



Article Extracting Domain-Specific Chinese Named Entities for Aviation Safety Reports: A Case Study

Xin Wang¹, Zurui Gan¹, Yaxi Xu^{2,*}, Bingnan Liu³ and Tao Zheng¹

- ¹ School of Computer Science, Civil Aviation Flight University of China, Guanghan 618307, China
- ² School of Economics and Management, Civil Aviation Flight University of China, Guanghan 618307, China
- ³ Department of Information Management, Air China, Chongqing 401120, China

Correspondence: xuyaxi@cafuc.edu.cn; Tel.: +86-13350090790

Abstract: Aviation safety reports can provide detailed records of past aviation safety accidents, analyze their problems and hidden dangers, and help airlines and other aviation enterprises avoid similar accidents from happening again. In a novel way, we plan to use named entity recognition technology to quickly mine important information in reports, helping safety personnel improve efficiency. The development of intelligent civil aviation creates demands for the incorporation of big data and artificial intelligence. Because of the aviation-specific terms and the complexity of identifying named entity boundaries, the mining of aviation safety report texts is a challenging domain. This paper proposes a novel method for aviation safety report entity extraction. First, ten kinds of entities and sequences, such as event, company, city, operation, date, aircraft type, personnel, flight number, aircraft registration and aircraft part, were annotated using the BIO format. Second, we present a semantic representation enhancement approach through the fusion of enhanced representation through knowledge integration embedding (ERNIE), pinyin embedding and glyph embedding. Then, in order to improve the accuracy of specific entity extraction, we constructed and utilized the aviation domain dictionary which includes high-frequency technical aviation terms. After that, we adopted bilinear attention networks (BANs), the feature fusion approach originally used in multi-modal analysis, in our study to incorporate features extracted from both iterated dilated convolutional neural network (IDCNN) and bi-directional long short-term memory (BiLSTM) architectures. A case study of specific entity extraction for an aviation safety events dataset was conducted. The experimental results demonstrate that our proposed algorithm, with an F1 score reaching 97.93%, is superior to several baseline and advanced algorithms. Therefore, the proposed approach offers a robust methodological foundation for the relationship extraction and knowledge graph construction of aviation safety reports.

Keywords: civil aviation; aviation specific terms; ERNIE; pinyin embedding and glyph embedding; entity extraction; aviation domain dictionary; BAN; aviation safety

1. Introduction

Data-driven frameworks for analyzing aviation safety events have recently received attention. The demand for safety information analysis by civil aviation safety personnel is constantly increasing. The Civil Aviation Administration of China (CAAC) attaches great importance to the management of civil aviation safety information and has established data systems such as Aviation Safety Information System, Service Difficulty Reporting System (SDR) and Bird Strike Database. However, the processing and mining of existing safety information has not seen sufficient traction. Efficiently extracting and processing safety information has emerged as a pressing imperative, necessitating a proactive response. Leveraging new-generation information technology empowers civil aviation safety personnel with an intuitive, real-time, and convenient means to process safety information, which leads to the enhanced management of civil aviation safety endeavors [1]. Named entity



Citation: Wang, X.; Gan, Z.; Xu, Y.; Liu, B.; Zheng, T. Extracting Domain-Specific Chinese Named Entities for Aviation Safety Reports: A Case Study. *Appl. Sci.* **2023**, *13*, 11003. https://doi.org/10.3390/ app131911003

Academic Editors: Peter Blišťan, Ladislav Hvizdák and Lucia Domaracká

Received: 4 September 2023 Revised: 27 September 2023 Accepted: 3 October 2023 Published: 6 October 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https:// creativecommons.org/licenses/by/ 4.0/). recognition technology can effectively extract key entities from a text, thereby reducing interference from redundant information. So far, there are few studies that combine aviation safety with named entity recognition technology. Therefore, this paper attempts to conduct named entity recognition research on aviation safety to further promote academic progress in related fields. At the same time, by demonstrating the use of this technology to quickly mine important information in reports, it can help safety personnel improve efficiency.

Safety reports are information collected about actual or potential safety defects. The purpose of safety reports is to improve the security of aircraft operations via the timely detection of operational hazards and system defects. Appropriate and remedial measures can be determined through the timely analysis of safety data, playing a significant role in the prevention of accidents. As shown in Figure 1, an aircraft may potentially encounter thunderstorms, icing, turbulence, and volcanic ash during its operation, which poses a substantial threat to the integrity and safety of the aircraft. Aviation safety reports should be used to record the date, aircraft type, operation, event, etc., of accidents. Utilizing specialized methodologies to analyze aviation safety reports can equip personnel across diverse departments with invaluable insights, thereby enhancing their expertise and bolstering the overall assurance of flight safety. The common bird strike issues are shown in Figure 2.



Figure 1. Aviation flight hazards.



Figure 2. Bird strike issues.

Specific entity extraction refers to the process of identifying the boundaries of specific entities, extracting named entities with a specific meaning, and classifying them into predefined entity categories [2]. In general, pre-defined entity categories are usually the key information in a text. The task of specific entity extraction is to extract key information from a large amount of specific text information. Specific entity extraction plays a crucial role in domain fields such as medicine, aviation and agronomy. In addition, it can lay a valuable foundation for entity relationship extraction and knowledge graph construction. Specific entity extraction is usually considered a sequence labeling task and uses token-based classification techniques such as named entity recognition (NER).

With the expansion of research fields, NER tasks are increasingly receiving attention, but there is still limited recognition of technology terms. Most NER studies only include general named entities such as date, location, and people, which are insufficient to reflect significant concepts in a specific research field. Terms are a type of more granular, meaningful and interpretable entity that better reflect the semantics of a text. In addition to general entities, aviation safety reports contain more terms with civil aviation significance. The automatic extraction of civil aviation safety report terms refers to the named entity recognition of terms with specific civil aviation meanings such as aircraft type, operation, and aircraft part.

The research object of this paper is the case study of aviation safety reports obtained from an aviation company, which are text files recorded for aviation on a weekly basis. An aviation safety report corresponding to a flight includes a textual description of the event. Each file records safety events that occurred within a week. The text length of each event varies considerably. The analysis of the events within the aviation safety reports plays a crucial role in furthering aviation safety, as it can be used to discover defects and research human performance among other things. By extracting meaningful and interpretable entities from safety events, it helps relevant personnel to quickly understand the process of safety events and to summarize experiences. The dataset of aviation safety reports used in the paper is from 2014 to 2016 and comes from an airline. The shortest sentence length in the dataset is 73, the longest sentence length is 1860, and the average sentence length is 166. Ten kinds of entities and sequences, such as event, company, city, operation, date, aircraft type, personnel, flight number, aircraft registration and aircraft part, were annotated using the BIO format. By analyzing the important entities in the reports, detailed background information is provided for the event investigation. This helps investigators determine the cause of the event and provides a basis for developing corresponding preventive measures.

