

# Automated Chemical Reaction Extraction from Scientific Literature

Jiang Guo,<sup>§</sup> A. Santiago Ibanez-Lopez,<sup>§</sup> Hanyu Gao, Victor Quach, Connor W. Coley, Klavs F. Jensen, and Regina Barzilay\*

 Cite This: <https://doi.org/10.1021/acs.jcim.1c00284>

 Read Online

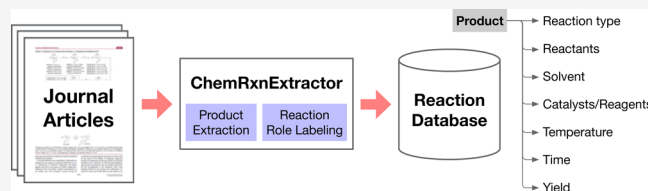
ACCESS |

 Metrics & More

 Article Recommendations

 Supporting Information

**ABSTRACT:** Access to structured chemical reaction data is of key importance for chemists in performing bench experiments and in modern applications like computer-aided drug design. Existing reaction databases are generally populated by human curators through manual abstraction from published literature (e.g., patents and journals), which is time consuming and labor intensive, especially with the exponential growth of chemical literature in recent years. In this study, we focus on developing automated methods for extracting reactions from chemical literature. We consider journal publications as the target source of information, which are more comprehensive and better represent the latest developments in chemistry compared to patents; however, they are less formulaic in their descriptions of reactions. To implement the reaction extraction system, we first devised a chemical reaction schema, primarily including a central *product*, and a set of associated reaction roles such as *reactants*, *catalyst*, *solvent*, and so on. We formulate the task as a structure prediction problem and solve it with a two-stage deep learning framework consisting of *product extraction* and *reaction role labeling*. Both models are built upon Transformer-based encoders, which are adaptively pretrained using domain and task-relevant unlabeled data. Our models are shown to be both effective and data efficient, achieving an F1 score of 76.2% in product extraction and 78.7% in role extraction, with only hundreds of annotated reactions.



## INTRODUCTION

Scientific literature (e.g., journal articles and patents) has long been a critical information source to synthetic chemists for finding ways to perform particular chemical reactions or synthetic procedures of interest. To reduce the time and costs entailed by information retrieval, as well as to facilitate access to reaction data, commercial efforts have been invested in constructing structured databases from unstructured literature, such as Reaxys<sup>1</sup> and SciFinder<sup>2</sup> among others. These databases are generally populated by human experts through manual extraction from literature, which is costly, time consuming, and expertise intensive, especially with the exponential growth of scientific chemical publications in recent years.<sup>3</sup> This challenge motivates the development of automated methods for reaction extraction from unstructured literature data.

Information extraction in the chemical domain has gained increasing attention over the past decade. Existing work has concentrated on named entity recognition (NER) and the extraction of their associated properties, such as OSCAR (Open Source Chemistry Analysis Routines),<sup>4</sup> and ChemDataExtractor.<sup>5</sup> Only very few works have targeted the extraction of chemical reactions which, compared to chemical compounds extracted by NER, are more structured, informative, and also practically useful. NER helps in associating compounds with documents, but chemists still need to go to the original article to see the context for that species, whereas reactions are often what the chemist wants to know about. Two representative toolkits developed for this

purpose are ChemicalTagger<sup>6</sup> and OPSIN.<sup>7</sup> ChemicalTagger went beyond entity extraction and used a grammar-based phrase parser to identify action phrases and relationships between entities. It has been specifically developed for extracting information from patents, taking advantage of its highly stylized and formulaic language. OPSIN took a mixture of outputs from ChemicalTagger and employed a set of rules to determine four essential chemical roles, including *product*, *reactant*, *solvent*, and *catalyst*. These rule-based systems represent good starting points for this endeavor, but they are heavily dependent on manually designed rules and are sensitive to the noise introduced by either language use or preprocessing steps, which limits their scalability to nonpatent data such as journal articles. Language used in academic journals is often of higher complexity and less formulaic than patent literature. For instance, one sentence can describe multiple reactions or one reaction with different products/yields under different conditions. This complexity requires the development of more advanced natural language processing (NLP) models with higher capacity. Another type of reaction data which is growing in popularity is synthesis action sequences, which

**Special Issue:** From Reaction Informatics to Chemical Space

**Received:** March 10, 2021

contain details required by a bench chemist or a robotic system to conduct a reaction. Mehr et al. parsed synthetic procedures in literature into machine-executable actions via pattern matching with expert-defined heuristics.<sup>8</sup> Vaucher et al. instead presented a deep-learning sequence to sequence model to convert unstructured experiment procedures into action sequences, using a combination of expert annotation and a rule-based system for training.<sup>9</sup>

In this study, we focus on developing a method for automatically parsing journal articles and extracting reactions into a schema that is consistent with prior databasing efforts like Reaxys and SciFinder. We devised a schema that represents chemical reactions in a unified structured semantic frame, which contains a *major product* as the central element and a set of essential chemical roles as its associated slots. Consider the following passage drawn from the chemical literature:<sup>10</sup> *Reaction of diphenylacetylene with complex 19A led to only cycloheptadienone 23A in 30% yield; with (phenyl-cyclopropyl)-carbene complex 19B, cycloheptadienone 25 was produced in 53% yield.*

There are two reactions described in the passage with products being 23A and 25 respectively (they are identifiers pointing to specific structures in diagrams). For the reaction that produces 23A, *reactants* include *diphenylacetylene* and *19A*, and *yield* is 30%. The same chemical (e.g., *diphenylacetylene* in the example above) can participate in multiple reactions. Note that this schema is not complete in terms of what is needed to reproduce a reaction but can greatly benefit chemists in multiple ways. Besides providing chemists with structured and easily accessible reaction information, data in this format can also be directly utilized in computer-aided chemistry for training automated systems of reaction prediction,<sup>11–14</sup> reaction condition recommendation,<sup>15,16</sup> and synthesis planning.<sup>17–19</sup>

We proposed a two-stage cascading framework for reaction extraction, which consists of two primary submodules: *product extraction* and *reaction role labeling*. At the first stage, a sequence tagging model is employed to recognize all the possible products mentioned in the given (preprocessed) text. For each of the products, a role labeling model is then used to extract all possible reaction roles from their context and fill corresponding slots as defined in the schema. Both models are data driven and built with deep neural networks and thus require annotated data for the training and evaluation in the very first place. To this end, we have defined comprehensive guidelines for annotating chemical literature texts to obtain chemical reaction data, from which task-specific training data can be further compiled for product extraction and reaction role labeling, respectively.

