ABSTRACT
Materials informatics requires large-scale collection and analysis of material synthesis procedures described in the literature for designing materials using computational methods. However, existing studies have not performed the paragraph-level analysis of the procedures. Moreover, since most of the synthesis procedures are described in natural language in articles and technical documents, it is necessary to structure them in a format that can be handled by computers through information extraction. Therefore, in this study, we construct a pipeline system that extracts synthesis procedures from text in the form of a flow graph and analyzes each procedure as a flow graph rather than a set of processes. The extraction system extracts entities by the deep learning model and relations between entities by the rule-based extractor from all paragraphs in the literature and selects procedures that include valid structures of entities and relations. Our evaluation of a benchmark dataset gave micro-averaged F-scores of 0.807, 0.830, and 0.609 for the entity extractor, relation extractor, and pipeline extractor, respectively. We applied this system to a large amount of literature and extracted approximately 90,000 flow graphs (procedures) containing approximately 4 million entities and 3 million relations. We performed several analyses, including taking statistics of the extracted graphs and checking frequent subgraphs for the extracted graphs. Commonly used methods in materials science were confirmed from our analyses; for example, ethanol is often dried by heating at 60 °C, and less-reactive noble gases are rarely included in the products. As a result, we experimentally confirmed that the extracted procedures were reasonable.

INDEX TERMS
Artificial intelligence, data mining, information extraction, materials, materials science and technology, materials informatics, natural language processing, neural networks.

I. INTRODUCTION
Materials informatics has been attracting attention in the development of new materials by analyzing existing material properties and synthesis procedures for computational materials design. Materials informatics contributes to the reduction of development costs by reducing the number of experiments using real materials through computational tests and analyses. In order to perform such an approach, it is necessary to statistically analyze a vast amount of information.

One of the major challenges in computational experimental design is the collection of structural data of material names and synthesis procedures [2]–[6]. However, it is difficult to create a recipe database for materials informatics from real tests conducted in a laboratory because the knowledge of experts has been often implicit and not well documented. The development of actual procedures for materials development requires a huge amount of time, of the order of years, and development by experts through repeated tests based on experience and intuition.

Several studies have attempted to extract materials synthesis procedures from the literature on natural language processing [7]–[13]. Mysore et al. created a corpus to extract synthesis procedures from the literature in the field of inorganic materials [12]. We previously created a corpus for a specialized field of inorganic materials and studied the extraction methods [7]. Mysore et al. also attempted to extract synthesis procedures as flow graphs using a deep learning method [13]. Despite these recent proposals of methods for extracting procedures from the scientific literature, there have been few studies analyzing the procedures for materials synthesis extracted from actual papers.
In this study, we analyzed the procedures for materials synthesis extracted from large-scale literature databases by a system that extracts the procedures as flow graphs and confirmed the characteristics of the procedures extracted by the system. We built a pipeline extraction system based on the synthesis procedure extraction method of Kuniyoshi et al. [7], and applied this system to a large set of literature for analysis to examine the applicability of the extracted procedures. We analyzed the materials synthesis procedures as a complete process, rather than as single units of operation by collecting statistics as a flow graph.

II. METHODS

We constructed a pipeline system that extracts synthesis procedures from raw data in a graph form from literature raw data based on a method of Kuniyoshi et al. [7], as shown in Figure 1. First, the pipeline extracts all paragraphs from the raw data collected by preprocessing them into a text format that can be input into subsequent models. Then, the system applies neural entity extraction and rule-based relation extraction to all paragraphs. Finally, the procedures are extracted from paragraphs based on simple rules to exclude fragmented procedures, such as those lacking a product.

A. MATERIAL SYNTHESIS PROCEDURES

Synthesis procedure using a flow graph is defined as same as the previous studies of Mysore et al. [12]. In the flow graph, edges connect each node, which is the synthesis operation and material, to represent relations between entities, including the order of operations and the input of materials into the operation.

The flow graph is defined in the materials science procedural text corpus to structure the flow of a procedure [12]. In this corpus, the synthesis procedure is annotated for the inorganic materials literature, and mentions of entities and relations between them are directly annotated in the document. Moreover, the data consisted of 200 training and 15 development and evaluation datasets.

In addition, in the aforementioned corpus, 19 entity labels and 16 relation labels were defined. An entity is defined as an element in a procedure, which can be categorized into several types such as an operation or material among others. Conversely, a relation is defined as a relationship between entities that describes the order of operations or the input of a material to an operation. The statistics and descriptions for each of these labels are shown in Table 1 and 2.

B. SYNTHESIS PROCEDURE EXTRACTION PIPELINE

We built a pipeline that enables the extraction of synthesis procedures from raw data. It can be divided into four parts. First, all paragraphs from the raw literature data are extracted for preprocessing. Second, entities from the paragraphs are extracted using an entity extractor. A bidirectional long short-term memory [14] and conditional random field [15]
K. Makino et al.: Extracting and Analyzing Inorganic Material Synthesis Procedures in Literature

TABLE 1. Definitions and statistics of entity labels from the materials science procedural text corpus [12].

| Entity label               | Train | Dev  | Test  | Description                                |
|----------------------------|-------|------|-------|--------------------------------------------|
| OPERATION                  | 3,249 | 212  | 242   | Operations in the procedure                |
| MATERIAL                   | 4,271 | 277  | 316   | Materials in the procedure                 |
| NONRECIPE-MATERIAL         | 329   | 33   | 25    | Materials not in the procedure             |
| NUMBER                     | 2,872 | 224  | 219   | Number                                     |
| PROPERTY-MISC             | 481   | 25   | 16    | Material property                          |
| PROPERTY-TYPE              | 124   | 10   | 4     | Material property type (e.g., name of numeric condition) |
| PROPERTY-UNIT             | 92    | 7    | 8     | Units of numerical materials property      |
| AMOUNT-UNIT                | 1,193 | 96   | 98    | Units of the amount of materials           |
| AMOUNT-MISC                | 149   | 14   | 7     | Other expressions for the amount of materials (e.g., few) |
| CONDITION-MISC            | 468   | 32   | 20    | Operating conditions                       |
| CONDITION-UNIT            | 1,363 | 101  | 87    | Units of operating conditions              |
| CONDITION-TYPE            | 119   | 2    | 1     | Types of operating conditions              |
| SYNTHESIS-APPARATUS       | 433   | 20   | 34    | Apparatuses in the procedure               |
| CHARACTERIZATION-APPARATUS| 54    | 2    | 11    | Apparatuses for the evaluation             |
| APPARATUS-UNIT            | 89    | 6    | 16    | Units of apparatus properties              |
| APPARATUS-PROPERTY-TYPE   | 26    | 0    | 6     | Types of apparatus properties              |
| MATERIAL-DESCRIPTION      | 1,214 | 67   | 89    | Descriptor of materials                    |
| APPARATUS-DESCRIPTION     | 165   | 10   | 9     | Descriptor of apparatus                    |
| BRAND                      | 291   | 30   | 27    | Brand of materials or apparatuses          |
| META                       | 128   | 12   | 13    | Name of synthesis method                   |
| REFERENCE                  | 106   | 10   | 11    | A citation ID                              |