At present, although NER has been well applied in many domains, such as medicine and agriculture, its deployment in digital aviation is very limited. Hou et al. [3] pointed out that airport abnormal event reports are usually stored in an unstructured manner, which is inconvenient for relevant personnel to conduct further analysis. They proposed the use of the Chinese named entity recognition model to structurally process texts and extract key information. Wang et al. [4] effectively and quickly obtained key information, such as the uncivilized penalties, deadlines and behaviors of on-board civil aviation passengers for various uncivilized behaviors information using a model. Xing et al. [5] aimed to obtain the large number of nested and compound named entities in civil aviation enterprise entities, and spliced overlapping dilated convolutional neural network (ODCNN) and bidirectional long short-term memory (BiLSTM) models in different ways to identify different categories of operations, commercial services and general services. However, their results are still limited in the number of aviation entity categories and their efficiency. Meaningful entities are not proper-name entities, which leads to the difficulty of identifying them from unstructured text. Therefore, an efficient approach to extracting entities in a specific domain is required.

This paper conducts research on Chinese specific entity extraction methods for aviation safety reports. Based on existing aviation safety reports, the dataset–aviation safety event is constructed. In view of the insufficient capture of entity boundary information via the general named entity recognition model and the strong professionalism of the research object in the paper, a specific named entity recognition method for aviation safety events that integrates enhanced representation through knowledge integration embedding (ERNIE), domain dictionary, general dictionary, glyph information, and pinyin information is used. We use the ERNIE model for character embedding to obtain character vectors. By training pinyin vectors and glyph vectors separately, we enhance the performance of character embedding vector representation via the integration of these three vectors. In order to incorporate domain word information, we introduce a domain dictionary and trained it to obtain word vectors. At the same time, we also introduce a general dictionary to enable non-domain words to be correctly segmented. At the same time, iterated dilated convolutional neural network (IDCNN) and BiLSTM dual-channel models are constructed at the encoding layer. To fully exploit the features of the dual channels, bilinear attention

networks (BANs) [6] are used to dynamically fuse features extracted from two channels. We conduct experiments to compare different embedding structures, feature extraction structures, feature fusion structures, and the performance of each entity. We use precision, recall, F1 score, time per iteration, optimal training duration, and memory as indicators to demonstrate that the proposed model achieves good results.

2. Related Work

2.1. Approaches Based on Rule and Dictionary

Early approaches to NER tasks were based on rules and dictionaries for entity matching [7,8]. Designing rules based on domain-specific dictionaries and grammars, provided that all entities can be exhaustively defined in a dictionary, is a highly effective method. The matching speed is faster. However, in practical applications, it is impossible to describe all entities. Grammar-rule-based methods require the design of a complex set of rules that need to be constantly updated and maintained, which requires high human and material costs. When using rule-based and dictionary-based NER methods, they usually show high precision and low recall. The transferability of these methods is relatively low [9].

2.2. Approaches Based on Statistical Machine Learning

NER is essentially a multi-classification or sequence labeling task. The category of each entity can be identified to achieve character-level classification tasks via the machine learning algorithm [10,11]. Commonly used machine learning algorithms include the hidden Markov model (HMM) [12], support vector machine (SVM) [13] and conditional random fields (CRFs) [14]. HMM performs well in processing sequence data without the need for manually writing rules. However, for data from different fields and languages, it is necessary to adjust model parameters and state transition matrices, which requires a high level of professional knowledge. SVM can classify named entity recognition tasks based on multiple features such as parts of speech, word frequency, and contextual information. However, the complexity of training time is relatively high. CRFs provide a flexible and globally optimal annotation framework for named entity recognition, with fast training and testing speeds. However, different feature methods have significant impacts on the results. Some automation is possible. However, statistical machine learning algorithms also have some disadvantages. Statistical machine learning requires a sufficient amount of data. For named entity recognition, the number of entities is usually unbalanced, which affects the performance of the model. In addition, it is necessary to choose an appropriate feature extraction method. If the quality of the features is poor, it will affect the performance of the model.

2.3. Approaches Based on Deep Learning

In recent years, several approaches have been proposed which attempt to utilize deep learning, achieving fruitful results in various fields. The deep learning framework is versatile and does not need to consume resources in feature engineering like machine learning does. The performance of neural networks and the effectiveness of training methods have been widely recognized [15]. Relevant researchers have conducted a lot of research on named entity recognition using neural networks. The named entity recognition framework based on deep learning is mainly divided into three parts. The first is the embedding layer, which converts text into a vector form that can be recognized by the computer. Commonly used methods include Word2Vec [16], Glove [17], etc. The second part is the encoding layer, which obtains contextual dependencies based on the network

structure and extracts hidden meanings in a text. Most of the current methods are based on RNN, CNN and transformer [18] structures. The third part is the label sequence decoding layer, which predicts and classifies characters in a text. The most widely used method is CRF. For example, Wan, Q. [19] proposed the extraction of medical named entities from Chinese electronic medical records (CEMR) based on the ELMo-ET-CRF model. In addition, Su [20] proposed GlobalPointer based on the global pointer idea, which realizes the indiscriminate processing of nested and non-nested NER. It can achieve effective results in both non-nested and nested experiments. The most commonly used named entity encoding and decoding structure is the BiLSTM-CRF structure [21], and most of the current research on named entity recognition is also based on this model structure. For example, the paper in [22] proposed a clinical named entity recognition model combining multi-head self-attention, BiLSTM neural networks and conditional random fields.

The pre-trained model has a relatively high status in current natural language processing (NLP). The development of pre-training models has benefited from the advancement of various technologies, including the improvement of computing hardware, the availability of big data, the development of representation learning, the improvement of model architecture, and the advancement of multi-modal data fusion algorithms. The pre-trained model is a model that is stored after training in a large corpus via the consumption of large resources. The model can be fine-tuned and used in various basic tasks of NLP [23]. In recent years, several pre-training models, such as bidirectional encoder representation from transformers (BERT) [24] and generative pre-trained transformer (GPT) [25], have been proposed which have been widely used in NLP. In the named entity recognition task, the Bert pre-training model or the Bert-based pre-training model embedding to generate word vectors has been proven to achieve better results in many studies [26,27]. The traditional word vectors Word2Vec and GloVe are both based on statistical methods. Although these methods can capture the semantic information of words to some extent, they often have certain limitations, such as an inability to capture complex relationships between words. In contrast, both BERT and GPT models adopt the transformer architecture and are trained via a self-attention mechanism. The transformer architecture enables the model to capture complex relationships between words and uses a large number of unsupervised learning methods for training, which improves the quality of vectors.

