Automatic Information Extraction in the Third-Generation Semiconductor Materials Domain Based on DKNet and MANet

XIAOBO JIANG, (Member, IEEE), KUN HE, AND BORUI YANG
School of Electronic and Information Engineering, South China University of Technology, Guangzhou 510641, China
Corresponding author: Kun He (hk15616172426@163.com)

This work was supported in part by the National Natural Science Foundation of China under Grant U1801262, and in part by the Science and Technology Project Foundation of Guangdong Province under Grant 2019B010154003.

ABSTRACT
The third-generation semiconductor materials (TGSMs) is a frontier scientific domain, where researchers need to consult extensive literature for the entity information on materials, devices, preparation methods, and experimental performances, and sort out the complex relations between them. However, the explosion of relevant papers has far exceeded researchers’ reading ability. In this article, the TGSM-field automatic information extraction is conducted based on entity recognition (ER) and relation extraction (RE) techniques. First, the corpora used for ER and RE in this field are created. Second, aiming at the complexity of the entities, a neural network using domain knowledge (DKNet) is proposed to improve ER performance. It uses the keyword sequence of each entity type as prior knowledge, adds a dedicated embedding to encode entity categories, then combines prior knowledge and encoded vectors with the context through a gated information fusion module to assist recognition. As for the indicative word dependence problem of entity relations, a multi-aspect attention-based network model (MANet) is proposed to enhance the attention to relation-indicative words, thereby improving the RE performance. Finally, F1 scores of 74.5 and 85.9 were achieved on the created ER and RE test sets, outperforming other advanced models by 3.4 ∼ 10.1, which is the best performance of the TGSM-field automatic information extraction.

INDEX TERMS
Automatic information extraction, entity recognition, relation extraction, third-generation semiconductor materials, gated information fusion, multi-aspect attention.

I. INTRODUCTION
In recent years, third-generation semiconductor materials (TGSMs), represented by SiC and GaN, have been widely used in optoelectronic and microelectronic devices due to their excellent properties. During research and development of TGSMs, researchers need to consult extensive relevant literature for the entity information on materials, devices, preparation methods, and experimental performances, and to sort out the complex relations between them. However, according to the statistics of this paper, the current growth of literature in this field has exceeded 1,000 papers/month, far outpacing researchers’ reading ability. Therefore, using natural language processing (NLP) technology to conduct automatic information extraction in the TGSM domain can significantly alleviate researchers’ burden as well as assist their work on the development of TGSMs.

Entity recognition (ER) and relation extraction (RE), as two key techniques of automatic information extraction, aim to identify specific types of entities from text and classify the relations between them, thereby supporting the construction and management of domain knowledge systems. For instance, given the sentence “the magnetic properties of the GaN film doped by C was investigated”, it is required to identify two material-type entities “C” and “GaN film”, and determine the relation between them as “component-whole”. Currently, ER and RE research in the public domain (news, encyclopedias) has achieved significant progress. For example, Mu et al. [1] identified entities such as persons, places, and organizations in news texts; Guo et al. [2] extracted entity relations from encyclopedia texts. However, limited by the lack of corpus, the complexity of entities, and
indicative word dependency problem in entity relations, the research on ER and RE in the TGSM field still face great challenges.

First, in the professional scientific field, tasks of ER and RE require customized datasets with annotation standards for supervised learning, and the TGSM domain is no exception. However, according to the survey of this paper, there is no constructed corpus nor annotation specification in this field. Therefore, it is necessary to formulate a reasonable scheme for the annotation of TGSM-field entities and their relations according to the researchers’ needs, and to create the ER and RE datasets for this domain.

Second, TGSM-field entities are complex as they contain massive material science terms (e.g., “pre-coated Pt/Zn seed layer”, “metal-organic chemical vapour deposition”), which are hard-identified. Additionally, plenty of abbreviations (e.g., “IGBT”, “HEMT”) makes the ER in the TGSM field more challenging. As a result, these entities require prior domain knowledge to be well-recognized. However, existing ER models used in materials science are directly transferred from the public domain. They focus on learning features from the original text but ignore the use of prior domain knowledge, which limits the ER performance. For example, Friedrich et al. [3] used a prevalent ER model in the public domain (i.e., the pre-trained BERT [4] combined with a bidirectional long short-term memory (BiLSTM) and a condition random field (CRF) layer) to recognize entities in the solid oxide fuel cells (SOFCs) domain without using prior knowledge about solid oxide materials. Weston et al. [5] employed a similar model to identify inorganic material entities without using domain knowledge either.

![FIGURE 1. Indicative word dependency of the RE in the TGSM field.](image)

Third, entity relations in the TGSM field have indicative word dependency (i.e., the relation between two entities in a sentence is related to several indicative words), as shown in Figure 1(a), the relation between two material entities can be judged by “fabricated on”. Therefore, it is worth paying more attention to relation-indicative words (RIWs). Some RE work used the attention mechanism to adjust attention to each word [6], [7]. They perform self-attention on the entire sentence, so that RIWs in the same sentence are unchangeable. However, in the TGSM field, many same sentences have distinct RIWs due to different aimed entities. For instance, the sentence in Figure 1(b) is the same as in (a), but the entities differ, so the focused RIWs should be changed to “using”. Therefore, it is necessary to design an attention mechanism, which can focus on different RIWs according to specific entities, to improve the RE performance.