Overall: 17,216 1,190 1,259

TABLE 2. Relations in the materials science procedural text corpus.

| Relation class              | Train | Dev  | Test  | Description                                                      |
|-----------------------------|-------|------|-------|------------------------------------------------------------------|
| NEXT_OPERATION              | 2,398 | 184  | 202   | Next operation from before operation                             |
| RECIPE_PRECURSOR            | 876   | 67   | 89    | Input material of synthesis procedure                            |
| RECIPE_TARGET               | 363   | 31   | 22    | Target material of synthesis procedure                           |
| PARTICIPANT_MATERIAL        | 1,723 | 113  | 124   | Intermediate generated material                                  |
| SOLVENT_MATERIAL            | 463   | 28   | 33    | Solvent material                                                 |
| ATMOSPHERIC_MATERIAL        | 183   | 11   | 14    | Atmosphere material                                              |
| NUMBER_OF                   | 2,805 | 219  | 209   | Relation between number and its unit                             |
| PROPERTY_OF                 | 586   | 35   | 21    | Property of material                                             |
| AMOUNT_OF                   | 1,512 | 130  | 121   | Material quantity properties                                    |
| CONDITION_OF                | 1,810 | 132  | 107   | Condition for operation                                         |
| APPARATUS_OF                | 455   | 20   | 36    | Apparatus condition for operation                                |
| APPARATUS_ATTR_OF           | 90    | 6    | 11    | Numerical requirements for the apparatus                        |
| DESCRIPTOR_OF               | 1,495 | 91   | 102   | Description of the subject                                      |
| BRAND_OF                    | 423   | 42   | 41    | Brand of the subject                                            |
| TYPE_OF                     | 164   | 7    | 13    | Condition of numerical conditions                                |
| CORREL_OF                   | 267   | 12   | 14    | Coreference                                                      |

Overall: 16,113 1,128 1,159

(Bi-LSTM-CRF) model [16] is used to predict entities using the representation of the pretraining model Mat-ELMo [17], which was trained on the literature in the field of inorganic material science. Third, the synthesis procedure is extracted based on the entities using a rule-based relation extractor. The entity and relation extractors are based on the method of Kuniyoshi et al. [7], which effectively extracted synthesis procedures from the literature. Finally, the paragraphs are classified depending on whether they contain the synthesis procedure or not based on the simple rule using the extracted entities and relations. Only the paragraphs that contain the synthesis procedure are selected.

1) PREPROCESSING
In the preprocessing stage of the data, all paragraphs in the literature were extracted. In this study, although several deep learning methods have been proposed to select paragraphs that mention material synthesis procedure [12], [13], [18], we decided to use all paragraphs to extract procedures because paragraph classification requires additional paragraph-level annotation and model development.

2) ENTITY EXTRACTOR
Entities were extracted by formulating the sequence labeling task with IOB (inside—outside—beginning) tags using a Bi-LSTM-CRF model [16] with Mat-ELMo [17] for the token embeddings. Mat-ELMo is an ELMo (Embeddings from Language Models) [19] pretrained with materials literature. We adopted this model although there are several other methods for entity extraction [20], because the survey confirmed that this method is effective for extracting synthesis procedures [7].
The input text was tokenized and embedded into a dense vector representation for each token using Mat-ELMo, as follows.

\[ e = \text{Mat} - \text{ELMo}(t), \]  

(1)

where \( t = [t_1, t_2, \ldots, t_L] \) is a list of tokens with length \( L \) and \( e = [e_1, e_2, \ldots, e_L] \) is a list of embeddings for each token. The probabilities of classes for tokens are obtained from LSTM.

\[ p = \text{LSTM}(e), \]  

(2)

where \( p = [p_1, p_2, \ldots, p_L] \) is a list of probabilities of classes for the tokens. The CRF is applied to \( p \) to determine the class from the probability.

\[ c = \text{CRF}(p), \]  

(3)

where \( c = [c_1, c_2, \ldots, c_L] \) is a list of classes for the tokens in the input text. The log-likelihood of the prediction of sequences was maximized to train this model.

The Target-Material entities were automatically induced by considering the Material entities connected by the Recipe_Target relation as the Target-Material entities. This is necessary because it is difficult to develop a simple rule to distinguish the Recipe_Target relation from other relations in Section II-B3, and neural methods are suitable for distinguishing the relation. By introducing the Target-Material entities, we can extract the Recipe_Target relation using a rule as in Section II-B3.

3) RULE-BASED RELATION EXTRACTOR

The relation extractor is a rule-based model used to extract the relation between an entity pair. We adapted the rules of Kuniyoshi et al. [7] to the materials science procedural text corpus, which depends on the entity labels of an entity pair, the distance between an entity pair, and the order of occurrence of the entities. According to the combination of labels of the entities, the rules were divided into three types: Operation–Operation, Operation–Material, and other relations. In the following description of the rules, the starting point of a relation is called the head, the ending point is called the tail, and an edge is denoted as Head–Tail.

a: OPERATION–OPERATION

The relation Operation–Operation takes only a Next_Operation label, which indicates the progress of the operation, e.g., if Operations appear in the order of A, B, C, and D, Next_Operation relations will be \( A \rightarrow B, B \rightarrow C, \) and \( C \rightarrow D. \)

Next_Operation: We assumed that Operation is described in the order of the operation progression. Therefore, Operation entities are connected in the order in which they appear.

b: OPERATION–MATERIAL

The Operation–Material relations mean Operation is performed using Material. The relation classes are divided into five categories depending on the function of the Material: (1) Recipe_Precursor indicates the input of a material, (2) Recipe_Target indicates the generation of a product, (3) Participant_Material indicates the generation of an intermediate product, (4) Solvent_Material indicates the solvent material of an operation, and (5) Atmospheric_Material indicates the atmosphere of an operation.

We classified Solvent_Material, Atmospheric_Material, and Participant_Material labels based on dictionary matches because the words in Material are distinctive. A dictionary was prepared for each label. The relations connect a Material to the nearest Operation in the sentence if Material matches in the dictionary because these relations take specific Material entities. The dictionaries are listed in Tables 3, 4, and 5.

| TABLE 3. Dictionary for solvent_material. |
|------------------------------------------|
| Regular expression                        |
| .water.*                                 |
| .(lalcoholglycol).ol.*                   |
| .NaOH.*                                  |
| .HCl.*                                   |
| .acetone.*                               |
| .acid.*                                  |
| .H2O.*                                   |
| .chloroform.*                            |
| .sodium hydroxide.*                      |
| .DMP.*                                   |
| .THF.*                                   |
| .N,N dimethylformamide.*                 |
| .hexane.*                                |
| .toluene.*                               |
| .H2SO4.*                                 |
| .EtOH.*                                  |

| TABLE 4. Dictionary for atmospheric_material. |
|-----------------------------------------------|
| Regular expression                           |
| .sur.*                                      |
| .argon.*                                    |
| .Ar.*                                       |
| .N2.*                                       |
| .nitrogen.*                                 |
| .H2.*                                       |
| .oxygen.*                                   |
| .O2.*                                       |
| .hydrogen.*                                 |
| .CH4.*                                      |
| .H2S.*                                      |
| .He.*                                       |

For a Recipe_Target label, as Material is the target to be extracted as Target-Material by the entity extractor in the previous section, the relation extractor connects the Target-Material to the nearest Operation.