2.4. Previous Methods for Aviation Safety Reports

Safety is a primary concern for civil aviation. Rose, L. et al. [28] applied structural topic model (STM) separately to a dataset of 13,336 standardized ASRS event narratives and a dataset of 386 non-standardized NTSB accident and incident reports. The STM outputs obtained were a list of topics that described each dataset. Jiao, Y. et al. [29] used the XGBoost classifier and OC-POS vectorization methods to classify reports. Robinson, D. [30] used themes identified via SMEs, and developed new conceptual frameworks. When provided with topics modeled via LDA in a structured manner, SMEs were able to independently identify themes, which demonstrates some of the utility of NLP for aviation safety researchers.

2.5. Previous Methods for Named Entity Recognition

Recently, there have been many studies on named entity recognition in domain datasets. Yuan, T. et al. [31] introduce the ERNIE Adv BiLSTM Att CRF model to address the issue of food safety named entity recognition. In the embedding layer, the ERNIE pre-training model is used to obtain character embedding vectors, while adversarial training is introduced to reduce the impact of dataset noise. The feature extraction layer uses the BiLSTM-CRF structure. He, L. [32] proposed the BERT-MF-BiGRU-CRF model to solve the problem of credit named entity recognition for shipping enterprises. The pre-trained character features were spliced with word vector features, pos features, and word length features. BiGRU-CRF is used in feature extraction models. These models are currently very popular. However, the proposed model did not fully utilize the characteristics of the food

safety field. BiLSTM or BiGRU can only obtain sequence information, and their ability to extract local features is weak.

3. Methods

To address the problem of specific entity extraction with strong professionalism, we proposed a novel and highly scalable model. The structure of the specific entity extraction model for aviation safety events based on fusion with a domain dictionary, general dictionary, glyph information, pinyin information and ERNIE character embedding information in the embedding layer is shown in Figure 3. The model consists of three parts: the embedding layer, the feature extraction layer, and the CRF tag sequence decoding layer. First, the ERNIE pre-training model [33] is used in the embedding layer to map the input text to a character-level embedding vector. Glyph embedding is obtained based on different scripts of a Chinese character and is able to capture the character semantics from the visual features. Pinyin embedding characterizes the pronunciation of Chinese characters, as the same character has different pronunciations with different meanings. Character embedding, glyph embedding, and pinyin embedding are combined to obtain a new character embedding vector. In order to strengthen the ability of the model to identify entity boundaries, the text is segmented via the domain dictionaries and generic dictionaries. Word2vec is used to convert Chinese words into Chinese word vectors, with the same dimensions as character vectors. The character vectors and the word vectors are fused character by character to obtain the output vector of the embedding layer. The fusion vectors are input to BiGRU anthe d IDCNN for feature extraction. If the features extracted via BiGRU and the IDCNN are directly concatenated, there may be redundant or repeated information, resulting in a decrease in model performance. The model uses a BAN to dynamically fuse the two extracted features.



Figure 3. Architecture of model.

3.1. Embedding Layer

The characteristic of the ERNIE pre-training model lies in its transformer-based multilayer self-attention bidirectional modeling ability, as well as the use of a large-scale corpus for training, which can be used in various NLP tasks. We use the ERNIE pre-training model to generate word embedding vectors for the dataset in this paper. Pinyin, as one of the important symbols of Chinese characters, can provide pronunciation information for each character. As another fundamental feature of Chinese characters, glyphs provide structural information and visual features for each character. Incorporating the two basic features of pinyin and glyphs into the training process of the character embedding model can provide a richer feature representation for the model. A civil aviation dictionary contains professional terms in the field of civil aviation, which are often difficult to cover in general dictionaries. Therefore, by using domain dictionaries, we can incorporate these specialized terms into the training process of word embedding vectors, thereby enabling the model to better understand and process the semantic information of these words.

The specific entity extraction task is a character-level prediction and classification task. When character vectors are used as input, the relationship between characters can be fully exploited. By introducing the features of Chinese character glyphs and pinyin, the embedding layer can capture the semantic information of Chinese character glyphs and pinyin. The glyph feature can help the model to better distinguish different Chinese character glyphs. The pinyin feature can help the model to distinguish different Chinese characters with similar pronunciations. This can improve the model's performance and enable the model to better understand the meaning of a text. However, for Chinese, the information contained in a single character is relatively limited. Chinese entities are usually made up of several characters. It is difficult to identify entity boundaries using character vectors as input entities. The difficulty of identifying entity boundaries can be solved by introducing domain dictionaries. There are many domain words in the aviation domain. However, domain dictionaries are usually small in size. It is difficult to train rich word embedding vectors using only a small number of domain words. The embedding information of words is relatively limited. The paper proposes a model that combines general word embedding vectors and domain word embedding vectors. A fusion method is proposed that combines fusion character features and double Chinese word features. Character vectors and word vectors have different characteristics and functions in entity extraction. Character vectors mainly focus on the detailed features of individual characters or text. Word vectors pay more attention to the contextual relationships and semantic information between words. Character vectors and word vectors provide information of different granularities, and can complement and coordinate with each other. By integrating information of different granularities, a model's ability to capture semantic information in a text can be improved, and the boundary position of entities can be more accurately determined. This can solve the problem of difficult-to-determine physical boundaries. By introducing professional terms from domain dictionaries, the vocabulary scope of the model can be expanded, enabling it to better understand and process semantic information on specific domains, thereby more accurately identifying entity boundaries and extracting entity terms.

First, the text is divided into characters. The characters are mapped to character vectors using the pre-training model ERNIE-3.0. Pinyin embedding and glyph embedding in this article refer to the ChineseBERT [34] scheme. Each character is instantiated as a 24 × 24 image with floating point pixels from 0 to 255. We convert the 24 × 24 × 3 vector into a 2352 vector. We use the open-sourced pinyin package to generate pinyin sequences for its constituent characters. Tone is represented by adding a special marker at the end of a character sequence. Pinyin embedding is generated through a CNN model with a width of 2. The domain dictionary is constructed during the labeling process, and is used to train word vectors. Character vectors and word vectors are merged character by character to obtain character segmentation is given as X = {x₁, x₂, ..., x_n} and character vectors are generated via fusion to obtain C = {c₁, c₂, ..., c_n}. The sequence of the domain dictionary is X' = {x₁', x₂', ..., x_m'}, and the word vector W = {w₁, w₂, ..., w_m} is obtained after training. The character vector C, the word vector W and the word vector W' are concatenated character by character to obtain CDW = {cdw₁, cdw₂, ..., cdw_n}.

3.2. Encoding Layer

In order for the model to fully learn the characteristics and key information of the context, two modules are designed in the layer: BiGRU and THE IDCNN#. The output matrices, $CDW = \{cdw_1, cdw_2, ..., cdw_n\}$, of the embedding layer are sent to two modules, and the outputs of each module are dynamically fused.