To overcome these challenges, in this work, two datasets named TGSM-ER and TGSM-RE are created to support the ER and RE tasks in the TGSM domain. Two neural network models, named DKNet (Domain Knowledge Network) and MANet (Multi-aspect Attention Network), are designed to implement the ER and RE tasks, respectively.

1) The TGSM-ER and TGSM-RE datasets are manually annotated. The annotation scheme is discussed by the authors and several invited TGSM-filed researchers.

2) The DKNet model aims at the TGSM-field entities with complex terms, and uses prior domain knowledge to improve the ER performance. The prior knowledge is obtained by pre-designing keyword sequences of the entity types (e.g., the keyword sequence of the “method” entity type is “method, approach, processing, evaporation, etching, sputtering, aging, coating, annealing, doping”). Each of them is concatenated with the original text sentence as the model input. Meanwhile, each entity type is encoded to a vector by an entity type embedding (ETE). Then, a designed GIF (Gated Information Fusion) module combines prior knowledge and entity type vectors with context features to assist the ER.

3) The MANet model aims at the RIWs dependency of the RE in the TGSM domain, and uses a designed multi-aspect attention (MAA) mechanism to enhance the attention to the RIWs and improve the RE performance. Based on the self-attention of the input text sentence, MAA further makes the context interact with the entity information and entity type vectors to adaptively change RIWs in the same sentences with distinct entities.

The experimental results show that the F1 scores, achieved on the created TGSM-ER and TGSM-RE datasets, are 74.5 and 85.9, respectively, which are the best performance of the TGSM-field automatic information extraction, outperforming other advanced models by 3.4 ~ 10.1.

II. RELATED WORKS

A. AUTOMATIC INFORMATION EXTRACTION

Automatic information extraction has now been widely used in the public and biomedical fields. Mu et al. [1] identified persons, places, organizations, and other types of entities in news texts. Guo et al. [2] extracted the cause-effect, content-container, and other entity relations from encyclopedia text. Chen et al. [8] carried out the biomedical ER to recognize proteins, DNAs, RNAs, and other biomedical entities from domain literature. Eberts and Ulges [9] identified drug and disease entities from medical documents, and extracted their adverse effect relations.

In contrast, the materials science domain lacks research on automatic information extraction, which needs to be further developed. In previous works, Friedrich et al. [3] identified materials, devices, properties, and other entities from solid oxide materials literature. Weston et al. [5] recognized
inorganic materials, properties, and application entities from relevant papers. This paper conducted automatic information extraction in the TGSM field, including corpus construction and model research.

B. ENTITY RECOGNITION
Mainstream ER models use the sequence tagging method to tag each word in the input sentence with “entity type + BIO (Begin, Inner, Other)”, and a sequence of words with the same tag is the identified entity. Le et al. [10] used BiLSTM for the ER task based on sequence tagging. Lample et al. [11] combined BiLSTM with a CRF layer to improve the tagging behavior. Zhao et al. [12] further merge the BiLSTM + CRF model with a convolutional neural network (CNN). In recent years, large-scale pre-trained models have been widely used in ER to improve performance. Devlin et al. [4] applied the pre-trained BERT for the ER. Friedrich et al. [6] added the attention mechanism to the ER task. They enumerate all spans (i.e., subsequences) from the input sentence, and then classify a span as an entity or non-entity. Eberts and Ulges [9] proposed a span-based model SpERT. Wang et al. [14] jointly used BERT and CNN to incorporate local information into the span representation. Shen et al. [15] combined BERT, BiLSTM, and SoftNMS algorithm to improve the classification.

However, these ER models use only context information, limiting the performance in some scientific fields containing massive complex terms, such as the TGSM domain. The DKNet model presented in this paper utilized the extra prior domain knowledge to improve the cognition of domain terms. Besides, it has dual pointer to directly find the start and end positions of the entities in a sentence, which is friendly to the text with dense entities.

C. RELATION EXTRACTION
In the early related works, LSTM and CNN are widely used in the RE task. Xu et al. [16] combined the LSTM with the shortest dependency path (SDP) algorithm. Santos et al. [17] designed a CNN model based on a ranking algorithm. Cai et al. [18] jointly used the LSTM and CNN for RE. With the raising importance of the attention mechanism, researchers introduced it in the RE task to adjust the model’s attention to different words and improve performance. Zhang et al. [6] added the attention mechanism to the gated recurrent unit (GRU). Guo et al. [2] combined the attention mechanism with GRU and CNN. Yin et al. [7] also designed a RE model that merges the BiLSTM, CNN, and the attention mechanism together. Currently, pre-trained models are widely applied in the RE. Wu et al. [19] proposed an R-BERT model specific for the RE task. Soares et al. [20] used an optimized BERT model combined with a proposed filling-in-the-blank method to better perform in the RE.

These models perform self-attention on the entire sentence, leading to the unchanged RIWs in the same sentences with different aimed entities. However, in the TGSM field, there are many same sentences whose aimed entities and RIWs differ. The MANet designed in this paper uses specific entity information and entity type vectors to enhance the attention to distinct RIWs in the same sentences.

III. CORPUS AND DATASETS CONSTRUCTION
The first TGSM-field corpus is built in this work, and two datasets, TGSM-ER and TGSM-RE, for supervised learning of ER and RE tasks in this field are created, respectively. The workflow is shown in Figure 2, including three main steps, preprocessing, annotation, and evaluation.