Considering that reaction data sets are both labor intensive and expertise demanding to annotate, we sought to reduce the reliance on a huge amount of labeled data typically required for supervised training of deep neural models. Inspired by the recent dominant *pretraining-and-finetuning* paradigm in NLP,<sup>20</sup> we first pretrained a Transformer-based text encoder, named *ChemBERT*, on vast amounts of unlabeled literature texts. This encoder was then coupled with task-specific decoders and fine-tuned using the limited amount of annotations of each end task. In addition, input texts to reaction role labeling are expected to be *reaction relevant*, i.e., describing at least one chemical reaction and its major product, thus forming a much confined subspace of the general chemical literature texts. Given this fact, we introduced an adaptive pretraining

approach with reaction-relevant text retrieval to find a subspace of the unlabeled data that is more distributionally similar to our target task. Continual pretraining on this subspace produced a task-adaptive encoder, *ChemRxnBERT*, which brought further improvements to reaction role labeling.

Experiments show that our models are both effective and data efficient. We achieved an F1 score of 76.2% for product extraction and 78.7% for role labeling, using only hundreds of training instances for each task. The code, annotated data, and trained models for reaction extraction are publicly available to the community.<sup>21</sup>

## METHODS

**Reaction Schema.** A chemical reaction can be described as a process of chemical transformation from one set of chemical substances to another. A desired reaction schema is thus supposed to be informative enough to reflect such a transformation, primarily including the source chemicals, the outcomes, and the reaction conditions. In addition to being chemically informative, we expect the schema to be succinct and friendly to data-driven models. Following this design principle, we introduced a schema that represents reactions in a unified semantic frame, which contains *product* as a central factor and eight associated reaction roles (*reactants*, *reaction type*, *catalyst/reagents*, *workup reagents*, *solvent*, *temperature*, *time*, and *yield*) as slots to be filled. Table 1 contains detailed explanations for each of the predefined roles in the schema. Figure 1 shows the extracted reactions from an example text using the schema.

**Table 1. Reaction Schema Used in This Work, with Detailed Explanations of Each Specific Role**

Reaction Role	Description
Product	Chemical substance that is the final outcome (major product) of the reaction
Reactants	Chemical substances that contribute heavy atoms to the product
Catalyst/Reagents	Chemical substances that participate in the reaction but do not contribute heavy atoms (e.g., acid, base, metal complexes)
Workup reagents	Chemical substances that are used after the reactions to terminate the reactions or obtain the products (e.g., quenching reagents, extraction solvent, neutralizing acids/bases)
Solvent	Chemical substances that are used to dissolve/mix other chemicals, typically quantified by volume and used in superstoichiometric amounts (e.g., water, toluene, THF)
Temperature	Temperature at which the reaction occurs
Time	Duration of the reaction performed
Reaction type	Descriptions about the type of chemical reaction
Yield	Yield of the product

**Data and Annotations.** In contrast to the majority of published chemical information extraction tools that use patent information, the target source of text we considered in this work is journal articles. To this aim, we used a collection of ~200,000 published articles in multiple chemistry journals from 1906 to 2016. Details are shown in Table 2 regarding the number of articles per journal and Figure 2 regarding the number of articles by publication date.

For each reaction-relevant article, only a few passages of the whole body text contain well-formed reaction descriptions, and they are usually not explicitly sectioned. We first employed a set of rules based on keywords matching (e.g., *afford/s/ed*,

We found that compound 3a could be effectively transformed to (E)-3-methyleneisindolin-1-one in the presence of a catalytic amount of both  $K_2CO_3$  and  $Bu_3P$  at 60 °C, while it was fully transformed to (Z)-3-methyleneisindolin-1-one at 150 °C in the presence of  $Bu_3P$ .



<b>Reaction 1</b>	
Product:	(E)-3-methyleneisindolin-1-one
Roles:	
Reactant:	['3a']
Catalysts:	[' $K_2CO_3$ ', ' $Bu_3P$ ']
Temperature:	60 °C
<b>Reaction 2</b>	
Product:	(Z)-3-methyleneisindolin-1-one
Roles:	
Reactant:	['3a']
Catalysts:	[' $Bu_3P$ ']
Temperature:	150 °C

**Figure 1.** Example of extracted reactions using the proposed schema. The passage was drawn from Chen et al.<sup>22</sup>

**Table 2. Number of Articles per Journal in Our Corpus**

Journal name	Articles
Journal of the American Chemical Society	95,668
The Journal of Organic Chemistry	72,482
Organic Letters	23,631
Journal of Organic Chemistry	295
Organic Process Research & Development	2440

yield/s/ed, produce/s/ed, etc.) and section filtering for the selection of the passages most likely containing reaction information. In particular, we discarded the experimental sections, as reaction descriptions in these sections often contain very detailed procedures about the synthesis, which are not well aligned to our schema. The resulting passages are then preprocessed (sentence splitting, tokenization) using the *ChemDataExtractor* toolkit.<sup>5</sup>

Next, we fed all preprocessed passages into our annotation tool<sup>23</sup> built on Amazon Mechanical Turk (MTurk). We equipped the tool with rich features that allow annotators to (1) annotate and validate reaction roles, (2) reject and classify a paragraph, and (3) consult the original article for greater context.

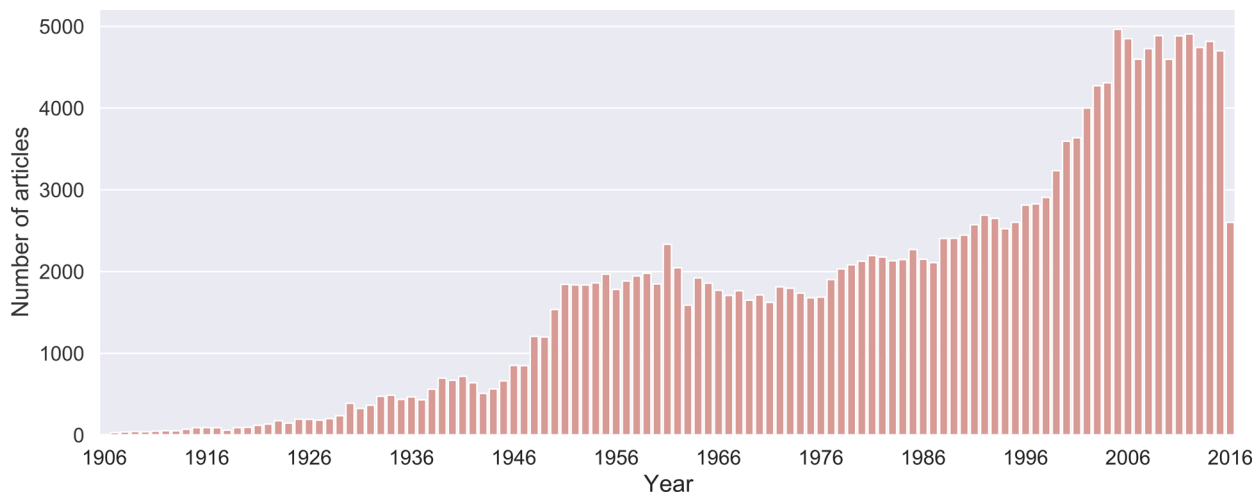
We employed 13 graduate students and postdocs in chemistry or chemical engineering laboratories and two postdocs in computer science for the first-round annotation. Then, we manually checked the annotation quality and consistency and refined our annotation guideline by clarifying some ambiguous terms, followed with an additional overall