3.2.1. BiGRU

The RNN is a type of neural network that processes sequential data. However, there are problems of gradient explosion and gradient disappearance when processing long sequences. By introducing the gate mechanism and memory cells to store long-range dependencies, LSTM can solve the problem of gradient explosion and gradient disappearance. There are three gate mechanisms: the forget gate, input gate and output gate. The input gate determines what information should be input, the forget gate controls what information should be forgotten, and the output gate controls what information should be output. This improves the model's ability to handle long-range dependencies. BiLSTM is a form of bidirectional LSTM. LSTM can only capture directional context information, and it cannot use the following contextual information. LSTM can only rely on the previous information, which makes the extracted features have certain limitations. By capturing front-to-back and back-to-front contextual dependencies, BiLSTM can fully obtain more comprehensive features. GRU and LSTM can achieve similar effects, but in comparison, GRU training consumes less. BiGRU contains two gating units: the update gate and the reset gate. Compared to BiLSTM's three gating units, BiGRU has fewer parameters and a faster computation speed. In addition, like BiLSTM, BiGRU can also transfer information in both directions and obtain comprehensive features.

The structure of the GRU is shown in Figure 4. x_t denotes the input at moment t, and h_t denotes the input of the hidden state at moment t. The GRU network forward propagation weight parameter update formula is

$$zt = \sigma(wxz \cdot xt + whz \cdot ht - 1 + bz) \tag{1}$$

$$rt = \sigma(wxr \cdot xt + whr \cdot ht - 1 + br)$$
⁽²⁾

$$ht = \tanh(wxh \cdot xt + rt \otimes whh \cdot ht - 1) \tag{3}$$

$$ht = zt \otimes ht - 1 + (1 - zt) \otimes ht \tag{4}$$

where *ht* donates the candidate hidden state, h_t donates the hidden state which is the output of the GRU unit, r_t is a reset gate for controlling whether or not the current content is remembered, z_t is an update gate for controlling the proportion of preorder memory and candidate memory in the current state, + denotes vector splicing, \otimes denotes a multiplication by the corresponding elements, σ denotes a sigmoid function, w is the weighting factor, and b is the offset term.



Figure 4. Structure of GRU.

The fusion vector $CDW = \{cdw_1, cdw_2, ..., cdw_n\}$ is the input. We obtain $Ht = \{ht_1, ht_2, ..., ht_n\}$ through the forward direction. We obtain $Ht = \{H't_1, H't_2, ..., H't_n\}$ through the backward direction. The functions of splicing two directions result in $H = \{Ht; H't\}$.

Aviation entities in aviation safety events are long and complex in composition, and there is the phenomenon of nested entities. BiGRU effectively captures contextual information in text through its bidirectional characteristics and GRU's update mechanism. BiGRU is introduced to obtain the contextual features of a text to solve the recognition error problem caused by long entity names. Two one-way GRUs are combined together to form BiGRU to obtain the forward and reverse information to fully extract the contextual features of aviation safety events.

3.2.2. IDCNN

The CNN model first appeared in the field of computer vision (CV) and is mainly used to solve various problems in CV. Related scholars created some deformations for the input layer of CNN and proposed a text classification model, TextCNN. TextCNN in NLP is similar to the CNN structure in CV, which consists of an input layer, a convolutional layer and a pooling layer. TextCNN has achieved fruitful results in some NLP tasks. However, there are still some problems. The convolution kernel of TextCNN is fixed, so the model can only recognize local features of a fixed size and cannot capture long sequence features. The extraction of multi-level features is usually better than that of single-level features. TextCNN can only extract single-level features, which has certain limitations.

To solve the above problems, the DCNN is a preferable improvement. The DCNN includes the parameter of dilated rate that can expand the receptive field. DCNN can cover a larger area, so the extracted features are not limited to local areas. The IDCNN extracts multi-level features by stacking DCNNs, and fully integrates shallow features and deep features.

During the convolution operation, the convolution width will increase exponentially with the increase in the number of layers because of the expansion width. Finally, all the input text will be covered quickly. For each Chinese character of the input text, the IDCNN outputs a vector, which is the probability of each tag corresponding to each character calculated via the IDCNN. The IDCNN model captures contextual information by using dilated convolution and stacking. Features are integrated together in a stacked manner to better understand and handle the internal structure and relationships of nested entities.

The structure of the IDCNN is shown in Figure 5. The first graph is a 3×3 convolution kernel, just like a common convolution operation. The second graph undergoes an expansion with a step size of 2, resulting in a perception range of 5×5 . The third graph is expanded with a step size of 4, expanding the perception range from 5×5 to 7×7 . In this paper, the IDCNN network used is composed of three identically sized dilation convolutional blocks with dilation widths set to 1, 1, and 2.



Figure 5. Structure of IDCNN.

3.2.3. Multi-Head Self-Attention

The multi-head attention mechanism improves the expressiveness of the model, and the multi-head attention mechanism is added after the BiGRU and IDCNN to focus attention on meaningful and interpretable entities. The features are linearly transformed to obtain a query, key and value matrix. The query, key and value matrix are used to compute the attention score. The input vector is mapped to different linear spaces to obtain multiple heads. The attention score of each head is calculated and they are concatenated together.

The formula for the self-attention score is as follows:

$$Attention(Q, K, V) = \operatorname{softmax}(\frac{QK^{T}}{\sqrt{dk}})V$$
(5)

where Q represents the query matrix, which queries the weight of the current position. K represents the key matrix, which calculates the similarity between the current position and other positions. The softmax function represents the normalization of weights and the sum of attention weights at all positions to 1, enabling the model to focus correctly in different positions, thereby improving its performance. V represents the value matrix, used to calculate the weighted sum of weights and values. d_k represents the dimension of the key matrix. The purpose of introducing d_k is to obtain a stable gradient.

Features are mapped to different spaces using different weight matrices, and attention scores are calculated separately in each space.

$$headi = Attention(QW_i^Q, KW_i^K, VW_i^V)$$
(6)

where W_i^Q , W_i^K , and W_i^V represent the weight matrix used for linear transformations; the value of *i* corresponds to different linear spaces.

The scores of multiple heads are concatenated.

$$MultiHead(Q, K, V) = Concat(head1, head2, ..., headh)$$
(7)

The Concat function concatenates attention weights from different subspaces.

Multi-head attention divides the input sequence into multiple heads, each of which can independently view the input, thus capturing contextual information. Different headers can focus on different parts of the input sequence, allowing for the processing of nested entities.

3.2.4. Bilinear Attention Networks

Bilinear attention networks were originally used to solve the problem of multimodal feature fusion. Bilinear attention is an attention mechanism that uses a bilinear function to compute the similarity between two sets of features. In the case of multimodal feature fusion, the two sets of features are image features and text features. The bilinear attention mechanism computes attention weights for each pair of features and uses these weights to generate a fused representation of the features. BANs capture contextual information by considering bilinear interactions between two sets of input channels.