The TGSM-field literature is obtained from IEEE Xplore, Nature, and Science databases, with a total of 500 papers. The machine learning library named GROBID1 is used to parse these PDF documents to the structured ones with XML format, and the Xpath parsing is responsible for converting them into TXT format. During preprocessing, each document is split into sentences, and those with less than 10 words or more than 50 words are deleted. Text noise such as special characters, quotation marks, and URL links are also deleted or replaced to obtain a cleaned TGSM-field corpus. In the annotation stage, several researchers in the TGSM field are invited to formulate annotation schemes of the entities and their relations together with the authors of this paper. After three months of work, the ER and RE datasets in the TGSM field, named TGSM-ER and TGSM-RE, respectively, finally came to fruition. To check the annotation quality, Cohen’s kappa coefficients of annotator agreement are calculated on two created datasets. The results show that scores are as high as 94 and 91, respectively.

A. TGSM-ER DATASET
The TGSM-ER dataset supports the ER task in the TGSM domain. It contains 5,500 data (i.e., sentences) and defines 4 entity types, namely material (MAT), device (DEV), method (MET), and performance (PER):
- MAT: TGSMs, e.g., “GaN”, “SiC”, and “LiNbO3”.
- DEV: devices based on TGSMs, e.g., “fast-switching SiC transistor” and “GaN HEMT power device”.

1https://github.com/kermitt2/grobid.
• MET: methods for preparation and manufacturing of materials and devices, e.g., “plasma-based dry etching” and “RF magnetron sputtering”.
• PER: the experimental performance, e.g., “piezoelectric property” and “chemical and thermal stability”.

**FIGURE 3.** The proportion of each type of entities in the TGSM-ER (a) and the proportion of each entity relation in the TGSM-RE (b).

The above-mentioned 4 types of entities, a total of 14,856, are generally the most concerned by TGSM-field researchers. The proportion of each type of entities is shown in Figure 3(a). The training set, validation set, and test set of the TGSM-ER are divided by 6:2:2.

### B. TGSM-RE DATASET

The TGSM-RE dataset supports the RE task in the TGSM field. It contains 9,000 text sentences and defines 5 relation types, namely **component-whole (CW)**, **producer-product (PP)**, **product-characteristic (PC)**, **product-application (PA)**, and **others (OT)**:

- **CW**: an altogether of the relation between a material and the material in which it is doped or grown, the cooperation relation between devices, and the inclusion relation between methods.
- **PP**: the relation between the method and the materials or devices prepared therefrom.
- **PC**: the relation between materials, devices or methods and the properties they provide.
- **PA**: the application relation between materials and their fabricated devices.
- **OT**: an altogether of relations other than the above.

The above 5 types of relations cover the relation between entities in the TGSM field from a high-level semantics. The proportion of each type of relation is shown in Figure 3(b). The training set, validation set, and test set of the TGSM-RE are also divided by 6:2:2.

## IV. METHODOLOGY AND MODEL DESIGN

### A. DKNet Model

The DKNet model is proposed to perform the ER task in the TGSM domain, which is defined as follows: Given an input sentence \( S \) and an entity type set \( C = \{c_i\}_{i=1,2,3,4} \), identify all entities from \( S \) and correctly classify them into a certain type in \( C \) to make an entity set \( E = \{(e_i, c_j)\}_{e_i \in T, c_j \in C} \).

In NLP, the input sentence needs to be tokenized first to get the input sequence \( X = \{w_1, w_2, \ldots, w_L\} \), where \( L \) is the sequence length. The model uses the start and end positions of the entity in \( X \) to represent this entity, i.e., \( e_i \leftrightarrow (p_{si}, p_{ei}) \). For example, the device entity “PIN diodes” in an input sequence “The required PIN diodes were obtained from Cree, Inc.” is represented as “(3, 4)”.

To use prior domain knowledge to assist recognition, the keyword sequence \( K = \{k_1, k_2, \ldots, k_M\} \) for each entity type, where \( M \) is the number of keywords, is pre-designed as prior knowledge, and a keyword sequence set \( I = \{K_i\}_{i=1,2,3,4} \) in the TGSM field is obtained, as shown in Table 1.

**TABLE 1.** The keyword sequences of entity types in the TGSM field.

| Entity Type | Keyword Sequence |
|-------------|-----------------|
| material (MAT) | “material, SiC, GaN, ZnO, AlN, SiO2, Si, N-type, P-type, polymer, film, surface, bandgap, crystal” |
| device (DEV) | “device, electronic, photonic, diode, transistor, sensor, MOSFET, FBAR, IGBT, HEMT, SAW, nanorod” |
| method (MET) | “method, approach, processing, evaporation, etching, sputtering, aging, coating, annealing, doping” |
| performance (PER) | “performance, conductivity, resistivity, velocity, frequency, mechanical, thermal, stability, power, energy” |

**FIGURE 4.** The overall architecture of the DKNet model.

The DKNet model recognizes different types of entities separately. Its overall architecture is shown in Figure 4, including an embedding module, a gated information fusion (GIF) module, and a filter module. The model inputs are the entity type label \( c \) and a concatenated sequence of \( K \) and \( X \). The outputs are two \( L \)-dimensional vectors, namely \( p_{si} \) and \( p_{ei} \). The \( i \)-th element of \( p_{si} \) is equal to the probability that the start position of the entity is \( i \), and the \( j \)-th element of \( p_{ei} \) is equal to the probability that the end position of the entity is \( j \).
is equal to the probability that the end position of the entity is \( j \). Using the outputs, the \( c \)-type entity set \( E \) can be obtained, and the total entity set is \( E = \{(E_i, c_i)|i = 1, 2, 3, 4\} \).