validation process by the same annotators. The entire annotation process took 280–240 h for the first-round annotation with a passage-level accuracy of 89.3%, and 40 h for the refining phase. The resulting corpora contains 329 passages, each annotated with one or more reactions. We followed a 8:1:1 ratio to split our corpora into training, development, and test sets. Table 3 summarizes the data statistics. The scarcity of training data raises severe challenges to learning a high-performance model.

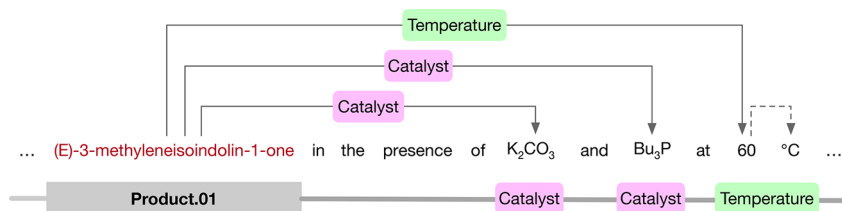
**Table 3. Data Statistics of Annotated Reaction Corpus, Including Number of Passages, Reactions, Passages with Multiple Reactions, and Product–Role Relation Pairs**

	Passages	Reactions	Passages (multireactions)	Product–Roles (relation pairs)
Train	251	599	159	2457
Development	41	96	22	482
Test	37	111	22	469

**Model. Task Formulation.** We formulated the reaction extraction task as a structure prediction problem which takes a sequence of tokens as input, and outputs the reaction structures, each containing a set of *product–role* relation pairs. We proposed a two-stage pipeline framework, combining a *product extraction* module and a *reaction role labeling* module for the extraction of reactions. At the first stage, the *product extraction* module aimed to identify all possible product entities from the given text. For each of the products, the *reaction role*



**Figure 2.** Number of journal articles by publication date in our corpus.



**Figure 3.** Reaction role extraction, a relation extraction problem (top), here formulated as a sequence labeling task conditioned on a given *product* (bottom). “Product.01” indicates the first product in the current text.

labeling module was then used to extract the associated elements corresponding to different reaction roles presented in its context, which together form the final reaction structure. The two pipelined modules were trained independently, and we compiled task-specific training data for each of them from our annotated corpora.

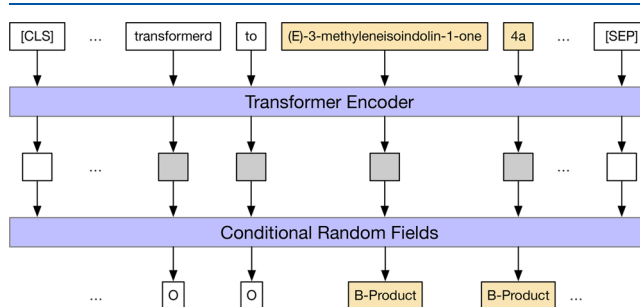
Product extraction can be formulated as a standard sequence labeling task over individual words. Role labeling, however, is essentially a *relation extraction* task aiming to identify the reaction role entities and classify their relationship to a given product into one of the predefined role categories. We formulated it as a conditional sequence labeling task by adding special markers to the input in order to inform the encoder about the target product, so that predictions for the related role tokens will be conditioned on both the input text and the given product. Figure 3 illustrates how the role labeling task is formulated.

In the rest of this section, we first introduce our architectural design of each module and then describe an adaptive pretraining strategy for effective learning in a low-resource regime.

**Product Extraction.** The goal of product extraction is to identify all entity spans that refer to certain items of chemical reactions. We assumed there were no nested products in the text and formulated this task as a sequence tagging problem under the “BIO” tagging scheme.<sup>24</sup> Specifically, given an input sequence of tokens, our model aimed to assign to each token a categorical label in the form of “[BI]-Type” or “O”, where “BI” indicates the position of a word within an entity span—“B” denotes the beginning of an entity and “I” denotes inside an entity—and “O” indicates that a token belongs to no entity. “Type” is a placeholder for any entity type to be extracted. In the case of product extraction, the only entity type of interest is *Product*.

Words of the input sequence were further tokenized into a set of subword tokens, namely, *wordpieces*,<sup>25</sup> before flowing through a Transformer encoder.<sup>26</sup> For instance, “K<sub>2</sub>CO<sub>3</sub>” was divided into: [“K”, “##2”, “##CO”, “##3”], where all wordpieces except the first one were prefixed with “##”. Using wordpieces can effectively improve the generalizability and robustness of machine learning models, especially for languages whose vocabulary has rich internal structures like chemical names. The Transformer encoder essentially consists of a stack of multihead self-attention layers and feed-forward layers, which computed a hidden representation for each of the wordpieces. We took the first wordpiece of each word as input to a conditional random field (CRF)<sup>27</sup> decoder for sequence labeling. At inference time, we used the Viterbi decoding algorithm<sup>28</sup> with a set of tag transition constraints coming with the “BIO” scheme, for example, “I-Product” must be following “B-Product”.

**Figure 4** illustrates the architecture of our tagging model. In this work, we considered the structure identifiers (e.g., “4a”) as



**Figure 4.** Model architecture of product extraction.

independent entities in order to facilitate future work on bridging texts with chemical diagrams where the identifiers will be used to locate the corresponding molecular structures.