This study employs bilinear attention networks to dynamically combine the features extracted from BILSTM and the IDCNN. This strategic fusion enables the comprehensive extraction of textual sequence information, thereby enhancing the overall performance of the model. The specific procedure involves obtaining two feature matrices from the two extracted features through linear transformations. Subsequently, the similarity matrix between the two feature matrices is computed using bilinear operation, yielding the attention matrix.

Features *A* and *B* are extracted from two channels. Features *A* and *B* are subjected to linear transformations to obtain the transformed representations, A' and B', respectively.

$$A' = A \cdot W_A \tag{8}$$

$$BI = B \cdot W_B \tag{9}$$

where W_A and W_B represent the weight matrix of linear transformation. *A* and *B* represent the features extracted from BiLSTM and the IDCNN, respectively. *A'* and *B'* represent the features obtained after linear transformation.

The bilinear pooling operation is performed between A' and B' to compute attention weights that capture the relationship between A' and B'. The attention weights are normalized using the softmax function.

$$A_weights = \text{softmax}(A' \cdot W_H \cdot B'^T)$$
(10)

where W_H is a learnable parameter matrix that represents the interaction relationship between two feature vectors. The softmax function represents a method of normalizing attention weights.

Element-wise multiplication between the attention weights and A' is performed to compute the weighted sum of the A feature.

$$A_weighted = A_weights \cdot A \tag{11}$$

The weighted feature A and feature B' are fused and concatenated to yield the output of the fused features.

$$concat_features = concat(A_weighted, B)$$
 (12)

3.3. CRF Layer

The CRF model gives a higher probability value to the reasonable label sequence by adding constraints between labels to obtain the best predicted sequence. CRF cannot consider the long-range dependencies of sentences such as those of the RNN structure, but it can better consider the linear weighted sum of local features of sentences to make the sequence more reasonable. In the CRF layer, the transition matrix is a parameter of the CRF model, which can randomly initialize the score of the transition matrix and then update it during training. In this way, CRF learns the dependency relationship between labels. For example, the probability of "B-Event" transitioning to "I-Event" is 0.9, but the probability of it transitioning to I-Date is "0.0006".

The following section outlines several rules pertaining to specific entity extraction:

- (1) The label of the first Chinese character in a sentence starts with "B -" or "O -"
- (2) In 'B-label1 I-label2 I-label3...' label 1, label 2 and label 3 should be the same entity category. For example, 'B- Event I- Event' is correct while 'B- Event I-Date' is incorrect.
- (3) The first Chinese character label of the specific entity should be 'B -' instead of 'I -'.

For a piece of text, the input text sequence is denoted by $x = \{x_1, x_2, x_3, ..., x_n\}$. For a sequence of predicted tags of the sentence $x, y = (y_1, y_2, ..., y_n)$, the tag sequence score of the whole sentence is equal to the sum of the scores of each position. The position score consists of two parts; one part is determined by the CRF before the model output is determined, and the other part is determined by the transition matrix of the CRF. The score of the label sequence y_i of the sentence x is

$$score(x,y) = \sum_{i=1}^{n} P_{i,yi} + \sum_{i=1}^{n+1} A_{yi-1,yi}$$
 (13)

where $P_{i,yi}$ denotes the score of the y_{ith} tag of the ith character, A is the transfer score matrix, $A_{yi-1,yi}$ denotes the score transferred from tag y_{i-1} to tag y_i , and y_1 and y_{n+1} are the tags added at the beginning and end of the sentence.

The scores of all labels are normalized.

$$P(y|x) = \frac{\exp(score(x,y))}{\sum \exp(score(x,y'))}$$
(14)

where y' represents the possible label values.

Finally, likelihood estimation is used to obtain the sequence of labels. The output sequence with the highest score is predicted via Equation (15) upon final decoding.

 $y* = \operatorname{argmax}(x, y') \tag{15}$

4. Experiments

4.1. Experimental Environment and Dataset

In the experiment, Python 3.7 and paddlepaddle 2.4.0 training frameworks are used. The workstation used in the experiment is manufactured by Dell China Co., Ltd., at Beijing, China. The configuration of this workstation is as follows: The CPU is Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz CPU. The memory is 32G RAM. The GPU is Nvidia Tesla V100-SXM2-16GB GPU.

Aviation safety reports are summary reports of aviation safety events made on a weekly basis, and they describe in detail the date, flight number, cause and resolution of aviation safety events that have occurred within a week. The research object of this paper is the aviation safety events dataset. Weekly reports from 2014 to 2016 are selected, with 150 documents in total. Examples of aviation safety events are shown in Figure 6.

<pre>**年*月*日, A***/B-****飞机执行 CA****(深圳-杭州) 航班, 落地后机务检 查发现右侧空速管遭鸟击, 更换后测试正 常, 放行飞机。 On * *, *, Airbus A***/B-**** aircraft operating flight CA**** (Shenzhen to Hangzhou) experienced a bird strike on the right-side pitot tube. After landing, during the maintenance inspection, it was discovered that the pitot tube was damaged due to the bird strike. The pitot tube was replaced, and subsequent tests confirmed its proper functionality. The aircraft was cleared for release.</pre>	**年*月*日, B***/B-**** 低机执行 CA****(哈密-北京)航班, 落地后机务检 查发现7号扰流板被外来物击伤, 填胶修理 后放行飞机。 On * *, *, Airbus B***/B-**** aircraft operating flight CA**** (Hami to Beijing) experienced foreign object damage to its number 7 spoiler during landing. Upon conducting a maintenance inspection after landing, it was observed that the spoiler had been struck by a foreign object. The aircraft underwent a repair process involving filling with sealant, following which it was cleared for release.
<pre>**年*月*日, A***/B-****飞机执行CA***(北京-米兰) 航班, **年*月*日上午在米兰 机场货库发现该航班运载的一桶货物出现泄 漏现象。 On * *, *, Airbus A***/B-**** aircraft was operating flight CA*** (Beijing to Milan). On the morning of * *, *, at the cargo warehouse of Milan airport, it was discovered that a barrel of cargo carried on this flight exhibited a leakage.</pre>	**年*月*日, A***/B-****飞机执行CA***(香港-北京) 航班, 落地后卸货人员发现前 下货舱一件货物包装破损。 On * *, *, Airbus A***/B-**** aircraft operated flight CA*** (Hong Kong to Beijing). Upon landing, cargo personnel noticed that the packaging of a piece of cargo in the forward lower cargo compartment was damaged.

Figure 6. Examples of civil aviation safety events. (Replace sensitive information with *).