1) EMBEDDING MODULE

The embedding module has two functions: Encoding words in the input sequence into vectors (i.e., word embedding) and encoding the input entity type \( c \) into vectors (i.e., entity type embedding).

The input sequence \( X_{in} \) is a concatenation of the keyword sequence \( K \) and the original text sequence \( X \):

\[
X_{in} = ([CLS], K, [SEP], X, [SEP])
\]

where the special word “[CLS]” is employed to represent the semantic information of the entire sequence, and “[SEP]” is used to separate \( K \) and \( X \).

Before word embedding, the byte pair encoding (BPE) algorithm [21] is used for fine-grained tokenization on the input sequence \( X_{in} \). BPE decomposes infrequent words into ordinary subwords. For example, the word “nanotube” can be decomposed to “nano” and “tube”, effectively alleviating the OOV (Out Of Vocabulary) problem in word embedding.

Then, the pre-trained model DeBERTa [22] is used for word embedding on the tokenized input sequence to obtain the context representation \( h_o \):

\[
h_o = \text{DeBERTa}(\text{BPE}(X_{in})) \in \mathbb{R}^{l \times d}
\]

where \( l \) is the length of the tokenized input sequence and \( d \) is the dimension of word vectors. Assuming that the lengths of \( K \) and \( X \) after BPE tokenization are \( l_k \) and \( l_x \), respectively, then \( l = l_k + l_x + 3 \).

DeBERTa is used instead of other widely used pre-trained models (e.g., BERT and SciBERT) for two main reasons. First, DeBERTa uses a disentangled attention mechanism to represent each word using two vectors of its content and position, considering that the attention weight between two words depends on not only their contents but also their relative positions. This mechanism is conducive to the ER task in the TGSM field, which is found in this paper. Second, DeBERTa uses an enhanced mask decoding mechanism at the output layer, which alleviates the mismatch between pretraining and fine-tuning.

In addition to word vectors, the vector of entity type is also involved, and a dedicated embedding matrix \( \text{ETE} \) with the size of \( 4 \times d \) is added to the module. \( \text{ETE} \) is adaptively learned by backpropagation during model training and maps the input entity type \( c \) into a \( d \)-dimensional vector representing the general features of \( c \)-type entities:

\[
h_c = \text{ETE}(c) \in \mathbb{R}^d
\]

2) GIF MODULE

The GIF module combines the prior domain knowledge, the vector of entity type, and context features to enhance the cognition of domain entities, therefore improving the ER performance. The module aims to gain the initial probability vector \( p_0 \) of the entity position.

First, the word vectors belonging to the keyword sequence \( K \) are converted to a representation \( h_k \) of the prior knowledge through the average pooling:

\[
h_k = \text{AvgPooling}(\{h_o[i]|i \in [2, l_k + 1]\}) \in \mathbb{R}^d
\]

Then, \( h_o \) is transformed into \( h_{ok} \) and \( h_{ac} \) through two linear transformations with the Gaussian error linear unit (GELU) function, and a matrix-vector multiplication is conducted on \( h_{ok} \) and \( h_k \) to form an interactive attention score \( s_k \). Similarly, the score \( s_c \) of \( h_{oc} \) and \( h_c \) can be obtained.

\[
\text{gelu}(x) = \frac{1}{2}x[1 + \frac{2}{\sqrt{\pi}}\int_0^x e^{-t^2}dt]
\]

\[
s_k = \text{gelu}(h_o \cdot W_k + b_k) \cdot h_k^T \in \mathbb{R}^l
\]

\[
s_c = \text{gelu}(h_o \cdot W_c + b_c) \cdot h_c^T \in \mathbb{R}^l
\]

Subsequently, \( h_o \) is converted into a vector \( h_{ag} \) with a length of \( l \) through linear transformation, and each element in \( h_{ag} \) is scaled to 0 \~ 1 through the sigmoid function to form a weight vector \( g_w \). It is used for the gated fusion between \( s_k \) and \( s_c \) to obtain the fused interactive result \( s_f \).

\[
g_w = \frac{1}{1 + \exp[-(h_o \cdot v_g + b_g)]} \in \mathbb{R}^l
\]

\[
s_f = g_w \odot s_k + (1 - g_w) \odot s_c \in \mathbb{R}^l
\]

where “\( \odot \)” denotes the element-wise multiplication.

Finally, the elements in \( s_f \) are normalized to 0 \~ 1 through a proposed \( \text{talu} \) function to form an initial probability vector \( p_0 \) of the entity position:

\[
p_0 = \frac{\exp(s_f)}{\exp(s_f) + \exp(-s_f)} \in \mathbb{R}^l
\]

where the derivative value of the \( \text{talu} \) function at the zero point is twice that of the widely used \( \text{sigmoid} \) function, which helps distinguish words with approximate scores.

Two GIA modules with distinct parameters are contained within the DKNet model to follow the above steps. As a result, two initial probability vectors of the start and end entity positions are obtained, i.e., \( p_{s0} \) and \( p_{e0} \), respectively. Specifically, the \( i \)-th element in \( p_{s0} \) (or \( p_{e0} \)) represents the probability that the \( i \)-th word is in the start (or end) position of an entity.