Recipe_Precursor connects all Material except Target-Material that do not match the dictionary of Solvent_Material, Atmospheric_Material, and Participant_Material to the nearest Operation.

c: REMAINING RELATIONS

The remaining nine relation labels are defined between the other pairs of entity labels: Property_Of, which indicates the condition of a material; Condition_Of, which indicates the condition of an operation; Number_Of, which
TABLE 5. Dictionary for participant_material.

| Regular expression | Description |
|---------------------|-------------|
| *solution.*         | precipitates.* |
| *mixture.*          | carbon.*     |
| *product.*          | silica.*     |
| *samples.*          | HCl.*        |
| *precipitate.*      | NaBH₄.*      |
| *powder.*           | slurry.*     |
| *suspension.*       | Cu.*         |
| *precursor.*        | H₂O.*        |
| *sample.*           | Samples.*    |
| *products.*         | ZnO.*        |
| *chemicals.*        | KOH.*        |
| *powders.*          | compound.*   |
|                     | GO.*         |
| *solid.*            | NaOH.*       |
| *pellets.*          | films.*      |
| *solvent.*          | graphene.*   |
| *materials.*        | polymer.*    |
| *particles.*        | CTAB.*       |
| *material.*         | zeolite.*    |
| *gel.*              | SnO₂.*       |
| *reagents.*         | membrane.*   |
| *precursors.*       | hydrochloric acid.* |

Coref_Of: The relation is not detected by the rules because it is difficult to describe by rules. Although coreference needs to be taken into account to extract more accurate recipes, we leave it for future work to tackle the extraction of this relation, for example, combining extractors with high performance such as deep learning.

4) SELECTING PROCEDURES

Although only a few paragraphs actually contain procedures, our method attempts to extract procedures from all paragraphs. Therefore, we select the extracted flow graphs that actually contain procedures based on the extracted entities and relations.

We applied a simple approach to select the procedures to avoid additional annotation by developing a paragraph classifier that recognizes whether a procedure is present. As the target of this research is to extract procedural sequences, the extracted procedures require the inclusion of sequences wherein target materials are generated from other materials. Therefore, we selected the flow graphs that contain Recipe_Targets that are synthesized from Recipe_Precursors with some Operations.

III. EVALUATION

In our experiments, we first evaluated our pipeline to determine whether it could be used in practice using a benchmark dataset. In the evaluation of the pipeline, we separately evaluated entity extraction, relation extraction, and the pipeline of entity and relation extraction.

A. EVALUATION SETTINGS FOR ENTITY AND RELATION EXTRACTORS

We used the publicly available materials science procedural text corpus [12] to train the entity extractor and develop rules of the relation extractor. The corpus consists of 200 documents for training, 15 for development, and 15 for testing. The evaluation of the entity and relation extractors was performed using the testing portion of the corpus in which precision,

indicates the relationship between a number and a unit; Amount_Of, which indicates a condition of a quantity; Type_Of, which indicates the type of the numerical condition; Brand_Of, which indicates the brand of a material or equipment; Apparatus_Of, which indicates the equipment used in an operation; Apparatus_Attr_Of, which indicates the numerical condition of the equipment; and Descriptor_Of, which indicates other conditions. For these labels, the rules are defined based only on the labels of the head and tail entities and the distance between them. We explain the detailed rules in the remainder of this section.

Property_Of: This relation takes Property-Unit or Property-Misc as the head and Material, including Target-Material or Nonrecipe-Material as the tail. When Property-Unit is the head, it is connected to the nearest Material in the sentence. When Property-Misc is the head, it is connected to the nearest Material or Nonrecipe-Material in the sentence.

Condition_Of: This relation takes Condition-Unit or Condition-Misc as the head and Operation as the tail. Condition-Unit and Condition-Misc are connected to the nearest Operation in the sentence using this relation.

Number_Of: Number is connected to the nearest Property-Unit, Condition-Unit, or Apparatus-Unit that appear after the Number in the sentence. We assume that, in most cases, the relation between num and its unit matches a pattern of “(Number) (Unit)” such as “100 mL”.

Amount_Of: This relation connects Amount-Unit and Amount-Unit to the nearest Material or Nonrecipe-Material in the sentence.

Descriptor_Of: When Material-Descriptor is the head, it is connected to the nearest Material or Nonrecipe-Material in the sentence. When Apparatus-Descriptor is the head, Synthesis-Apparatus or Characterization-Apparatus can be a tail, but only the nearest Synthesis-Apparatus in the sentence is connected because Characterization-Apparatus is an apparatus for measuring characteristics and detailed descriptions are rarely provided.

Apparatus_Of: This relation connects Synthesis-Apparatus and Characterization-Apparatus with the nearest Operation with the priority given to the Operation that appears before the Apparatus in the sentence.

Type_Of: Property-Type and Apparatus-Property-Type are connected to the nearest Property-Unit and Apparatus-Unit in the sentence with this relation, respectively. When Condition-Type is the head, it is connected to the nearest Condition-Unit that appears before the Condition-Type in the sentence.

Brand_Of: The relation connects Brand to the nearest entities that may have brands (i.e., Material, Nonrecipe-Material, Synthesis-Apparatus, and Characterization-Apparatus) in the sentence.

Apparatus_Attr_Of: Apparatus-Unit is connected to the nearest Synthesis-Apparatus or Characterization-Apparatus.

Coref_Of: The relation is not detected by the rules because it is difficult to describe by rules. Although coreference needs to be taken into account to extract more accurate recipes, we leave it for future work to tackle the extraction of this relation, for example, combining extractors with high performance such as deep learning.
recall, micro-F (overall F-score), and macro-F (averaged F-score of individual classes), were used as the evaluation metrics.

Flair [21], a library of machine learning for natural language processing, was used to implement the entity extractor. Meanwhile, the pretrained model Mat-ELMo [17] was not updated during training because catastrophic forgetting [22] happened in the preliminary experiment due to the small size of the corpus. Moreover, we used a stochastic gradient descent method for training. The learning rate was halved from the initial value of 0.1 when there was no improvement in performance over three epochs, and the method was stopped when the learning rate became smaller than 0.0001. A model that showed the best performance for the development data was used.

To verify the effectiveness of the setting for the corpus, we compared our entity extractor with BERT (Bidirectional Encoder Representations from Transformers)-base [23] and ELECTRA (Efficiently Learning an Encoder that Classifies Token Replacements Accurately)-base [24], which have shown excellent performance in various natural language processing tasks. BERT is a transformer-based model that was pretrained by masked language modeling in a large number of studies [25]. Conversely, although ELECTRA is similar to BERT, it has shown higher performance due to an improved training method. To evaluate these transformer-based models, the embeddings of our entity extractors were replaced with these models, and the training was performed in the same way.

We evaluated the performance of the relation extractor separately from the entity extractor by assigning gold entities to verify the validity of the developed rules. This is the same setting as the rules that were developed. We also evaluated the performance of both extractors in the pipeline to determine the performance of the extraction system, i.e., the output of the entity extractor is fed into the relation extractor.