By analyzing civil aviation safety events, consulting the relevant literature and consulting experts in related fields, the specific entities of civil aviation safety events are classified, defined and divided into 10 categories. The open-source data annotation tool 'doccano' is used to annotate the specific entities, and the BIO format is used for sequence tagging tasks. If the current character is the beginning of a specific entity, it is marked as B-label. I-label indicates that the current character is inside the specific entity. If it does not belong to any entity, it is marked as O-label. Finally, 29,129 specific entities are obtained, and the statistics of the number of entities in each category are shown in Table 1.

Table 1. Numbers of entities.

Entity	Numbers	
Event	1603	
Company	1601	
City	1600	
Operation	3895	
Date	1595	
Aircraft Type	3349	
Personnel	1417	
Flight Number	7097	
Aircraft Registration	5120	
Aircraft Part	1852	
Total	29,129	

After the specific entity labeling process, the identified entities are extracted and subsequently organized into a domain dictionary. In total, 4151 unique and non-repetitive domain-specific proper nouns are obtained, which are used to embed domain entity information into the specific entity extraction model.

Examples of extracted information are shown in Figure 7. Important entities are extracted from text through named entity recognition, reducing redundant information. Useful information from a large amount of unstructured text is extracted and stored in a structured manner, facilitating subsequent data analysis and processing.

年*月*日, A*/B-****飞机执行CA***(北京-米兰) 航班, **年*月*日上午在米兰机场货库发现该航班运载 的一桶货物出现泄漏现象。
On * *, *, Airbus A***/B-**** aircraft was operating flight CA*** (Beijing to Milan). On the morning of * *, *, at the cargo warehouse of Milan airport, it was discovered that a barrel of cargo carried on this flight exhibited a leakage.
Name entity extraction: Date: * *,* Aircraft Type: A***/B-**** Flight Number: CA*** City: Beijing, Milan Event: a barrel of cargo carried on this flight exhibited a leakage

Figure 7. Examples of extracted information. (Replace sensitive information with *).

4.2. Evaluation Indicators

This paper uses the *F*1 score to evaluate the effect of the model. When the positive and negative samples are extremely unbalanced, it is easy to be misled by focusing only on the

accuracy rate. The *F*1 score can not only comprehensively consider accuracy and recall, but can also better comprehensively evaluate the performance of the model.

$$Precision = \frac{TP}{TP + FP} \cdot 100\%$$
(16)

$$Recall = \frac{TP}{TP + FN} \cdot 100\% \tag{17}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \cdot 100\%$$
(18)

where *TP* refers to the number of samples correctly predicted by the model as positive, with a positive true value and a positive prediction result. *FP* refers to the number of negative samples incorrectly predicted by the model as positive, with a negative true value but a positive prediction result. *FN* refers to the number of positive samples incorrectly predicted by the model as positive value but a positive predicted as negative, with a true positive value but a predicted negative result.

4.3. Model Parameters

The parameters of the model in this paper are shown in Table 2.

Table 2. Model parameters.

Parameters	Value
Batch_size	64
Epoch	200
learning_rate	0.0001
BiGRU_hidden_size	300
kernel_size	3
num_filters	128
Blocks	4
dilation_l	[1,1,2]
embedding_num	768
BiGRU_num_layer	2
IDCNN_input_dropout	0.5
IDCNN_hidden_dropout	0.2

4.4. Experimental Results and Discussions

In this section, we compare the performance of our approach with that of existing state-of-the-art algorithms as well as that of baseline methods. In order to verify the effectiveness of the model, the impact of the model structure and the domain dictionary embedding method on the specific entity extraction task of textual datasets is discussed. BiLSTM-CRF, BiGRU-CRF and IDCNN-CRF are designed and compared with the IDCNN-BiLSTM-CRF structure in the paper. On this basis, a comparison of ERNIE embedding, ERNIE embedding with a general dictionary, ERNIE embedding of our model is proposed. 'Embedding' is the method used by the model in the embedding layer. 'F1 score' is the F1 value score calculated by the model during testing. 'Time per iteration' is the time it takes for one epoch of the model.

All of the above models perform relatively well in Tables 3–6. Overall, the effect of the ERNIE embedding model integrated with double word embeddings, pinyin embedding and glyph embedding is better than that of other embeddings, and is improved by 0.03–0.35%. The optimal training time is generally reduced. There is some increase in the single iteration time due to there being more features. BiLSTM-CRF, BiGRU-CRF, and IDCNN-CRF are classic methods in named entity recognition that can achieve good results. Compared to the BiLSTM model, BiGRU requires less training time and consumes less resources while

its performance can reach the same level as that of BiLSTM. Compared to BiGRU, the IDCNN consumes more time during training, indicating that BiGRU is the optimal choice for single-channel models when extracting aviation safety reports. In addition, the F1 score of the IDCNN-BiGRU hybrid model used in this paper is better than that of other single models, indicating that the hybrid model can extract features from different angles and has better research significance. When keeping the feature extraction structure unchanged, the ERNIE word embedding vector in the embedding layer is compared with other character and word fusion vectors. If a general dictionary is used to train word vectors, it is difficult to accurately partition the words in aviation safety texts, resulting in a slight decrease in the performance of the model. Replacing a regular dictionary with a domain dictionary can improve the performance of the model by accurately partitioning the text, resulting in better results than those of the ERNIE word embedding vector. If a general dictionary and domain dictionary are used together, not only can specialized terms related to a specific field be identified, but also, the common dictionary has richer feature representations, further improving the results. Adding pinyin feature vectors and glyph feature vectors to the character embedding vector can also improve the F1 score to some extent.

Table 3. Comparison of experimental results of different embeddings for BiLSTM-CRF.

Embedding	F1 Score	Time Per Iteration	Optimal Training Duration
ERNIE	0.9751	21.4 s	501 s
ERNIE + general dictionary	0.9744	20.6 s	301 s
ERNIE + domain dictionary	0.9752	21.8 s	383 s
ERNIE + double dictionary	0.9757	21.0 s	473 s
ERNIE + double dictionary + Pinyin + Glyph	0.9779	21.7 s	489 s

Table 4. Comparison of experimental results of different embeddings for BiGRU-CRF.

Embedding	F1 Score	Time Per Iteration	Optimal Training Duration
ERNIE	0.9757	20.4 s	381 s
ERNIE + general dictionary	0.9755	20.0 s	172 s
ERNIE + domain dictionary	0.9767	21.9 s	336 s
ERNIE + double dictionary	0.9777	20.5 s	889 s
ERNIE + double dictionary + Pinyin + Glyph	0.9780	22.1 s	271 s

Table 5. Comparison of experimental results of different embeddings for IDCNN-CRF.