3) FILTER MODULE

The filter module aims to filter the elements in \( p_{s0} \) and \( p_{e0} \), therefore, some impossible start (or end) entity positions can be excluded from consideration.

First, since the specific words “[CLS]”, “[SEP]”, and the keyword sequence \( K \) contain no entities, the relevant parts in \( p_{s0} \) and \( p_{e0} \) are dropped out, and the \( p_{s1} \) and \( p_{e1} \) are gained.

Second, by considering that BPE make some infrequent words apart into multiple subwords, only the subword with the highest probability is reserved, and others are deleted.
from $p_{r1}$ and $p_{r2}$ to obtain the final outputs $p_s$ and $p_e$. Meanwhile, the lengths of $p_s$ and $p_e$ are set to $L$.

Finally, the filter module uses a specific rule to obtain the entity set $E$ with the target type $c$. The pseudocode for this rule is summarized in Algorithm 1.

**Algorithm 1 Entity Set $E$ Acquisition Rule**

**Input:** probability vectors $p_s$ and $p_e$; sequence length $L$; **Output:** entity set $E$;
1: Initialize an empty set $E = []$, a probability threshold $P_t$, and an upper limit of entity length $l_m$;
2: for $i$ from 0 to $L$ do
3:    for $j$ from $i$ to $L$ do
4:        if $p_s[i] \geq P_t$, $p_e[j] \geq P_t$, and $j - i \leq l_m$, then
5:            add the coordinate $(i, j)$ to $E$;
6:            break out of the iterations of $j$;
7:        end if
8:    end for
9: end for
10: return $E$;

**B. MANET MODEL**

The MANet model is proposed to perform the RE task in the TSGM domain, which is defined as follows: Given an entity relation type set $R = \{r_i | i = 1, 2, 3, 4, 5\}$ and a sentence $S$ with two identified entities $e_1$ and $e_2$, classify the relation between $e_1$ and $e_2$ into a certain category in $R$. Similarly, the input sentence is tokenized first to form an input sequence $X$.

![The overall architecture of the MANet model.](image)

The overall architecture of the MANet model is shown in Figure 5, including an embedding module and a multi-aspect attention (MAA) module. The model inputs are the sequence $X$, and two entity type label $c_1$ and $c_2$, corresponding to $e_1$ and $e_2$, respectively. The output is a 5-dimensional vector $p_r$. The $i$-th element of $p_r$ is equal to the probability that the relation between $e_1$ and $e_2$ is $r_i$.

1) **EMBEDDING MODULE**

The embedding module in the MANet model is similar to that in the DKNet model. It also consists of word embedding and entity type embedding. Added by a special word "[CLS]" and some words to label $e_1$ and $e_2$, the input sequence $X_{in,r}$ has its expression as follows:

$$X_{in,r} = \{[CLS], \ldots, e_1, 1/[e_1], \ldots, [e_2], e_2, /[e_2], \ldots\}$$

After the fine-grained tokenization by BPE, $X_{in,r}$ is fed to DeBERTa for word embedding, and a context representation $h_{out}$ with the size of $l \times d$ is obtained. Meanwhile, the input $c_1$ and $c_2$ are encoded into two $d$-dimensional vectors, $h_{c1}$ and $h_{c2}$, through ETE, respectively.

2) **MAA MODULE**

The MAA module uses the proposed interactive attention (I-Att) algorithm to interact entity information and entity type vectors with context features, respectively, to enhance the attention on RIs, thereby improving RE performance.

First, the entity representations $h_{e1}$ and $h_{e2}$ of $e_1$ and $e_2$, respectively, and the global semantic vector $h_{cls}$ of "[CLS]" are separated from the whole sequence representation $h_{out}$. The sizes of $h_{e1}$ and $h_{e2}$ are $l_e \times d$ and $l_e \times d$, respectively, where $l_e$ and $l_e$ refer to the length of $e_1$ and $e_2$. $h_{cls}$ is a $d$-dimensional vector.

Then, two independent I-Att calculations are performed on $h_{e1}$ and $h_{e2}$ with $h_{out}$, respectively, and two interactive results $h_{i1}$ and $h_{i2}$ are obtained:

$$h_{i1} = l - \text{Att}_1(h_{out}, h_{e1}) \in \mathbb{R}^d$$
$$h_{i2} = l - \text{Att}_2(h_{out}, h_{e2}) \in \mathbb{R}^d$$

where the pseudocode for the I-Att calculation is summarized in Algorithm 2.