**B. EVALUATION OF THE ENTITY EXTRACTOR**

The evaluation results of the entity extractors, including transformer-based models, are shown in Table 6. The entity extractor successfully extracted entities with the highest performance, showing that it is important to use the model learned from the literature in the same domain (i.e., the materials science field) as the target corpus. From these results, we confirmed that Mat-ELMo is a suitable pretrained model to extract entities from the literature in materials science, which aligns with the finding by Kuniyoshi et al. [7]. As the transformer-based models use subword tokenization obtained from general text such as those found in Wikipedia, they might have difficulties in obtaining representations specific to the materials field.

As detailed results, evaluation on each class and a confusion matrix are shown in Table 7 and Figure 2, respectively. Overall, classes with few training data (cf. Table 1), classes with many variations to mention, and classes that are ambiguous with other classes tended to have lower performance. Entities that are the backbone of procedures (i.e., Operation and Material entities) were extracted with high performance, while Target-Material entities were extracted with low performance. This is because Target-Material is often confused with Material and it is written in a cumbersome notation; in other words, such entity may contain symbols such as,

| Class                          | Precision | Recall | Macro-F | Micro-F |
|-------------------------------|-----------|--------|---------|---------|
| Operation                     | 0.794     | 0.938  | 0.860   | 0.800   |
| Material                      | 0.768     | 0.881  | 0.820   | 0.804   |
| Target-Material               | 0.458     | 0.524  | 0.489   | 0.452   |
| NonRecipe-Material            | 0.450     | 0.429  | 0.439   | 0.452   |
| Number                        | 0.949     | 0.936  | 0.942   | 0.946   |
| Property-Misc                 | 0.300     | 0.400  | 0.343   | 0.367   |
| Property-Type                 | 0.250     | 0.250  | 0.250   | 0.250   |
| Property-Unit                 | 0.500     | 0.500  | 0.500   | 0.500   |
| Amount-Misc                   | 0.273     | 0.429  | 0.333   | 0.333   |
| Amount-Unit                   | 0.969     | 0.949  | 0.959   | 0.959   |
| Condition-Misc                | 0.588     | 1.000  | 0.741   | 0.741   |
| Condition-Type                | 0.667     | 1.000  | 0.125   | 0.125   |
| Condition-Unit                | 0.914     | 0.977  | 0.944   | 0.944   |
| Synthesis-Apparatus           | 0.677     | 0.677  | 0.677   | 0.677   |
| Characterization-Apparatus    | 0.444     | 0.364  | 0.400   | 0.400   |
| Apparatus-Unit                | 0.667     | 0.308  | 0.421   | 0.421   |
| Apparatus-Property-Type       | 0.000     | 0.000  | 0.000   | 0.000   |
| Material-Descriptor           | 0.663     | 0.708  | 0.685   | 0.685   |
| Apparatus-Descriptor          | 0.533     | 0.889  | 0.667   | 0.667   |
| Brand                         | 0.657     | 0.852  | 0.742   | 0.742   |
| Concept                       | 0.429     | 0.462  | 0.444   | 0.444   |
| Reference                     | 0.571     | 0.727  | 0.640   | 0.640   |

**TABLE 7.** Detailed results of entity extraction.
TABLE 8. Performance of the relation extractor when the gold entities were given and extracted by our extractor.

| Class                      | Entity given | Pipeline |
|----------------------------|--------------|----------|
|                            | P            | R        | P            | R            | F            | F            |
| NEXT_OPERATION             | 0.875        | 0.990    | 0.929        | 0.544        | 0.743        | 0.628        |
| RECIPE_PRECURSOR           | 0.474        | 0.733    | 0.579        | 0.360        | 0.628        | 0.458        |
| RECIPE_TARGET              | 0.952        | 0.952    | 0.952        | 0.360        | 0.571        | 0.533        |
| PARTICIPANT_MATERIAL       | 0.794        | 0.435    | 0.562        | 0.551        | 0.574        | 0.446        |
| SOLVENT_MATERIAL           | 0.438        | 0.656    | 0.525        | 0.340        | 0.531        | 0.415        |
| ATMOSPHERIC_MATERIAL       | 0.375        | 1.000    | 0.545        | 0.270        | 0.833        | 0.408        |
| NUMBER_OF                  | 0.962        | 0.946    | 0.954        | 0.888        | 0.854        | 0.871        |
| PROPERTY_OF                | 1.000        | 0.950    | 0.974        | 0.250        | 0.300        | 0.273        |
| CONDITION_OF               | 0.990        | 0.980    | 0.985        | 0.750        | 0.891        | 0.814        |
| AMOUNT_OF                  | 0.937        | 0.781    | 0.852        | 0.722        | 0.614        | 0.664        |
| APPARATUS_OF               | 0.846        | 0.971    | 0.904        | 0.526        | 0.588        | 0.556        |
| APPARATUS_ATTR_OF          | 0.818        | 0.900    | 0.857        | 0.400        | 0.200        | 0.267        |
| DESCRIPTOR_OF              | 0.975        | 0.929    | 0.952        | 0.560        | 0.600        | 0.580        |
| BRAND_OF                   | 0.917        | 0.564    | 0.698        | 0.364        | 0.308        | 0.333        |
| TYPE_OF                    | 1.000        | 0.875    | 0.933        | 0.077        | 0.125        | 0.095        |
| COREF_OF                   | 0.000        | 0.000    | 0.000        | 0.000        | 0.000        | 0.000        |
| Macro                      | 0.829        | 0.832    | 0.830        | 0.576        | 0.645        | 0.609        |
| Micro                      | 0.791        | 0.772    | 0.762        | 0.510        | 0.444        | 0.459        |

As for entities that are supplementary information for procedures, the numbers were extracted in high performance since they have few variations in the numbers. Conversely, _Misc and Type have variations, ns were extracted with errors. Moreover, Apparatus-Property-Type was not extracted at all. This is because the mentions of Apparatus-Property-Type are ambiguous from our observation and there are only a few annotations and no examples in the development set; thus, we were not able to check the performance beforehand. Note that Apparatus-Property-Type is not likely to have a significant impact on the synthesis procedure and it is not critical in our analysis of the brief overview of the procedures in the literature.

C. EVALUATION OF THE RELATION EXTRACTOR

The results for the rule-based relation extractor in the pipeline setting with gold entities are shown in Table 8. Our extractor with gold entities obtained a micro-F of more than 0.8 without using any sophisticated deep-learning methods. The descriptions of the procedures are written in certain patterns (e.g., step-by-step explanation), and the simple rules were able to capture these patterns. On the other hand, the extraction performance of the pipeline had a micro-F of 0.609, which means that even if the input entities include entity extraction errors, the prediction of the relations is not really affected, even for the rule-based extractor.

For the detailed evaluation, the performance of each relation class for both the given entity and pipeline settings is also shown in Table 8 and the confusion matrix for the case where the gold entity is given is shown in Figure 3.