The model was trained with maximum likelihood estimation (MLE). We denote the input sequence as  $\mathbf{x} = \{x_1, \dots, x_n\}$ , where  $x_i$  is the  $i$ th word, and a sequence of labels as  $\mathbf{y} = \{y_1, \dots, y_n\}$ .  $Y(\mathbf{x})$  denotes the set of possible label sequences for  $\mathbf{x}$ . The conditional probability  $P(\mathbf{y}|\mathbf{x}; \theta)$  is given by

$$P(\mathbf{y}|\mathbf{x}; \theta) = \frac{\exp(s(\mathbf{x}, \mathbf{y}))}{\sum_{\mathbf{y}' \in Y(\mathbf{x})} \exp(s(\mathbf{x}, \mathbf{y}'))}$$

where  $s$  is a scoring function combining a transition score and an emission score

$$s(\mathbf{x}, \mathbf{y}) = \sum_{i=0}^n T_{y_i, y_{i+1}} + \sum_{i=1}^n E_{i, y_i}$$

where  $T$  is the transition scoring matrix to be estimated during training, and  $E_{i, y_i}$  corresponds to the score for the  $i$ th word being tagged as  $y_i$ .  $E_i$  is obtained through a fully connected layer which projects the hidden representation of the  $i$ th word (representation of its first wordpiece), denoted as  $\mathbf{h}_i$  to the label space.

$$E_i = \text{ReLU}(\mathbf{W}^{\text{prod}} \mathbf{h}_i + \mathbf{b}^{\text{prod}})$$

where  $\mathbf{W}^{\text{prod}} \in \mathbb{R}^{l \times d_h}$  and  $\mathbf{b}^{\text{prod}} \in \mathbb{R}^{l \times 1}$  are the weight matrices and biases of this linear projection, respectively. We have  $l = 3$  labels for this task.

**Reaction Role Labeling.** For each of the products recognized in the first stage, we proceeded to identify and classify its associated reaction roles into predefined role types in our reaction schema. Consider the example as shown in Figure 5, where our aim is to extract reaction roles for the product *(E)-3-methyleneisindolin-1-one*. To make the encoder aware of the target *product*, we enclosed the product entity with two special markers “[P]” and “[/P]”, thus forming a span



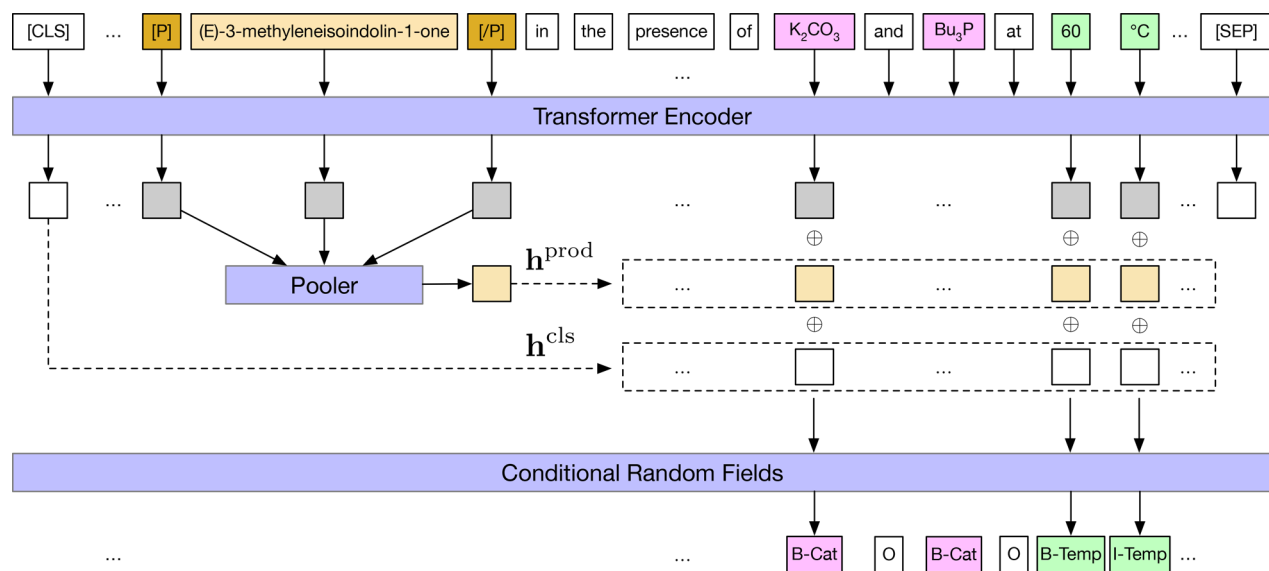


Figure 5. Model architecture of chemical role extraction.

$x_s, x_{s+1}, \dots, x_e$ , where  $s$  indicates the index of “[P]” in the tokenized sequence, and  $e$  is the index of “[/P]”. A *product* representation was then obtained by pooling over all word representations within this span.

$$\mathbf{h}^{\text{prod}} = \text{Pooling}(\mathbf{h}_{s:e})$$

Herein, we used Max-pooling since it gave superior performance than alternatives (e.g., taking the representation of starting token “[P]”) in our experiments.

Conditioned on this product representation, we performed sequence tagging over all the remaining tokens for the recognition of associated reaction roles. As with the product extraction task, we continued using the “BIO” tagging scheme. In the example shown in Figure 5,  $\text{K}_2\text{CO}_3$  and  $\text{Bu}_3\text{P}$  are two catalysts used in the reaction that leads to product (E)-3-methyleneisindolin-1-one, so they are tagged as “B-Cat”, while the temperature phrases  $60\text{ }^\circ\text{C}$  are tagged as “B-Temp” and “I-Temp” respectively. In addition, we found adding a sentence representation (i.e., representation of the “[CLS]” token, denoted as  $\mathbf{h}^{\text{cls}}$ ) which captures global semantics of the input text to be beneficial for the role labeling task. As a consequence, the final representation at each position used as input to the decoder was the concatenation of three vectors, giving the following emission scoring function

$$E_i = \text{ReLU}(\mathbf{W}^{\text{role}}[\mathbf{h}_i \oplus \mathbf{h}^{\text{prod}} \oplus \mathbf{h}^{\text{cls}}] + \mathbf{b}^{\text{role}})$$

where  $\mathbf{W}^{\text{role}} \in \mathbb{R}^{l \times 3d_h}$  and  $\mathbf{b}^{\text{role}} \in \mathbb{R}^{l \times 1}$ . The number of possible labels per token is 17 (eight roles combined with “[B/I]” and an additional “O” indicating *not-a-role*).

**Adaptive Pretraining.** We leveraged the *pretraining-and-finetuning* paradigm to train the product extraction and role labeling models. The key idea was to first pretrain the Transformer encoder on large-scale unlabeled texts using unsupervised objectives and then fine-tune it on task-specific labeled training data, which is of limited size.