**Algorithm 2 Interactive Attention Algorithm**

**Input:** context representation $h_{out}$, entity representation $h_e$; **Output:** interactive result $h_i$;
1: Initialize two trainable matrices, $W_o$ and $W_e$, and two trainable bias, $b_o$ and $b_e$;
2: $h_{ot} \leftarrow \text{gelu}(h_{out} \cdot W_o + b_o) \in \mathbb{R}^{l \times d}$;
3: $h_{cl} \leftarrow \text{gelu}(h_e \cdot W_e + b_e) \in \mathbb{R}^{l \times d}$;
4: $s_{em} \leftarrow \text{MaxPooling}_{\text{column}}(h_{ot} \cdot h_{cl}^T) \in \mathbb{R}^{l}$;
5: $s_{em} \leftarrow \frac{\exp(s_{em})}{\sum_{i=1}^{D} \exp(s_{em})} \in \mathbb{R}^{l}$;
6: $h_i \leftarrow s_{em} \cdot h_{out} \in \mathbb{R}^{d}$;
7: return $h_i$;

Subsequently, five $d$-dimensional vectors, i.e., $h_{cls}$, $h_{i1}$, $h_{i2}$, $h_{e1}$, and $h_{e2}$ are concatenated together into a $5d$-dimensional vector $h_{cat}$, which is further converted into
respectively. Specifically, two their loss functions, which are shown in Eq. (16) and Eq. (16), update the parameters of DKNet and MANet to minimize The backpropagation (BP) algorithm is used to train and update the parameters of DKNet and MANet to minimize their loss functions, which are shown in Eq. (16) and Eq. (16), respectively. Specifically, two L-dimensional binary vectors, \( y \) and \( y^r \), are the real start and end position distributions of entities. The i-th element in \( y \) (or \( y^r \)) is equal to 1 only if \( i \) is the start (or end) position of an entity, otherwise to 0. The 5-dimensional one-hot vector \( h \) is the real distribution of the entity relation.

\[
\begin{align*}
\text{loss}_1 &= - \sum_{i=1}^{L} [y_i \ln p_{si} + (1 - y_i)(1 - \ln p_{si}) \\
\text{loss}_2 &= - \sum_{i=1}^{5} r_i \ln p_{ri}
\end{align*}
\]

(14)

C. MODEL TRAINING

The backpropagation (BP) algorithm is used to train and update the parameters of DKNet and MANet to minimize their loss functions, which are shown in Eq. (16) and Eq. (16), respectively. Specifically, two \( L \)-dimensional binary vectors, \( y \) and \( y^r \), are the real start and end position distributions of entities. The i-th element in \( y \) (or \( y^r \)) is equal to 1 only if \( i \) is the start (or end) position of an entity, otherwise to 0. The 5-dimensional one-hot vector \( h \) is the real distribution of the entity relation.

\[
\begin{align*}
\text{loss}_1 &= - \sum_{i=1}^{L} [y_i \ln p_{si} + (1 - y_i)(1 - \ln p_{si}) \\
\text{loss}_2 &= - \sum_{i=1}^{5} r_i \ln p_{ri}
\end{align*}
\]

(16)

Finally, the probability function \( \text{softmax} \) is used to process \( f \) through linear compression transformation.

\[
h_f = [h_{cls}|h_{i1}|h_{i2}|h_{i1}|h_{c2}] \cdot W_f + b_f \in \mathbb{R}^5
\]

(14)

V. EXPERIMENTS AND ANALYSIS

In this section, experiments are conducted on the constructed TGSM-ER and TGSM-RE datasets first, which is to verify the feasibility of using ER and RE techniques for automatic information extraction in the TGSM field and prove the effectiveness of the proposed method and models. Second, an ablation study is carried out, and the contribution of each module in designed models to the performance improvement is analyzed in detail.

A. HYPERPARAMETER SETTING AND EVALUATION METRICS

The experiments are conducted by using a GTX1080Ti GPU, and the hyperparameter setting is shown in Table 2.

| Hyperparameter | Value |
|----------------|-------|
| vocabulary size for BPE tokenization | 50265 |
| embedding size \( d \) (i.e., vector dimension) | 768 |
| batch size of data for training | 16 |
| learning rate for training | \( 3 \times 10^{-6} \) |
| weight decay rate for training | 0.01 |
| dropout rate for model | 0.3 |
| probability threshold \( P \) | 0.5 |
| upper limit of entity length \( L \) | 10 |

The evaluation metrics used for ER and RE tasks are precision (P), recall (R), and F1-score (F1), whose formulaic expressions are shown in Eq. (18)-(20).

\[
\text{Precision} = \frac{\text{NTP}}{\text{NTP} + \text{NFP}}
\]

(18)

where \( \text{NTP} \) represents the number of samples that the model predicts correctly, \( \text{NFP} \) represents the number of incorrectly predicted samples, and \( \text{NFN} \) represents the number of missed correct samples.

By considering that a certain difference is in the number of samples of various types, the macro-averaging approach is adopted for the average calculation of P, R, and F1. It first calculate the performance of each type of samples, and then average the results of all types.

B. RESULTS ON TGSM-ER AND TGSM-RE

In this section, the ER experiment is implemented on the constructed TGSM-ER dataset using the proposed DKNet model, and the result is shown in Figure 6. The designed RE model MANet is conducted on the created TGSM-RE dataset, and the result is shown in Figure 7. During multiple epochs of supervised training, the ability of the two proposed model gradually strengthened and stabilized, and finally achieved 74.5 and 85.9 F1 scores of ER and RE tasks on the test sets, respectively, according to the best results on validation sets. It has reached the expected goal of this work, proving the feasibility and effectiveness of the TGSM-domain automatic information extraction using ER and RE techniques.

Then, to verify the superiority of the proposed models and methods for automatic information extraction in the TGSM domain, 10 representative and advanced ER and RE models, selected for comparison, are trained and evaluated on the created TGSM-ER and TGSM-RE datasets.