For the results with gold entities, our extractor, which is based only on the class of the entity and the pattern of occurrence, can perform extractions with high performance for classes that have few variations, for which we could easily create rules. For instance, Next_Operation and Recipe_Target are classes that describe the general flow of the synthesis procedure. The rules are created based on the observation of instances in the training data, where most procedures are written down in a similar way. For Next_Operation, we constructed a rule that connects the Operations in order, assuming that all Operations are described in processing orders. Under this rule, the recall of Next_Operation was 0.990, which indicates that our observation is common among the procedures. In addition, the performance of Condition_Of and Property_Of were high, because they have only a limited number of linguistic patterns. In contrast, the performance for classes such as Recipe_Precursor, Participant_Material, Solvent_Material, and Atmospheric_Material, which were extracted by dictionary-based rules, was low because our extractor considers only text without context and cannot handle a variety of occurrence patterns. Incidentally, Coref_Of was not extracted at all because there was no established rule for this relation.
For the pipeline result, the Next_Operation class, which is a key relationship for the sequence of operations in the procedure, had an F-score above 0.6. This indicates that the main flow of the procedure can be extracted to some extent, even if the extraction error has propagated from the previous entity extractor. In contrast, classes such as Property_Of, Brand_Of, and Apparatus_Attr_Of had extremely low performance due to errors in entity extraction. However, since such relation classes have little effect on the analysis in Section IV, these low performances are not considered significant.

IV. LARGE-SCALE EXTRACTION

In this section, we discuss the application of the extraction system to a large set of literature that was collected and analyzed to extract a synthesis procedure graph. In Section IV-A, we discuss large-scale extraction. We analyzed the procedures extracted from a large body of literature and investigated their characteristics in subsequent sections. In this investigation, we aimed to verify the rationality of the extracted procedures and determine whether they can be used to obtain useful information for materials development or other tasks. We performed four analyses: exploring frequently occurring subgraphs to verify typical procedures (Section IV-B); checking the frequency of the elements in Target-Material to obtain the trends in the elements and materials used (Section IV-C); calculating the term frequency-inverse document frequency (TF-IDF), an index based on co-occurrence, to check the relationship between Target-Material and Operation (Section IV-D); and analyzing an extracted procedure as a case study (Section IV-E).

A. APPLICATION OF EXTRACTOR

Our system was applied to extract synthesis procedures from a large set of documents that were collected from the Journal of Materials Chemistry A (JMCA), which is published by The Royal Society of Chemistry and focuses on areas such as batteries, fuel cells, sustainable materials, photovoltaics, supercapacitors, and water splitting. These articles were used as the literature source for large-scale extraction. We purchased all articles from 2015 to 2019 in XML format. The total number of articles was 14,310.

To extract synthesis procedures from large-scale literature, we performed the following process as described in Section II. First, as a preprocessing step, we extracted all paragraphs from the articles in XML format. Then, the entity and relation extractors developed on the material synthesis procedural corpus were directly applied to extract entities and relations. Finally, we selected flow graphs representing the procedures from all paragraphs. Although this method was developed for the material science procedural corpus, it can be used seamlessly because the domain of the corpus matches the domain of JMCA in terms of materials, and they are similarly described in our observations.

From 14,310 studies, we obtained 347,480 paragraphs in total and extracted 89,578 procedures. Therefore, approximately one-third of the paragraphs contained procedures. Surprisingly, each literature mentioned six procedures on average, even considering miscounts due to prediction errors. The statistics of the extracted entities and relations are shown in Table 9 and 10, respectively.

B. FREQUENT SUBGRAPHS

We extracted subgraphs that frequently appear in the data set using gSpan [26], which is an algorithm for mining frequent subgraphs, and then checked whether they were reasonable. If the frequently occurring subgraphs are reasonable from the human point of view, the flow graphs of the extracted procedures can be said to contain typical procedures.

The top 10 frequent subgraphs that contained Operation entities are shown in Figure 4 for each number of nodes in the subgraphs. The frequent subgraphs in the figure shows an example of a general procedure; for example, the first and sixth most-frequent 4-node subgraphs indicate that the compounds are heated and dried at 60 °C to remove ethanol, while...
the fourth most-frequent in the 5-node graph shows that 60 °C is appropriate for drying ethanol. From this observation, the graphs in the figure clearly show the general processes and we can conclude that the extracted procedures include typical processes.

C. ANALYSIS OF ELEMENTS IN TARGET-MATERIAL

We analyzed Target-Material to test the feasibility of the extracted procedures. In our results, every procedure has Target-Material because those without it were filtered out. It was difficult to directly analyze them because Target-Material is unique in most literature. Therefore, we analyzed the elements included in Target-Material.

As a basic step, the included elements were counted and the element names were extracted from left to right, ignoring valence and other numerical values. We filtered out the Target-Material that was determined to be composed of one element or none to avoid the effect of erroneous extraction.

The frequency of the elements is summarized in Figure 5. The elements were normalized using the number of Target-Material entities and scaled logarithmically. Elements with a single letter may be overestimated owing to errors in parsing elements contained in Target-Material. For example, tin (IV) oxide (SnO2) nanoparticles mentioned as “SnO2NP” are recognized as a combination of SnO2, nitrogen (N), and phosphorus (P), even though in this case, NP represents nanoparticles.

The results show that elements O, C, N, P, and S, which are commonly contained in various compounds, and Ni and Ti, which are conventionally used in electrodes, appear...
frequently. Moreover, it was confirmed that the less reactive noble gases (elements in the rightmost column of Figure 5) and high-period elements (third row from the bottom) that are difficult to obtain are not widely used in Target-Material. Considering that the literature set used for extraction included the topic of solar cell electrodes, group 6 elements W and Mo were often identified as they are often used for this application. Therefore, the extraction results are in accordance with the domain of the data set. The extraction performance for Target-Material was not high in Table 7, but the general trend is considered reasonable.

**D. CORRELATION BETWEEN TARGET-MATERIAL AND OPERATION**

We calculated the TF-IDF values of Operations for each element in Target-Material to analyze the correlations between the Operations and elements. TF-IDF is calculated based on the frequency that a term appears and uniqueness of the term to the document. Since the purpose of this analysis was to discover a unique Operation for each Target-Material and to find the Operations that may be required to construct that material, TF-IDF is a suitable metric for this analysis. To identify unique Operations, we ranked the TF-IDF scores of all elements in Target-Material, where the top-ranked Operations were considered as important for constructing Target-Material.

Here, the Operations in a procedure are considered as terms in a document, and each element in the Target-Material of a procedure is considered as a document. A document $D_e$ corresponding to an element $e$ is defined in which a set of all procedures is $P$, a set of lemmatized Operations in a procedure $p$ is $O_p$, and a set of elements in Target-Material of a procedure $p$ is $E_p$.

$$D_e = \bigcup_{p \in P} \mathbb{U}[e \in E_p] \cap O_p$$  

$\mathbb{U}[-]$ is a function that returns a universal set $U$ if the condition in the bracket is satisfied, and an empty set $\emptyset$ otherwise; therefore, $D_e$ includes an aggregation of the Operations in the procedure to synthesize Target-Material containing element $e$.