Embedding	F1 Score	Time Per Iteration	Optimal Training Duration
ERNIE	0.9752	19.6 s	1013 s
ERNIE + general dictionary	0.9766	19.6 s	621 s
ERNIE + domain dictionary	0.9769	21.1 s	687 s
ERNIE + double dictionary	0.9774	19.9 s	714 s
ERNIE + double dictionary + Pinyin + Glyph	0.9779	21.1 s	816 s

Table 6. Comparison of experimental results of different embeddings for IDCNN-BiGRU-CRF.

Embedding	F1 Score	Time Per Iteration	Optimal Training Duration
ERNIE	0.9773	21.7 s	617 s
ERNIE + general dictionary	0.9770	22.6 s	489 s
ERNIE + domain dictionary	0.9785	21.9 s	548 s
ERNIE + double dictionary	0.9786	23.3 s	608 s
ERNIE + double dictionary + Pinyin + Glyph	0.9793	21.6 s	730 s

In this section, we compare the performance of our approach with existing advanced pre-training models. To verify the effectiveness of our model, the impact of different pre-trained models on specific entity extraction tasks in textual datasets is discussed. The traditional Word2Vec and the pre-training models ERNIE-1.0, ERNIE-3.0, Bert-base-Chinese, TinyBert and RoBERTa are used for comparison. 'Model' is the pre-trained model used.

'F1 score' is the F1 value score calculated via the model during testing. 'Memory' is the memory consumed via training the model.

The results of different pre-training models are shown in Table 7. Comparing Word2Vec with the current classic pre-training model, the F1 score has increased by approximately 3–4%. BERT is the basic pre-training model, and other pre-training models are improvements based on BERT. TinyBert is a distilled version of the BERT model, with fewer parameters and a faster training speed, but the pre-training model in the specific entity extraction model in this paper is average. Compared with BERT, RoBERTa has more parameters in the model, and proposes a dynamic mask. The F1 score of RoBERTa shows certain improvements in the specific entity extraction model compared with that of BERT. ERNIE integrates external knowledge sources, performs task-specific and domain-specific pre-training, and performs best in this experiment.

Table 7. Comparison results of specific entity extraction models based on different pre-trained language models.

Model	F1 Score	Memory
Word2Vec	0.9373	3.01 G
ERNIE-1.0	0.9752	14.54 G
ERNIE-3.0	0.9793	14.90 G
Bert-base-Chinese	0.9734	15.42 G
TinyBert	0.9676	11.47 G
RoBERTa	0.9745	14.57 G

In this section, we compare the performance of our approach with that of the existing feature fusion algorithm. In order to verify that the method of feature fusion used in the text is better, this paper designs two sets of traditional methods to directly add and splice vectors. The method used in this paper is compared with them.

The results of different feature fusion methods are shown in Table 8. Compared with the traditional feature fusion method, the method in this paper has achieved better results. The Concat method directly fuses features, which lead to more redundant information, and the improvement of classification prediction is relatively limited. However, our method calculates the similarity matrix of features and then fuses them, which leads to less redundant information, does not increase the training time and can improve the accuracy of the model.

 Table 8. Results of different feature fusion methods.

Model	F1 Score	Time Per Iteration
Add	0.9783	22.0
Concat	0.9774	21.9
BAN	0.9793	21.9

In this section, we show the performance of various entity categories. In order to further analyze the impact of specific entity extraction on civil aviation, the precision rate, recall rate and F1 score of each entity are extracted for comparison. The specific entity extraction method is used in this paper to analyze which entities are better and which entities need to be targeted for improvement.

The results of each entity are shown in Table 9. We have examined indicators for each category. The indicators of each entity category in the specific entity extraction task of the model in this paper are shown in the table above. Overall, the indicators of all entity categories have demonstrated outstanding performance and results. They work best with regular categories such as date, aircraft type and flight number. The identified domain entity categories, such as event, operation and aircraft part, show a certain degree of nesting significance, resulting in relatively favorable results. The obtained results still have room for improvement. In the future, we will attempt to more effectively handle the

model structure of nested entities. Furthermore, a larger dataset is needed to expand the number of entities and the civil aviation domain dictionary, enhancing the performance of the model's embedding layer. In addition, there is a certain imbalance in the number of entity categories in this dataset, and some data augmentation methods may be adopted in the future.

Entity	Precision	Recall	F1 Score
Event	0.9481	0.9639	0.9560
Company	1.0000	1.0000	1.0000
City	1.0000	1.0000	1.0000
Operation	0.9748	0.9845	0.9796
Date	1.0000	1.0000	1.0000
Aircraft Type	0.9969	1.0000	0.9984
Personnel	0.9940	0.9970	0.9955
Flight Number	1.0000	1.0000	1.0000
Aircraft Registration	0.9969	0.9969	0.9969
Aircraft Part	0.8736	0.8736	0.8736

Table 9. The results of each entity in the model.

5. Conclusions

In this paper, we proposed a novel model based on fusion with a double dictionary, fusion character features and bilinear attention networks. Outstanding performance and results were obtained from the self-build aviation safety events dataset. By fusing pinyin embedding vectors and glyph embedding vectors into character vectors, character vectors can learn richer character semantic information. By incorporating the aviation domain dictionary and the general dictionary at the embedding level, the model can better extract entity boundaries and identify correct classification results. Compared with the traditional fusion method, the BAN dynamic feature fusion method used in this paper can achieve better results without increasing the training time, which has certain research significance. We compare our approach to some baseline and advanced methods. Our experimental results indicate superior performance across all used evaluation measures. However, there are also some shortcomings that need to be addressed in future research. The F1 score of the aircraft part entity is relatively low due to the large number of aircraft part entities and the many nested entities. In the future, it is necessary to increase the relevant data to build a larger aviation safety events dataset, and to build a larger domain dictionary to further improve the performance of the model. Empirical results indicate that our approach is superior to several baseline and advanced approaches, laying a meaningful foundation for the subsequent relationship extraction and knowledge graph construction of aviation safety. At present, this study is still in the research stage. In fact, we will consider integrating it into safety analysis software for use by safety personnel in future research. In the future, we may continue to study the relationships between entities and further establish a knowledge graph.

Author Contributions: X.W. and Z.G.: conceptualization, methodology, validation, and writing—review and editing; Y.X.: conceptualization, methodology, and writing—review and editing; B.L. and T.Z.: methodology and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially supported by the R&D Program of Key Laboratory of Flight Techniques and Flight Safety, CAAC (No. FZ2022ZZ01), the Fundamental Research Funds for the Central Universities (No. J2022-048).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The dataset used in this paper is provided by an airline and it is confidential.