We started with the BERT model of Devlin et al.,<sup>20</sup> which has served as a general-purpose Transformer encoder pretrained on texts from mixed sources collected mainly from the BookCorpus<sup>29</sup> and English Wikipedia. BERT was trained using a joint objective of *masked language modeling*

(MLM) and *next sentence prediction* (NSP). By supervised fine-tuning on end tasks, BERT-based models have established new state-of-the-art performance over various NLP benchmarks since its initial development. It is also becoming the dominating language representation model among other variants.

As one of the most successful transfer learning paradigms, it is desirable that the unlabeled data used for pretraining has a similar distribution to the labeled data for task-specific fine-tuning. In our case, however, chemical texts appear to be highly different from the general-domain texts on which BERT has been trained. This distributional divergence raises difficulty for knowledge transferring, making BERT a suboptimal choice for chemical reaction extraction. In this work, we proposed to tailor BERT to the chemical domain, particularly for our reaction extraction task, through adaptive pretraining.

The expected input to the product extraction model can be any text from a chemical article, as there is no explicit clues for reliable predetermination of reaction-included texts except for a limited set of keywords. On the contrary, inputs to the role labeling model were guaranteed to contain at least one product and thus are expected to be *reaction relevant*. This difference in the expected input data distribution necessitates the development of two separate pretraining encoders for the two tasks. In fact, the inputs to the role labeling model should be a subset of the input space of the product extraction model. Therefore, we proposed a cascaded adaptive pretraining approach, which was composed of two phases: the *domain-adaptive pretraining* which produced a chemical domain-specific pretrained encoder (ChemBERT) and the *task-adaptive pretraining* which produced a task-specific pretrained encoder (ChemRxnBERT). These two resulting encoders were used for *product extraction* and *reaction role labeling*, respectively. The workflow is shown in Figure 6.

**ChemBERT.** To adapt BERT to the chemical literature domain (ChemBERT), we collected unlabeled texts from a set of 200,000 chemical journal articles for continual pretraining. After the same preprocessing and data filtering we used in the preparation of data annotation, we ended up with 1,860,693 passages, which have 9,478,043 sentences with over 217 M tokens. Note that we discarded the experiment sections which

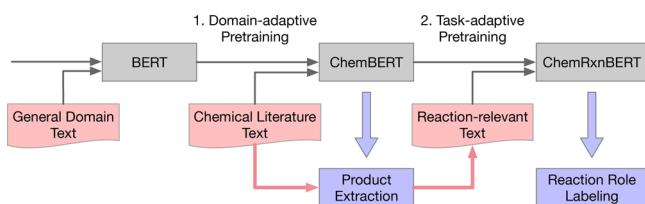


Figure 6. Workflow of domain- and task-adaptive pretraining.

usually contain detailed synthesis steps instead of high-level reaction descriptions. During the pretraining of ChemBERT, we retained the MLM objective with whole word masking but dropped the NSP objective which has been shown in prior studies to be hardly beneficial for most end tasks.<sup>30</sup> The pretrained ChemBERT was then used to initialize the Transformer encoder of the product extraction model (Figure 4) and fine-tuned afterward.

**ChemRxnBERT.** Pretraining of ChemRxnBERT requires a more constrained subset of chemical texts that is better aligned to the target task. The labeled training data of reaction role labeling, however, was insufficient to serve this goal. To address this issue, we proposed to use the product extraction model as a text retriever to automatically identify reaction-relevant data from the full chemical text space. Specifically, sentences that contain at least one product were selected, which gave about 10% (944,733 sentences) of the full unlabeled corpus.

To gain more insights into this process, we took a random sampled set of sentences from the unlabeled chemical texts and the small annotated data of role labeling, encoded them using the representation component (encoder) of the trained product extraction model, and computed their sentence embeddings by averaging contextual embeddings from the last layer. The resulting 768-dimensional sentence embeddings were then reduced to 2-D via principal component analysis (PCA)<sup>31</sup> and visualized in Figure 7a. We can clearly see that

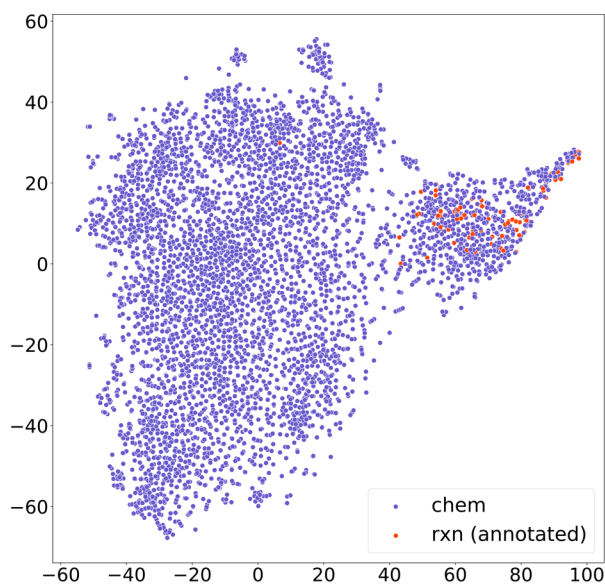
the annotated reaction data distributes compactly in a small subspace of the whole chemical data. Data points in this subspace should compose a desired set of data for pretraining ChemRxnBERT. Figure 7b shows that our retrieved sentences are well aligned with the target reaction data distribution. This subset of data was then used for task-adaptive pretraining.

## RESULTS AND DISCUSSION

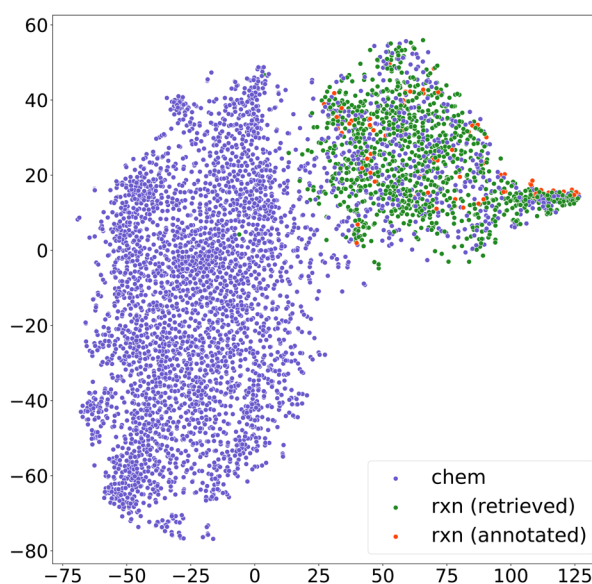
**Evaluation Setup and Baselines.** We experimented with a limited sequence length due to memory and optimization constraints. For product extraction, we found that most of the products can be inferred from context within the same sentence, so we performed sentence-level labeling to find all possible products of a given passage. Identification of roles, however, may involve cross-sentence reasoning in some cases. To determine a reasonable context size, we analyzed the distribution of product–role distances in our corpus, which is shown in Figure 8 (left, sentence-level distance; right, word-level distance). We find that 93% of the reaction roles can be found within a context size of three sentences to their corresponding product and 72% within the same sentence. To this end, we created two experiment settings for role labeling, which used context sizes of three sentences and one sentence, respectively.