The TF-IDF score $\text{TFIDF}(d, t)$ is defined as a function of a term $t$ in a document $d$, where $\text{count}(d, t)$ is a function that returns a term $t$ count in a document $d$, and $1[-]$ is a function that returns 1 if the condition in the bracket is satisfied and 0 otherwise.

$$\text{TF}(d, t) = \frac{\text{count}(d, t)}{\sum_{w \in d} \text{count}(d, w)}$$

![Figure 5](image-url)  

**FIGURE 5.** Number of individual elements included in target-material. The elements in gray did not appear in the extracted procedures.
The top 20 operations ranked by the TF-IDF score for each element in target-material.

| Rank | O   | C   | Co  | Ni  | Ti  |
|------|-----|-----|-----|-----|-----|
| 1    | obtain | prepare | prepare | obtain | obtain |
| 2    | form | form | synthesize | form | form |
| 3    | synthesize | synthesize | form | form | form |
| 4    | use | observe | use | use | use |
| 5    | observe | use | formation | observe | fabricate |
| 6    | fabricate | increase | observe | fabricate | observe |
| 7    | formation | formation | fabricate | formation | add |
| 8    | increase | fabricate | increase | increase | dry |
| 9    | add | dry | add | dry | dry |
| 10   | dry | dry | add | deposit | deposit |
| 11   | perform | perform | maintain | maintain | formation |
| 12   | wash | show | perform | grow† | anneal* |
| 13   | dissolve | decrease | wash | perform | wash |
| 14   | show | measure | grow† | calculate | achieve |
| 15   | achieve | calculate | dissolve | heat | heat |
| 16   | measure | achieve | calculate | wash | measure |
| 17   | decrease | maintain | heat | achieve | stir |
| 18   | heat | wash | achieve | heat | perform |
| 19   | deposit | remove | show | anneal* | dissolve |
| 20   | maintain | dissolve | anneal* | show | mix |

The top 20 Operations ranked by the TF-IDF score are shown in Table 11 for frequent typical elements (O and C) and metallic elements (Co, Ni, and Ti) shown in Figure 5. We lemmatized Operation using scispacy [27] before calculating the TF-IDF score. We observed the general Operation in the top 10 Operation. However, the 11th to 20th Operation are characteristic of each element. We used the Operation that occurs with the O element, which is one of the most common elements in the analysis. In addition, “grow†”, which belongs to the columns of Co and Ni, is a characteristic Operation as they are used when sheets or crystals are created. In addition, “anneal*”, which is seen for metallic elements, is a unique Operation for metals. These factors observed from Operation are consistent with the general knowledge in the field.

E. CASE STUDY OF EXTRACTED PROCEDURES

An example of an extraction procedure is shown at the bottom of Figure 1. In this study, we sampled 10 documents and checked 64 procedures that were included in the sampled documents to test the feasibility of the extracted procedures. The obtained procedures that were manually chosen from the sampled procedures provided information and conditions about each operation, as well as how the Target-Material was produced from the input material. However, several errors were identified, such as “0.075 mmol” in the middle of the paragraph that was not extracted as an entity and “water” and “ethanol” that were the Solvent_Material of “washed” were incorrectly connected to “placed” by the relation extractor. However, the general framework of the procedure was extracted, and the analysis of a large number of extracted procedures is expected to contribute to the development of materials.

V. RELATED WORK

A. NAMED ENTITY RECOGNITION

Named entity recognition, which is a task to extract terms from text, has been widely studied and various methods have been proposed [28], [29] including rule-based method [30], [31], feature-based machine learning method [32]–[34], and deep-learning-based methods [16], [35]–[38]. For rule-based methods, domain-specific rules are developed based on, e.g., syntactic-lexical patterns [30] and dictionary [31]. Then, feature-based machine learning methods are proposed to obtain flexibility. Features are constructed from, for example, word frequencies [32], syntactic features such as part-of-speech tag, and their combination [33], [34].

In contrast to these methods, deep-learning-based methods, which have been making progress in recent years, can extract named entities with even higher accuracy and flexibility. There are various entity extraction methods such as span-based models [36], transition-based models [39], [40], and sequence-to-sequence methods [38], but the most basic method is the sequence-labeling-based ones. The Flair library [21] used here also applies sequence labeling in the BiLSTM-CRF model [16], which uses a representation of the pretrained models [19], [23].

B. RELATION EXTRACTION

After the named entity extraction, the relation extraction is performed to extract the relations between entities. While most of the existing research on relation extraction has focused only on relations within a single sentence, recent studies have been extended to document-level relation extraction, which extracts relations across sentences. In general, document-level relations are more complex than sentence-level ones, which have been studied using rule-based [41] and shallow machine-learning-based methods [42], [43]. Deep-learning-based methods [44]–[46] are often used to extract document-level relations. For example, relation classification is performed based on context-embedded representations [44] and representing the document as a graph [45], [46]. However, for the extraction target of this study, we used a rule-based extractor because our observations confirmed a clear pattern of appearance of the relations.

C. INFORMATION EXTRACTION FOR MATERIALS SCIENCE

Several studies have attempted to extract material names with their properties [47], [48] and material synthesis procedures [7]–[11], [13] by natural language processing. Swain et al. extracted chemical entities and their properties from scientific documents [47]. Mysore et al. tried to obtain an inorganic material synthesis procedure to solve the situation where research on inorganic materials has not progressed as much as research on organic materials [13]. However, large-scale extraction and analysis of the extracted synthesis
procedures have not been performed. Kuniyoshi et al. also performed research on extracting synthesis procedures for inorganic materials, especially for all-solid-state batteries [7].

Material synthesis procedures are defined in various ways [7], [9], [12], but the main elements to be extracted are common: target materials; ingredients of target materials; operations such as mixing and annealing; and conditions such as temperature and time. These elements are the basis of materials synthesis. Analyses of the synthesis procedures in the literature have provided information about these materials for practical applications. For example, Mahbub et al. analyzed the experimental conditions to gain insight into solid-state battery materials from a database constructed from information in the literature [49]. In addition, Kim et al. predicted precursor materials of perovskite using text embedding [17], and Segler et al. designed synthesis procedures for organic molecules by predicting precursors from target materials [50]. These studies demonstrate the utility of materials synthesis procedures in the literature.

Moreover, studies on the analysis of the extracted material information were also conducted. Saal et al. developed a machine-learning-based model to predict new compounds from database information [51]. Raccuglia et al. predicted the reaction outcomes for the crystallization of templated vanadium selenites from experimental notebooks [52]. However, analysis of the synthesis procedures has not progressed significantly, and in particular, there is no research that has analyzed the procedure as a flow graph. In this context, the current study makes a significant contribution to the field, because we analyzed the synthesis procedures as graphs extracted from large amounts of literature.

VI. CONCLUSION

In this study, we constructed a pipeline system for extracting synthesis procedures from large literature datasets and analyzed the obtained synthesis procedures. Our pipeline extractor consists of: preprocessing; an entity extractor (comprising Mat-ELMo and Bi-LSTM-CRF models) with an F-measure of 0.807; a rule-based relation extractor (F-measure of 0.830); and valid procedure selection, giving an overall performance of 0.609.

We analyzed materials synthesis procedures in a graph form to consider synthesis as a linked process, not a set of individual processes. In the analysis, we confirmed that the extracted procedures were reasonable because they contained typical processes as subgraphs, the characteristics of each element were indicated, and the correlations between target materials and operations were reasonable.

