.; Li, J.; Xie, D.; Teng, C.; Zhao, L.; Li, F. TOE: A grid-tagging discontinuous NER model enhanced by embedding tag/word relations and more fine-grained tags. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2022**, *31*, 177–187. [CrossRef]
8. Min, S.; Lyu, X.; Holtzman, A.; Artetxe, M.; Lewis, M.; Hajishirzi, H.; Zettlemoyer, L. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv* **2022**, arXiv:2202.12837.
9. Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A. Language models are few-shot learners. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 1877–1901.
10. Su, J.; Lu, Y.; Pan, S.; Murtadha, A.; Wen, B.; Liu, Y. Roformer: Enhanced transformer with rotary position embedding. *arXiv* **2021**, arXiv:2104.09864. [CrossRef]
11. Karimi, S.; Metke-Jimenez, A.; Kemp, M.; Wang, C. CADEC: A corpus of adverse drug event annotations. *J. Biomed. Inform.* **2015**, *55*, 73–81. [CrossRef] [PubMed]
12. Wang, Z.; Xu, X.; Li, X.; Li, H.; Wei, X.; Huang, D. An Improved Nested Named-Entity Recognition Model for Subject Recognition Task under Knowledge Base Question Answering. *Appl. Sci.* **2023**, *13*, 11249. [CrossRef]
13. Huang, P.; Zhao, X.; Hu, M.; Tan, Z.; Xiao, W. T 2-NER: AT wo-Stage Span-Based Framework for Unified Named Entity Recognition with T emplates. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1265–1282. [CrossRef]
14. Jiang, X.; Song, C.; Xu, Y.; Li, Y.; Peng, Y. Research on sentiment classification for netizens based on the BERT-BiLSTM-TextCNN model. *PeerJ Comput. Sci.* **2022**, *8*, e1005. [CrossRef]
15. Tang, X.; Huang, Y.; Xia, M.; Long, C. A Multi-Task BERT-BiLSTM-AM-CRF Strategy for Chinese Named Entity Recognition. *Neural Process. Lett.* **2023**, *55*, 1209–1229. [CrossRef]
16. Li, D.; Yan, L.; Yang, J.; Ma, Z. Dependency syntax guided bert-bilstm-gam-crf for chinese ner. *Expert Syst. Appl.* **2022**, *196*, 116682. [CrossRef]
17. Li, W.; Du, Y.; Li, X.; Chen, X.; Xie, C.; Li, H.; Li, X. UD_BBC: Named entity recognition in social network combined BERT-BiLSTM-CRF with active learning. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105460. [CrossRef]
18. Zhang, W.; Meng, J.; Wan, J.; Zhang, C.; Zhang, J.; Wang, Y.; Xu, L.; Li, F. ChineseCTRE: A Model for Geographical Named Entity Recognition and Correction Based on Deep Neural Networks and the BERT Model. *ISPRS Int. J. Geo. Inf.* **2023**, *12*, 394. [CrossRef]
19. Ju, M.; Miwa, M.; Ananiadou, S. A neural layered model for nested named entity recognition. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Orleans, LA, USA, 1–6 June 2018; Long Papers. Volume 1.
20. Shibuya, T.; Hovy, E. Nested named entity recognition via second-best sequence learning and decoding. *Trans. Assoc. Comput. Linguist.* **2020**, *8*, 605–620. [CrossRef]
21. Tang, B.; Hu, J.; Wang, X.; Chen, Q. Recognizing continuous and discontinuous adverse drug reaction mentions from social media using LSTM-CRF. *Wirel. Commun. Mob. Comput.* **2018**, *2018*, 2379208. [CrossRef]
22. Su, J.; Murtadha, A.; Pan, S.; Hou, J.; Sun, J.; Huang, W.; Wen, B.; Liu, Y. Global pointer: Novel efficient span-based approach for named entity recognition. *arXiv* **2022**, arXiv:2208.03054.
23. Zaratiana, U.; Tomeh, N.; Holat, P.; Charnois, T. GNNer: Reducing overlapping in span-based NER using graph neural networks. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop, Dublin, Ireland, 22–27 May 2022.