The ER models for comparison are:
- **BiLSTM**: An optimized recurrent neural network (RNN) model widely used in ER to extract context features.
- **BiLSTM + CRF**: BiLSTM concatenated with a CRF layer at the output to enhance the sequence tagging.
- **BiLSTM + CNN + CRF**: An CNN-enhanced BiLSTM + CRF model that combines local features.
- **BERT**: An mainstream advanced ER model based on the pre-trained BERT.
- **SpERT**: A typical high-performance span-based model for ER task, proposed by Eberts and Ulges [2].

The RE models for comparison are:
- **BiLSTM + CNN**: A simple but effective RE model that combines BiLSTM and CNN.
- **BiLSTM + Att**: A BiLSTM-based RE model with the attention mechanism to capture context features.
- **BiLSTM + CNN + Att**: A hybrid model jointly using BiLSTM, CNN, and attention mechanism.
- **BERT + CNN**: An advanced and prevalent RE model combining BERT and CNN.
- **R-BERT**: A high-performance model optimized for RE task, designed by Wu et al. [19].

The experimental results of the above-mentioned models on the TGSM-ER and TGSM-RE datasets are shown in...
Table 3 and Table 4, respectively. The best F1-score of 74.5 on TGSM-ER is achieved by the proposed DKNet, which outperforms other comparative models by 3.9 ∼ 10.1, proving its superiority for the ER task in the TGSM field. The best RE performance, a F1 score of 85.9, is also achieved by the designed MANet model, which is 3.4 ∼ 7.6 higher than other comparison models, verifying its superiority for the TGSM-domain RE task.

Table 3 and Table 4 are discussed in detail. First, in the ER task, BiLSTM and CNN effectively capture global and local context features, but lack high-semantic-quality word vectors as support, moderating the performance. The BERT model provides superior word vectors pre-trained on large-scale corpora, and it achieves high performance by combining BiLSTM and CRF. However, these word vectors are trained on the text with general knowledge, and when directly transferring to the TGSM field, there is still a lack of cognition of domain terms that limits the performance. The span-based SpERT suffers from the same problem. The proposed DKNet, based on the pre-trained model, further utilizes the prior TGSM-domain knowledge to assist the recognition, and the best performance is achieved. In addition, the ER performance of the SpERT model is slightly higher than that of the BERT + BiLSTM + CRF, as it is pointed out that span-based models have a certain advantage on dense entities [23], which reveals a dense entity distribution of the TGSM-field literature.

Second, in the RE task, experimental results show that the attention mechanism has an advantage over CNN, and the joint use of the two does not markedly raise the F1 score. The BERT + CNN model, which improves the semantic quality of RIWs vectors, makes a significant RE performance boost. The R-BERT model further combines BERT with the entity information in the sentence, allowing its performance better than BERT. Inspired by R-BERT, the designed MANet model also utilizes entity information, but there are two chief differences between the two. On the one hand, MANet uses the interactive results between entities and the context, which is more efficient than independent entity information used by R-BERT. On the other hand, MANet additionally utilizes the category information of the entity, which is motivated by observing that knowing the types of two entities is beneficial for classifying the relation between them. These make the MANet model achieve the best RE performance.

The performances of different types of samples are further analyzed, and the results are shown in Figure 8 and Figure 9. Figure 8 illustrates the ER performance of different types of samples.
entities in the TGSM-ER dataset using the DKNet model, and Figure 9 shows the RE performance of distinct entity relations in the TGSM-RE dataset using the MANet model. For the ER task, “material” entities are the easiest to identify, with an F1 score of 83.0, whereas the “performance” entities are the most difficult to recognize, and its F1 score is only 64.4. For the RE task, the “product-application” relation is easy to extract, with the F1 score up to 93.3, whereas the “component-whole” relation (F1 = 81.9) and the “others” (F1 = 75.8) are harder to extract.

The results in Figure 8 and Figure 9 are discussed in detail. First, in the ER task, most TGSMs are composed of “SiC”, “GaN”, and “ZnO”, and the DKNet can identify “material” entities according to these words. However, the performance indicators of materials and devices in the TGSM domain are numerous and complex. It is difficult for the model to capture some high-frequency words or explicit features, resulting in an unsatisfactory recognition behavior on “performance” entities. Second, in the RE task, the “others” relation mixes all the remaining fine-grained relations excepted from the main ones, it is complex and diverse in semantics and hard to extract. Compared with the other three relations, the “component-whole” relation is more logical and harder to distinguish, whereas the rest of the three are closer to shallow semantics, which can usually be well-identified according to the entities and their types in the sentence.

C. RESULTS OF ABLATION STUDIES
The ablation studies aim to verify the effectiveness of each part of the proposed models for performance improvement. The DKNet model consists of three parts, namely an embedding module (including ETE and DeBERTa), a GIF module, and a filter module, and its ablation result is shown in Table 5. The MANet model comprises two parts, namely an embedding module and a MAA module, and its ablation result is shown in Table 6.

First, the results in Table 5 are discussed in detail. After removing the dedicated embedding ETE, the performance of ER in the TGSM field drops by 1.6, verifying the aid of ETE to capture features of different entity types. When replacing DeBERTa with the widely used BERT, the model degrades in performance by 2.9, but still outperforms other BERT-based models. It proves that the DKNet’s effectiveness is not entirely owing to DeBERTa, and also reveals DeBERTa’s advantage over BERT in the TGSM-domain ER task. The removal of the GIF module brings in the most performance drop, up to 3.4, showing the effectiveness and weightiness of the GIF module using extra prior domain knowledge to assist recognition. With the absence of the filter module, the F1 score also decreases by 1.5, which justifies the filtering step necessary for the ER task.