We evaluated the performance of product extraction and reaction role labeling models on separate test sets compiled from the annotation data set. In this setting, the reaction role labeling model used the ground-truth product as input. Since both tasks were formulated as sequence tagging, we use the standard metrics including *Precision* (*P*), *Recall* (*R*), and *F1-score* (*F1*). These are defined as

$$\text{Precision} = \frac{TP}{TP + FP}$$

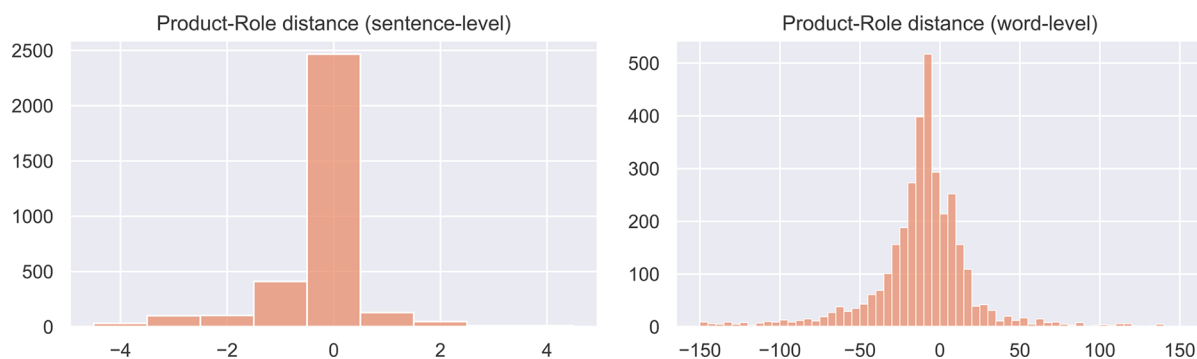


(a) General chemical texts *vs.* annotated reaction texts.



(b) General chemical texts *vs.* annotated reaction texts *vs.* retrieved reaction texts.

Figure 7. 2-D Visualization of chemical text embeddings. (a) Reaction data locate in a small subspace of the full chemical data. (b) Our retrieved reaction data aligned well with the annotated reaction data.



**Figure 8.** Distribution of product–role distances (negatives indicate roles to the left side of the target product).

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F1} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

where TP indicates true positives that the system correctly identified, FP indicates the false positives that the system incorrectly identified, and FN is the false negatives that the system failed to recognize.

We considered the BERT-based counterparts as our primary baseline models for comparison, which use the same architecture as ours except for the use of a general-domain BERT encoder. Throughout our experiments, we referred BERT as the pretrained *bert-base-cased* model officially released. Additionally, we compared to BioBERT, a BERT model pretrained on biomedical literature.<sup>32</sup> For product extraction, we also reported the performance of a pioneering rule-based system, OPSIN,<sup>7</sup> as well as a bidirectional LSTM (BiLSTM), which has been a standard approach for a wide range of tagging tasks in NLP.<sup>33</sup> OPSIN identifies *products* by a set of rules based on the tagging and parsing outputs of ChemicalTagger.<sup>6</sup> It was developed specifically for processing patent literature, which is highly different from journal articles in terms of language use. To implement the BiLSTM-based models, we trained 300-dimensional static word vectors using fastText<sup>34</sup> from the same unlabeled corpus as used for training ChemBERT.

**Product Extraction.** Table 4 presents the performance of product extraction models. As expected, OPSIN gives poor

**Table 4.** Performance of Product Extraction

	P (%)	R (%)	F1 (%)
OPSIN	18.8	5.4	8.4
BiLSTM (w/o CRF)	52.4	46.7	49.4
BiLSTM	54.3	49.1	51.6
BERT	78.8	56.8	66.0
BioBERT	76.4	61.3	68.0
ChemBERT	<b>84.6</b>	<b>69.4</b>	<b>76.2</b>

performance in our data, demonstrating the limit of rule-based methods in the processing of freer language used in journal articles. BERT confirms its strong representation capability and shows substantial gains over the BiLSTM encoders. ChemBERT achieves 10.27% absolute improvements in F1 over BERT, implying the need and effectiveness of domain-adaptive pretraining.

**Reaction Role Labeling.** Performances of reaction role labeling with context sizes of 1 (i.e., sentence-level) and 3 are shown in Table 5. At a sentence-level, ChemBERT achieved

**Table 5.** Performance of Reaction Role Labeling

	context size = 1			context size = 3		
	P (%)	R (%)	F1 (%)	P (%)	R (%)	F1 (%)
BERT	69.2	69.2	69.2	65.8	65.9	65.9
BioBERT	73.3	75.5	74.3	66.2	69.5	67.8
ChemBERT	77.0	76.4	76.7	<b>75.9</b>	<b>71.3</b>	<b>73.5</b>
ChemRxnBERT	<b>79.3</b>	<b>78.1</b>	<b>78.7</b>	70.5	69.6	70.1

substantial gains over BERT, while the task-adaptive ChemRxnBERT gives an additional 2% improvements in F1. We found that ChemBERT outperforms ChemRxnBERT on role labeling when using a larger context size. The reason should be that ChemRxnBERT is adapted from ChemBERT by sentence-level masked language modeling. Pretraining with a greater context size should be more desirable, which we leave as part of future work.