24. Zaratiana, U.; Tomeh, N.; Holat, P.; Charnois, T. Named Entity Recognition as Structured Span Prediction. In Proceedings of the Workshop on Unimodal and Multimodal Induction of Linguistic Structures (UM-IoS), Abu Dhabi, United Arab Emirates, 7–8 December 2022.
25. Wan, J.; Ru, D.; Zhang, W.; Yu, Y. Nested named entity recognition with span-level graphs. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics, Dublin, Ireland, 22–27 May 2022; Long Papers. Volume 1.
26. Fisher, J.; Vlachos, A. Merge and label: A novel neural network architecture for nested NER. *arXiv* **2019**, arXiv:1907.00464.
27. Sohrab, M.G.; Miwa, M. Deep exhaustive model for nested named entity recognition. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 31 October–4 November 2018.

28. Li, F.; Wang, Z.; Hui, S.C.; Liao, L.; Zhu, X.; Huang, H. A segment enhanced span-based model for nested named entity recognition. *Neurocomputing* **2021**, *465*, 26–37. [CrossRef]
29. Su, J.; Yu, H. Unified Named Entity Recognition as Multi-Label Sequence Generation. In Proceedings of the 2023 International Joint Conference on Neural Networks (IJCNN), Gold Coast, Australia, 18–23 June 2023.
30. Straková, J.; Straka, M.; Hajič, J. Neural architectures for nested NER through linearization. *arXiv* **2019**, arXiv:1908.06926.
31. Tan, Z.; Shen, Y.; Zhang, S.; Lu, W.; Zhuang, Y. A sequence-to-set network for nested named entity recognition. *arXiv* **2021**, arXiv:2105.08901.
32. Lewis, M.; Liu, Y.; Goyal, N.; Ghazvininejad, M.; Mohamed, A.; Levy, O.; Stoyanov, V.; Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv* **2019**, arXiv:1910.13461.
33. Hu, N.; Zhou, X.; Xu, B.; Liu, H.; Xie, X.; Zheng, H.-T. VPN: Variation on Prompt Tuning for Named-Entity Recognition. *Appl. Sci.* **2023**, *13*, 8359. [CrossRef]
34. Lee, D.-H.; Kadakia, A.; Tan, K.; Agarwal, M.; Feng, X.; Shibuya, T.; Mitani, R.; Sekiya, T.; Pujara, J.; Ren, X. Good examples make a faster learner: Simple demonstration-based learning for low-resource NER. *arXiv* **2021**, arXiv:2110.08454.
35. Gao, T.; Fisch, A.; Chen, D. Making pre-trained language models better few-shot learners. *arXiv* **2020**, arXiv:2012.15723.
36. Chen, Z.; Qian, T. Description and demonstration guided data augmentation for sequence tagging. *World Wide Web* **2022**, *25*, 175–194. [CrossRef]
37. Wang, Y.; Yu, B.; Zhu, H.; Liu, T.; Yu, N.; Sun, L. Discontinuous named entity recognition as maximal clique discovery. *arXiv* **2021**, arXiv:2106.00218.
38. Lynch, C.J.; Jensen, E.J.; Zamponi, V.; O'Brien, K.; Frydenlund, E.; Gore, R. A Structured Narrative Prompt for Prompting Narratives from Large Language Models: Sentiment Assessment of ChatGPT-Generated Narratives and Real Tweets. *Future Internet* **2023**, *15*, 375. [CrossRef]
39. Zeng, D.; Liu, K.; Lai, S.; Zhou, G.; Zhao, J. Relation classification via convolutional deep neural network. In Proceedings of the COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers, Dublin, Ireland, 23–29 August 2014.
40. Wang, L.; Cao, Z.; De Melo, G.; Liu, Z. Relation classification via multi-level attention cnns. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, Berlin, Germany, 7–12 August 2016; Long Paper. Volume 1.
41. Reimers, N.; Gurevych, I. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv* **2019**, arXiv:1908.10084.
42. Wang, B.; Lu, W. Combining spans into entities: A neural two-stage approach for recognizing discontiguous entities. *arXiv* **2019**, arXiv:1909.00930.
43. Fei, H.; Ji, D.; Li, B.; Liu, Y.; Ren, Y.; Li, F. Rethinking boundaries: End-to-end recognition of discontinuous mentions with pointer networks. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 2–9 February 2021.

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