VII. FUTURE WORK

For further study, we will attempt to improve the performance of procedure extraction and provide more insights into the synthesis procedures, represented as procedures for discovering new materials. Since only a limited number of classes can be extracted with high performance, improving the extraction performance of the system is our next challenge to obtain more accurate procedures. This could be achieved by improving entity extraction by applying normalization as a preprocess, e.g., replacing materials written in chemical formulas with special tokens or a resolved coreference. In addition, the relation extractor could be modified by applying a flexible method, such as deep learning, to identify classes that are difficult to distinguish by rules. Furthermore, we will continue to analyze the procedures extracted by the system and develop materials based on the obtained knowledge.

REFERENCES

[1] J. Zhang, B. Xiao, X. Liu, P. Liu, P. Xi, W. Xiao, J. Ding, D. Gao, and D. Xue, “Copper dopants improved the hydrogen evolution reaction of earth-abundant cobalt pyrite catalysts by activating the electrocatalytically inert sulfur sites,” J. Mater. Chem. A, vol. 5, no. 33, pp. 17601–17608, 2017, doi: 10.1039/C7TA05433E.

[2] C. Draxl and M. Scheffler, “NOMAD: The FAIR concept for big data-driven materials science,” MRS Bull., vol. 43, no. 9, pp. 676–682, Sep. 2018.

[3] G. R. Schleider, A. C. M. Padilha, C. M. Acosta, M. Costa, and A. Faziozzi, “From DFT to machine learning: Recent approaches to materials science—A review,” J. Phys., Mater., vol. 2, no. 3, Jul. 2019, Art. no. 032001.

[4] K. T. Butler, D. W. Davies, H. Cartwright, O. Isayev, and A. Walsh, “Machine learning for molecular and materials science,” Nature, vol. 559, pp. 547–555, Jul. 2018.

[5] S. Curtarolo, G. L. W. Hart, M. B. Nardelli, N. Mingo, S. Sanvito, and O. Levy, “The high-throughput highway to computational materials design,” Nature Mater., vol. 12, no. 3, pp. 191–201, 2013.

[6] J. M. Rickman, T. Lookman, and S. V. Kalinin, “Materials informatics: From the atomic-level to the continuum,” Acta Mater., vol. 168, pp. 473–510, Apr. 2019.

[7] F. Kuniyoshi, K. Makino, J. Ozawa, and M. Miwa, “Annotating and extracting synthesis processes of all-solid-state batteries from scientific literature,” in Proc. 12th Lang. Resour. Eval. Conf., Marseille, France, May 2020, pp. 1941–1950. [Online]. Available: https://www.aclweb.org/anthology/2020.lrec-1.299.

[8] A. Jain, S. Ong, G. Hautier, and W. Chen, “Commentary: The materials project: A materials genome approach to accelerating materials innovation,” APL Mater., vol. 1, Jul. 2013, Art. no. 011002.

[9] O. Kononova, H. Huo, T. He, Z. Rong, T. Botari, W. Sun, V. Tshitiyan, and G. Ceder, “Text-mined dataset of inorganic materials synthesis recipes,” Sci. Data, vol. 6, no. 1, Dec. 2019, Art. no. 203.

[10] T. Lookman, P. V. Balachandran, D. Xue, and R. Yuan, “Active learning in materials science with emphasis on adaptive sampling using uncertainties for targeted design,” NPJ Comput. Mater., vol. 5, no. 1, pp. 1–17, Dec. 2019.

[11] E. Kim, K. Huang, A. Saunders, A. McCallum, G. Ceder, and E. Olivetti, “Materials synthesis insights from scientific literature via text extraction and machine learning,” Chem. Mater., vol. 29, no. 21, pp. 9436–9444, Nov. 2017.

[12] S. Mysore, Z. Jensen, E. Kim, K. Huang, H.-S. Chang, E. Strubell, J. Flanigan, A. McCallum, and E. Olivetti, “The materials science procedural text corpus: Annotating materials synthesis procedures with shallow semantic structures,” in Proc. 13th Linguistic Annotation Workshop, Florence, Italy, Aug. 2019, pp. 56–64. [Online]. Available: https://www.aclweb.org/anthology/W19-4007.

[13] S. Mysore, E. Kim, E. Strubell, A. Liu, H.-S. Chang, S. Kompella, K. Huang, A. McCallum, and E. Olivetti, “Automatically extracting action graphs from materials science synthesis procedures,” in Proc. NIPS Workshop Mach. Learn. Molecules Mater., Long Beach, CA, USA, 2017, pp. 1–11.

[14] S. Hochreiter and J. Schmidhuber, “Long short-term memory,” Neural Comput., vol. 9, no. 8, pp. 1735–1780, 1997, doi: 10.1162/neco.1997.9.8.1735.
[15] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, “Conditional random fields: Probabilistic models for labeling and segmenting sequence data,” in Proc. 11th Int. Conf. Mach. Learning, San Francisco, CA, USA, 2001, pp. 282–289.

[16] Z. Huang, W. Xu, and K. Yu, “Bidirectional LSTM-CRF models for sequence tagging,” 2015, arXiv:1508.01958v1.

[17] E. Kim, Z. Ijima, S. van Grootel, K. Huang, M. Staal, S. Myorse, H.-S. Chang, E. Strubell, A. McCallum, S. Jegelka, and E. Olivierti, “Inorganic materials synthesis planning with literature-trained neural networks,” J. Chem. Inf. Model., vol. 60, no. 3, pp. 1194–1201, Mar. 2020. doi: 10.1021/acs.jcim.9b00995.

[18] H. Huo, Z. Rong, O. Kononova, W. Sun, T. Botari, T. He, V. Tshitoyan, and G. Ceder, “Semi-supervised machine-learning classification of materials synthesis procedures,” NPJ Comput. Mater., vol. 5, no. 1, pp. 1–7, Dec. 2019.

[19] M. E. Peters, M. Neumann, M. Iyyer, and M. Gardner, “Deep contextualized word representations,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol., vol. 1, New Orleans, LA, USA, Jun. 2018, pp. 2227–2237. [Online]. Available: https://www.aclweb.org/anthology/N18-1202

[20] S. Eltyeb and N. Salim, “Chemical named entities recognition: A review on approaches and applications,” J. Cheminformatics, vol. 6, no. 1, pp. 1–12, Dec. 2014.

[21] A. Akbik, D. Blythe, and R. Vollgraf, “Contextual string embeddings for sequence labeling,” in Proc. 27th Int. Conf. Comput. Linguistics, 2018, pp. 1638–1649.

[22] D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, “Overcoming catastrophic forgetting in neural networks,” Proc. Nat. Acad. Sci. USA, vol. 114, no. 13, pp. 3521–3526, Mar. 2017. [Online]. Available: https://www.pnas.org/content/114/13/552

[23] J. Devlin, M.-W. 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., vol. 1, Jun. 2019, pp. 4171–4186. [Online]. Available: https://www.aclweb.org/anthology/N19-1423

[24] Z. Ji, S. M. Huang, Q. Li, V. Le, and C. D. Manning, “ELECTRA: Pre-training text encoders as discriminators rather than generators,” in Proc. ICLR, 2020, pp. 1–18. [Online]. Available: https://openreview.net/pdf?id=r1tMH1BtvB

[25] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, K. Clark, M.-T. Luong, Q. V. Le, and C. D. Manning, “Attention is all you need,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2017, pp. 1–12. [Online]. Available: https://www.aclweb.org/anthology/D17-1068

[26] Y. Li, K. Bontcheva, and H. Cunningham, “Svm based learning system for named entity recognition,” in Deterministic and Statistical Methods in Machine Learning, J. Winkler, M. Niranjan, and N. Lawrence, Eds. Berlin, Germany: Springer, 2005, pp. 319–339.