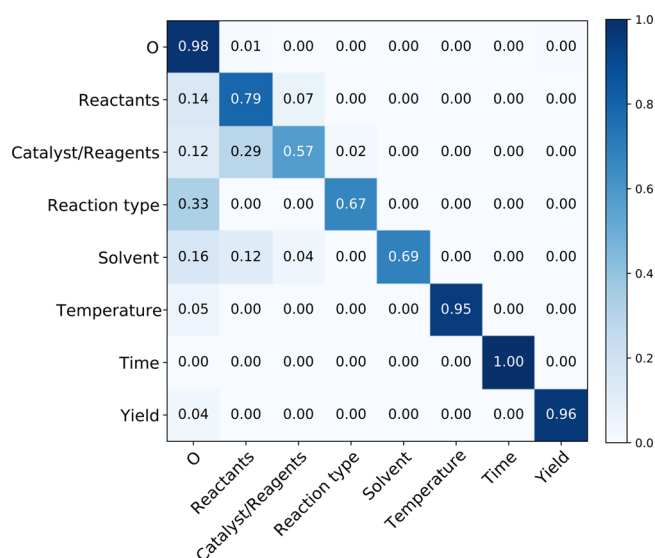
The breakdown performances by reaction role types are shown in Table 6. Some of the reaction roles appear to be

**Table 6.** Performance by Reaction Role Types

Reaction Role	P (%)	R (%)	F1 (%)
Reactants	80	82	81
Catalyst/Reagents	62	54	58
Solvent	92	72	80
Reaction type	86	67	76
Time	100	100	100
Temperature	77	81	79
Yield	76	96	85

more difficult to predict than others, such as *Catalyst/Reagents*. We excluded the *Workup reagents* roles here as they appear only very few times in the data set.

Figure 9 further illustrates, for the labels present in the ground truth, the corresponding labels predicted by our role labeling model. We can see two main types of prediction errors. First, many of the roles mentioned that were unseen in the training data were not successfully identified and thus labeled as O. Other than that, the main errors made by the model relate to the disambiguation between *Catalyst/Reagents* and *Reactants*. This is mainly because these two types of roles usually share similar contexts in a reaction description.



**Figure 9.** Entity-type confusion matrix: rows represent ground truth labels and columns represent predicted labels.

**Qualitative Analysis.** Next, we presented a qualitative analysis of the reactions extracted by our model to demonstrate its capabilities and potential weaknesses.

**Multireactions.** Figure 10 presents a few examples of extracted reactions from our data set. We first show a simple case which contains a single reaction (Example (A)). These cases are comparatively easy to solve, even with prior rule-based approaches. Example (B) is a multistep reaction, in which the product of the first reaction is a reactant for the

production of 29 in the second reaction. Traditional tagging-based or rule-based reaction extraction methods, however, have been unable to handle such cases. Example (C) describes the reaction of a compound ( $\text{CpFe}(\text{CO})_2\text{SiMe}_3$ ) when coupling with different reactants gives different outcomes (product and yield). It is worth noting that the *Yield* of a reaction may not be an exact number but can also be a vaguely-expressed natural language phrase indicating a yield range or even failure of a reaction (e.g., “not give an isolable quantity” in the example). The same applies to *Temperature* and *Time*, for example, “room temperature”, “over 4 h” etc. The data-driven nature of our approach enables us to extract these indicative phrases as reaction roles.

**Catalysts/Reagents vs Reactants.** To better understand the main errors our models make, we provide here a representative example where the *Catalysts/Reagents* roles are mistakenly predicted as *Reactants* (Figure 11). We find that *Catalysts/Reagents* and *Reactants* share a similar set of context patterns, such as “reaction with [ENTITY]”, “by treatment with [ENTITY]”, and “in the presence of [ENTITY]”. In these cases, contexts become less discriminative, while the only clue for resolving the ambiguity between the two roles is the entity itself. This poses additional challenges, as well as opportunities to further improve our model by incorporating potential external domain knowledge (e.g., dictionaries of catalysts/reagents) or chemical constraints of a valid reaction (e.g., atom mapping).

**Comparison with Reaxys.** To better understand the strengths and limitations of our approach, we conducted qualitative comparison between the reactions extracted by our system to the manually constructed Reaxys database.<sup>1</sup> We

#### Example (A): Single-Reaction

We were excited to find that , with 2.0 equiv of copper acetate and DMSO as the solvent ,  

 2-(3-(methylthio)naphthalen-2-yl)pyridine was obtained as a single product in 89 % yield at 125 °C ( Table 1 ) .

#### Example (B): Multi-Reaction

A mixture of sodium [1,2-<sup>13</sup>C<sub>2</sub>]acetate and sodium [1-<sup>14</sup>C]acetate was refluxed with trimethylphosphate to afford  

 methyl [1,2-<sup>13</sup>C<sub>2</sub>, 1-<sup>14</sup>C]acetate , which was converted to its lithium enolate ( 29 ) , which was added to the  

 imidazole derivative of 1-methylpyrrolidine-2-acetic acid ( 32 ) .

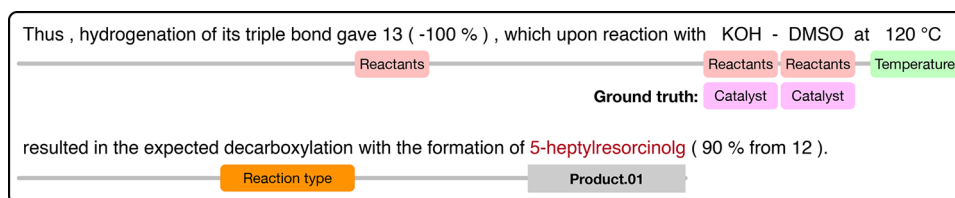
#### Example (C): Multi-Reaction

In an analogous fashion , the photolysis of  $\text{CpFe}(\text{CO})_2\text{SiMe}_3$  in the presence of 1a produced 2a in 21 %  

 yield , whereas the photolysis in the presence of 1b,c did not give an isolable quantity of 2b,c .

**Figure 10.** Examples of reactions extracted by our model. Passages were drawn respectively from Sharma et al.,<sup>35</sup> Leete et al.,<sup>36</sup> and Tobita et al.<sup>37</sup>





**Figure 11.** Incorrect prediction made by the model in distinguishing between Catalyst/Reagents and Reactants. The passage was drawn from Nicolaou et al.<sup>38</sup>

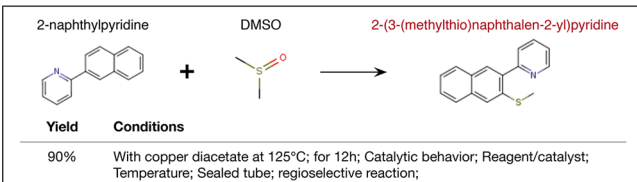
selected the three example passages in Figure 10 to analyze the differences. The corresponding reaction records in Reaxys were retrieved using the digital object identifiers (DOI). Below we summarize the major findings:

**Mismatch in Reaction Role Categorization.** Most compounds can be categorized in several ways. This ambiguity often results in different annotations produced by our system and Reaxys. For instance, in Figure 12, “DMSO” was identified

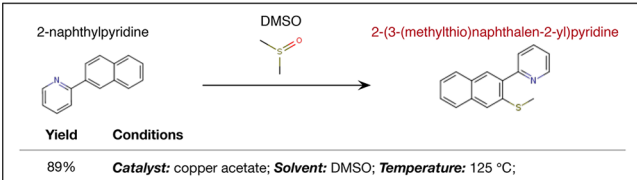
**Passage** (DOI: 10.1021/acs.joc.5b00443)

... To confirm these initial observations, we decided to extend the reaction conditions to 2-naphthylpyridine (1b) as the substrate. We were excited to find that, with 2.0 equiv of copper acetate and DMSO as the solvent, 2-(3-(methylthio)naphthalen-2-yl)pyridine was obtained as a single product in 89 % yield at 125 °C ( Table 1 ).