Second, the results in Table 6 are detailedly discussed. After deleting ETE, the TGSM-field RE performance drops by 2.2, which proves that the entity type information is conducive for the classification of entity relations. Replacing DeBERTa with BERT also leads to an F1-score decrease of 3.1 on RE performance, but the overall performance still surpasses that of the BERT and R-BERT models, which also shows the merit of DeBERTa for TGSM-field RE task and the MANet’s effectiveness not relying totally on DeBERTa. When the MAA module is removed, the performance drops most by 3.7, which verifies the usefulness and importance of the proposed multi-aspect attention mechanism for the RE task in the TGSM field.

The results of the ablation study demonstrate that each part of the proposed models plays a significant role of automatic information extraction in the TGSM domain, and there is no redundant design.

VI. CONCLUSION
To help TGSMs researchers efficiently obtain their desired entity and relation information from extensive literature, in this paper, the ER and RE techniques, based on proposed DKNet and MANet models, respectively, have been used to perform automatic information extraction in the TGSM field, and two supporting datasets TGSM-ER and TGSM-RE were created. Aiming at the entities with complex TGSM-domain terms, the DKNet used a novel GIF module, which combines
the prior domain knowledge with adaptively learned features of entity types, to assist recognition. Therefore, the best F1 score of 74.5 was achieved on the TGSM-ER, outperforming related models by 3.9 ~ 10.1. As for the RIWs dependence problem of entity relations, MANet used a proposed multi-aspect attention mechanism to enhance the attention on RIWs to improve RE performance. As a result, the best F1 score of 85.9, achieved on TGSM-RE by the MANet, is 3.4 ~ 7.6 higher than other RE models. Ablation studies further prove the effectiveness of the proposed models.

The methodology and achievement of this article can be used in future work to automatically construct a knowledge graph of TGSMs research and further perform knowledge reasoning. It will promote intelligent knowledge management in TGSMs studies and provide domain researchers with some inspiration. To this end, the built TGSM-field datasets ought to be expanded, and the proposed models also leaves much development space for continued optimization.

REFERENCES

[1] M. Xiaofeng, W. Wei, and X. Aiping, “Incorporating token-level dictionary feature into neural model for named entity recognition,” Neurocomputing, vol. 375, pp. 43–50, Jan. 2020.
[2] X. Guo, H. Zhang, H. Yang, L. Xu, and Z. Ye, “A single attention-based combination of CNN and RNN for relation classification,” IEEE Access, vol. 7, pp. 12467–12475, 2019.
[3] A. Friedrich, H. Adel, F. Tomazic, J. Hingerl, R. Benteau, A. Maruszczyk, and L. Lange, “The SOFC-exp corpus and neural approaches to information extraction in the materials science domain,” in Proc. 58th Annu. Meeting Assoc. Comput. Linguistics, 2020, pp. 1255–1268.
[4] J. Devlin, M. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., Jun. 2019, pp. 4174–4186.
[5] L. Weston, V. Tishby, J. Dagan, O. Kononova, A. Twarwaria, K. A. Persson, G. Ceder, and A. Jain, “Named entity recognition and normalization applied to large-scale information extraction from the materials science literature,” J. Chem. Inf. Model., vol. 59, no. 9, pp. 3692–3702, Jul. 2019.
[6] Y. Zhang, X. Li, and Z. Zhang, “Disease-pertinent knowledge extraction in online health communities using GRU based on a double attention mechanism,” IEEE Access, vol. 8, pp. 95947–95955, 2020.
[7] B. Yin, Y. Sun, and Y. Wang, “Entity relation extraction method based on fusion of multiple information and attention mechanism,” in Proc. IEEE 6th Int. Conf. Comput. Commun. (ICCC), Dec. 2020, pp. 2485–2490.
[8] Y. Chen, Y. Hu, Y. Li, R. Huang, Y. Qin, Y. Wu, Q. Zheng, and P. Chen, “A boundary assembling method for nested biomedical named entity recognition,” IEEE Access, vol. 8, pp. 214141–214152, 2020.
[9] M. Eberts and A. Ulges, “Span-based joint entity and relation extraction with transformer pre-training,” in Proc. 24th Eur. Conf. Artif. Intell., 2020, pp. 2006–2013.
[10] P. Le-Hong, M. P. Q. Nhat, T.-H. Pham, T.-A. Tran, and D.-M. Nguyen, “An empirical study of discriminative sequence labeling models for Vietnamese text processing,” in Proc. 9th Int. Conf. Knowl. Syst. Eng. (KSE), Oct. 2017, pp. 88–93.
[11] G. Lample, M. Ballesteros, S. Subramanian, K. Kawakami, and C. Dyer, “Neural architectures for named entity recognition,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., 2016, pp. 260–270.
[12] L. Zhao, X. Qiu, Q. Zhang, and X. Huang, “Sequence labeling with deep gated dual path CNN,” IEEE/ACM Trans. Audio, Speech, Language Process., vol. 27, no. 12, pp. 2326–2335, Dec. 2019.
[13] I. Beltagy, K. Lo, and A. Cohan, “SciBERT: A pretrained language model for scientific text,” in Proc. Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process. (EMNLP-IJCNLP), 2019, pp. 3615–3620.