[27] R. Ageri and G. Rigu, “Robust multilingual named entity recognition with shallow semi-supervised features,” Artif. Intell., vol. 238, pp. 63–82, Sep. 2016.

[28] Z. Fang, Y. Cao, T. Li, R. Jia, F. Fang, Y. Shang, and Y. Lu, “TEBNER: Domain specific named entity recognition with type expanded boundary-aware network,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2021, pp. 198–207. [Online]. Available: https://aclanthology.org/2021.emnlp-main.18

[29] M. G. Sohrab and M. Miwa, “Deep exhaustive model for nested named entity recognition,” in Proc. Conf. Empirical Methods Natural Lang. Process., 2018, pp. 2843–2849. [Online]. Available: https://aclanthology.org/D18-1309

[30] J. Fu, X. Huang, and P. Liu, “SpanNER: Named entity recognition as span prediction,” in Proc. 11th Int. Conf. Joint Conf. Natural Lang. Process., Aug. 2021, pp. 7183–7195. [Online]. Available: https://aclanthology.org/2021.acl-long.558

[31] Y. Zhang, H. Shen, Y. Zhao, Q. Liu, and D. Yin, “Learning tag dependencies for sequence tagging,” in Proc. 27th Int. Joint Conf. Artif. Intell., Jul. 2018, pp. 4581–4587. doi: 10.24966/jiai2018/2018.637

[32] Y. Goldberg and I. Nivre, “A dynamic Oracle for arc-ereager dependency parsing,” in Proc. COLING, Mumbai, India, Dec. 2012, pp. 959–976. [Online]. Available: https://aclanthology.org/C12-1059

[33] Z. Ji, T. Xia, M. Han, and J. Xiao, “A neural transition-based joint model for disease named entity recognition and normalization,” in Proc. 59th Ann. Meeting Assoc. Comput. Linguistics, Aug. 2021, pp. 2819–2827. [Online]. Available: https://aclanthology.org/2021.acl-main.219

[34] C. Quan, M. Wang, and F. Ren, “An unsupervised text mining method for relation extraction from biomedical literature,” PLoS ONE, vol. 9, no. 7, Jul. 2014, Art. no. e102039, doi: 10.1371/journal.pone.0102039

[35] Z. Zhang, “Weakly-supervised relation classification for information extraction,” in Proc. 15th ACM Conf. Inf. Knowl. Manage., New York, NY, USA, 2004, pp. 581–588. doi: 10.1145/1031710.1031759

[36] Z. GuoDong, S. Jian, Z. Jie, and Z. Min, “Exploring various knowledge in relation extraction,” in Proc. 43nd Ann. Meeting Assoc. Comput. Linguistics, Jun. 2005, pp. 427–434. [Online]. Available: https://aclanthology.org/P05-1053

[37] P. Verga, E. Strubell, and A. McCallum, “Simultaneously self-attending to all mentions for full-abstract biological relation extraction,” in Proc. Conf. North Amer. Chapter Assoc. Comput. Linguistics, vol. 1, New Orleans, LO, USA, Jun. 2018, pp. 872–884. [Online]. Available: https://www.aclweb.org/anthology/N18-1080

[38] W. Zhou, K. Huang, T. Ma, and J. Huang, “Document-level relation extraction with adaptive thresholding and localized context pooling,” in Proc. AAAI Conf. Artif. Intell., 2021, pp. 1–9.

[39] C. Chiu, A. Tomala, S. Matthews, E. Strubell, A. Saunders, A. McCallum, and E. Olivierti, “Connecting the dots: Document-level neural relation extraction with edge-oriented graphs,” in Proc. Conf. Empirical Methods Natural Lang. Process., Hong Kong, 2019, pp. 4925–4936. [Online]. Available: https://aclanthology.org/D19-1498

[40] M. C. Swain and J. M. Cole, “ChemDataExtractor: A toolkit for automated extraction of chemical information from the scientific literature,” J. Chem. Inf. Model., vol. 60, no. 10, pp. 1894–1904, Oct. 2016.

[41] E. Kim, K. Huang, A. Tomala, S. Matthews, E. Strubell, A. Saunders, A. McCallum, and E. Olivierti, “Machine-learned and codified synthesis parameters of oxide materials,” Sci. Data, vol. 4, no. 1, pp. 1–9, Dec. 2017.

[42] R. Mahbub, K. Huang, Z. Jensen, Z. D. Hood, J. L. M. Rupp, and E. A. Olivierti, “Text mining for processing conditions of solid-state battery electrolytes,” Electrochem. Commun., vol. 12, Dec. 2010, Art. no. 106860. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1388248110032113

[43] M. H. S. Segler, M. Preuss, and M. P. Waller, “Planning chemical syntheses with deep neural networks and symbolic AI,” Nature, vol. 555, no. 7698, pp. 660–664, Jan. 2018. doi: 10.1038/nature26221

[44] J. E. Saal, S. Kirkin, M. Aylk, B. Meredig, and C. Wolverton, “Materials design and discovery with high-throughput density functional theory: The open quantum materials database (OQMD),” JOM, vol. 65, no. 11, pp. 1501–1509, Nov. 2013.

[45] P. Raccuglia, K. C. Elbert, P. D. Adler, C. Falk, M. B. Wenny, A. Mollo, M. Zeller, S. A. Frieder, J. Schier, and A. J. Norquist, “Machine-learning-assisted materials discovery using failed experiments,” Nature, vol. 553, pp. 73–76, May 2016.
KOHEI MAKINO received the B.E. and M.E. degrees from the Toyota Technological Institute, Aichi, Nagoya, Japan, where he is currently pursuing the Ph.D. degree. From 2019 to 2021, he was a Research Assistant with the National Institute of Advanced Industrial Science and Technology. His research interests include deep learning, natural language processing, and information extraction.

FUSATAKA KUNIYOSHI received the M.S. degree in computer science from the Nara Institute of Science and Technology, in 2017. He is currently a Researcher with the National Institute of Advanced Industrial Science and Technology and Panasonic Corporation. His research interests include natural language processing and computer vision.

JUN OZAWA received the Ph.D. degree in system science from the Tokyo Institute of Technology, Yokohama, Japan, in 1998. Since 1990, he has been a Researcher at Panasonic. He is currently the Director of Panasonic—AIST Advanced AI Research Laboratory, National Institute of Advanced Industrial Science and Technology. His research interests include machine learning and its industrial applications.

MAKOTO MIWA received the Ph.D. degree from The University of Tokyo, in 2008. He is currently an Associate Professor with the Toyota Technological Institute and a Visiting Researcher with the National Institute of Advanced Industrial Science and Technology. His research interests include natural language processing, deep learning, and information extraction.

* * *