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

References

- 1. Shi, Y.; Chen, Y. Problems and countermeasures on aviation safety information management. J. Saf. Sci. Technol. 2010, 6, 116–120.
- Wang, Y.; Zhang, C.; Bai, F.; Wang, Z.; Ji, C. Review of Chinese Named Entity Recognition Research. J. Front. Comput. Sci. Technol. 2023, 17, 324–341.
- 3. Hou, Q.; Yuan, T.; Wang, L. Research on Detection and Recognition Method of Airport Abnormal Event Entities. *Comput. Meas. Control* **2022**, *30*, 62–69.
- 4. Cao, W.; Xu, X. Research on methods of identifying unruly passengers in civil aviation. J. Civ. Aviat. Univ. China 2022, 40, 24–30.
- Xing, Z.; Dai, Z.; Luo, Q.; Liu, Y.; Chen, Z.; Wen, T. Research on Name Entity Recognition Method in Civil Aviation Text. In Proceedings of the IEEE 2nd International Conference on Civil Aviation Safety and Information Technology, Weihai, China, 14–16 October 2020; IEEE: Piscataway, NJ, USA, 2020; pp. 23–29.
- 6. Kim, J.; Jun, J.; Zhang, B. Bilinear attention networks. *arXiv* 2018, arXiv:1805.07932.
- Alfred, R.; Leong, L.; On, C.; Anthony, P. Malay named entity recognition based on rule-based approach. *Int. J. Mach. Learn. Comput.* 2014, 4, 300–306. [CrossRef]
- 8. Yuan, J.; Pan, M.; Zhang, T.; Jiang, Y. Electricity safety domain named entity recognition based on rules and dictionaries. *Appl. Electron. Technol.* **2022**, *48*, 22–27.
- 9. Zhao, S.; Luo, R.; Cai, Z. Survey of Chinese Named Entity Recognition. J. Front. Comput. Sci. Technol. 2022, 16, 296–304.
- 10. Soomro, P.; Kumar, S.; Banbhrani; Shaikh, A.; Raj, H. Bio-NER: Biomedical Named Entity Recognition using Rule-Based and Statistical Learners. *Sci. Inf. Organ. Ltd.* **2017**, *8*, 163–170.
- Mozharova, V.; Loukachevitch, N. Combining knowledge and CRF-based approach to named entity recognition in Russian. In Proceedings of the Analysis of Images, Social Networks and Texts: 5th International Conference, AIST 2016, Yekaterinburg, Russia, 7–9 April 2016; Revised Selected Papers 5. Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 185–195.
- Morwal, S.; Jahan, N.; Chopra, D. Named entity recognition using hidden Markov model (HMM). *Int. J. Nat. Lang. Comput.* 2012, 1, 15–23. [CrossRef]
- 13. Ekbal, A.; Bandyopadhyay, S. Named entity recognition using support vector machine: A language independent approach. *Int. J. Electr. Comput. Eng.* **2010**, *4*, 589–604.
- Yao, L.; Sun, C.; Li, S.; Wang, X.; Xuan, W. CRF-based active learning for Chinese named entity recognition. In Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics, San Antonio, TX, USA, 11–14 October 2009; IEEE: Piscataway, NJ, USA, 2009; pp. 1557–1561.
- 15. Zhang, R.; Dai, L.; Wang, B.; Guo, P. Recent Advances of Chinese Named Entity Recognition Based on Deep learning. *J. Chin. Inf. Process.* **2022**, *36*, 20–35.
- 16. Sienčnik, S. Adapting word2vec to named entity recognition. In Proceedings of the 20th Nordic Conference of Computational Linguistics, Vilnius, Lithuania, 11–13 May 2015; pp. 239–243.
- 17. Ning, G.; Bai, Y. Biomedical named entity recognition based on Glove-BLSTM-CRF model. J. Comput. Methods Sci. Eng. 2021, 21, 125–133. [CrossRef]
- Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.; Kaiser, L.; Polosukhin, I. Attention is all you need. *arXiv* 2017, arXiv:1706.03762.
- Wan, Q.; Liu, J.; Wei, L.; Ji, B. A self-attention based neural architecture for Chinese medical named entity recognition. *Math. Biosci. Eng.* 2020, 17, 3498–3511. [CrossRef] [PubMed]
- 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.
- 21. Huang, Z.; Xu, W.; Yu, K. Bidirectional LSTM-CRF models for sequence tagging. arXiv 2015, arXiv:1508.01991.
- 22. Li, C.; Ma, K. Entity recognition of Chinese medical text based on multi-head self-attention combined with BILSTM-CRF. *Math. Biosci. Eng.* **2022**, *19*, 2206–2218. [CrossRef] [PubMed]
- 23. Yue, Z.; Ye, X.; Liu, R. A Survey of language Model Based Pre-training Technology. J. Chin. Inf. Process. 2021, 35, 15–29.
- 24. Devlin, J.; Chang, M.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv* **2018**, arXiv:1810.04805.
- 25. Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving Language Understanding by Generative Pre-Training. 2018. Available online: https://blog.openai.com/language-unsupervised (accessed on 3 September 2023).
- Wang, Y.; Sun, Y.; Ma, Z.; Gao, L.; Xu, Y.; Sun, T. Application of pre-training models in named entity recognition. In Proceedings of the 2020 12th International Conference on Intelligent Human-Machine Systems and Cybernetics, Hangzhou, China, 22–23 August 2020; IEEE: Piscataway, NJ, USA, 2020; Volume 1, pp. 23–26.
- 27. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* **2023**, *55*, 1–35. [CrossRef]
- Rose, L.; Puranik, G.; Mavris, N.; Arjun, R. Application of structural topic modeling to aviation safety data. *Reliab. Eng. Syst. Saf.* 2022, 224, 108522. [CrossRef]
- Jiao, Y.; Dong, J.; Han, J.; Sun, H. Classification and Causes Identification of Chinese Civil Aviation Incident Reports. *Appl. Sci.* 2022, 12, 10765. [CrossRef]

- 30. Robinson, D. Temporal topic modeling applied to aviation safety reports: A subject matter expert review. *Saf. Sci.* **2019**, *116*, 275–286. [CrossRef]
- 31. Yuan, T.; Qin, X.; Wei, C. A Chinese Named Entity Recognition Method Based on ERNIE-BiLSTM- CRF for Food Safety Domain. *Appl. Sci.* 2023, 13, 2849. [CrossRef]
- 32. He, L.; Wang, S.; Cao, X. Multi-Feature Fusion Method for Chinese Shipping Companies Credit Named Entity Recognition. *Appl. Sci.* 2023, *13*, 5787. [CrossRef]
- Zhang, Z.; Han, X.; Liu, Z.; Jiang, X.; Sun, M.; Liu, Q. ERNIE: Enhanced language representation with informative entities. *arXiv* 2019, arXiv:1905.07129.
- 34. Sun, Z.; Li, X.; Sun, X.; Meng, Y.; Ao, X.; He, Q.; Wu, F.; Li, J. Chinesebert: Chinese pretraining enhanced by glyph and pinyin information. *arXiv* **2021**, arXiv:2106.16038.

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.