#### Reaxys



#### ChemRxnExtractor



**Figure 12.** Comparison between the extracted reaction of our system (ChemRxnExtractor) with the manually abstracted reaction in Reaxys for Example (A) (Figure 10). Chemical names are converted to structural formulas for better demonstration.

as a *solvent* by our system, which conforms to the text description (“DMSO as the solvent”). Reaxys instead categorized DMSO as a *reactant*, as DMSO had indeed participated in this reaction as a sulfur source.

**Rounding vs Exact Reporting of Numerical Values.** We noticed that in some cases Reaxys reports rounded numerical values. In contrast, our system is designed to report exact values as stated in input articles. This is illustrated in the reaction yield value (Figure 12), extracted by our system as 89% as stated in the text and rounded to 90% in Reaxys.

**Ability to Extract from Global Contexts.** Our extractions are based on a limited context scope (i.e., passage) and thus can fail to extract certain reaction roles whose inference requires global context (e.g., full document). For instance, in Figure 12, Reaxys includes the *Time* condition (“12h”) and additional conditions such as the *reaction procedure* (“Sealed tube”), which our system failed to extract. While in the original article these are described in a separate section and apply to all

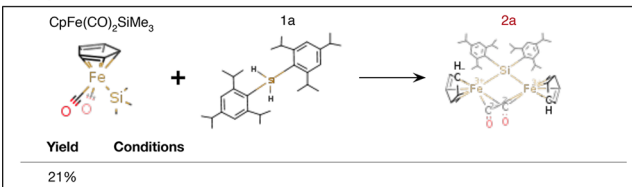
the experiments performed in the article, in this specific passage they are not mentioned.

**Chemical Entity Grounding.** In reaction descriptions, chemicals are often represented by identifiers linking to specific structural depictions in diagrams (e.g., Figure 13).

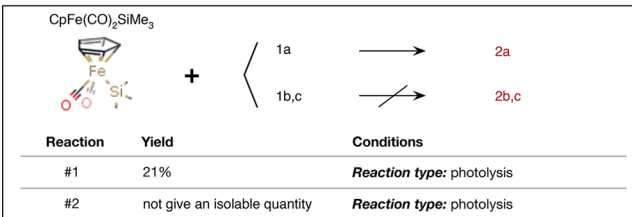
**Passage** (DOI: 10.1021/ja00131a029)

... In an analogous fashion, the photolysis of  $\text{CpFe}(\text{CO})_2\text{SiMe}_3$  in the presence of 1a produced 2a in 21 % yield, whereas the photolysis in the presence of 1b,c did not give an isolable quantity of 2b,c.

#### Reaxys



#### ChemRxnExtractor



**Figure 13.** Comparison between ChemRxnExtractor with the manually abstracted reaction in Reaxys for Example (C) (Figure 10).

Therefore, chemical entity grounding is a critical step before populating the extracted reactions into databases. In Reaxys, these chemicals are manually grounded by human experts. In contrast, our automated system should be coupled with additional optical chemical structural recognition (OCSR) tools for chemical entity grounding. OCSR has been an important and challenging step toward fully automated chemical literature mining. Existing efforts include rule-based methods<sup>39</sup> and the recent deep learning-based models.<sup>40,41</sup> However, the development of a sufficiently accurate, robust, and open-source solution for OCSR remains a challenge.

**Reaction Coverage.** Negative reactions or failures (e.g., Reaction #2 in Figure 13) are mostly ignored in Reaxys. These negative data can be of important scientific value, and this work demonstrates the potential to systematically extract them from chemical literature. Some reactions may also not be included in Reaxys due to potential human preference. For instance, the reactions in Example (B) of Figure 10 were not recorded in Reaxys. This is a reaction to produce an intermediate used in subsequent syntheses and thus is likely to be considered “nonessential” compared to other reactions described in the article and thus is neglected.

## CONCLUSION

This work implemented an automated system for reaction extraction from chemical literature. We introduced a new product-centric chemical reaction schema aligning with existing manually curated commercial databases, and collected a small amount of annotations following this schema. The task was decomposed into two cascaded subtasks, namely product extraction and reaction role labeling, and individual modules were developed for each of them. Both modules were built on an encoder-decoder framework, in which a Transformer is used as the encoder, and conditional random fields as the decoder for (conditional) sequence labeling. To cope with the data-scarce challenge, we proposed domain- and task-adaptive pretraining using large-scale unlabeled corpus extracted from the literature. Our system was able to achieve an F1 score of 76.2% for product extraction and 78.7% for role extraction, which significantly outperformed prior rule-based approaches, as well as stronger BERT and BioBERT baseline models. Qualitative analysis on multireactions extraction showed that our system was indeed able to uncover complex product-role relations in texts. Meanwhile, the current system still makes mistakes in distinguishing Catalysts/Reagents and Reactants due to their largely shared context patterns. Finally, we compared our extractions to the reaction records in the manually constructed Reaxys database and analyzed the strengths and limitations of our approach, which sheds light on future directions.

## DATA AND SOFTWARE AVAILABILITY

The 200,000 journal articles used in this work were shared between the American Chemical Society and MIT under a private access agreement. Our annotated corpus, code, and pretrained models are publicly available under the MIT license on GitHub.<sup>21</sup>

## ASSOCIATED CONTENT

### Supporting Information

The Supporting Information is available free of charge at <https://pubs.acs.org/doi/10.1021/acs.jcim.1c00284>.

Link to the code and demonstration of our annotation tool and detailed description of the annotation guidelines, including the overall annotation process, and general annotation rules regarding each type of chemical entities (product, reaction roles) defined in our reaction schema (PDF)

## AUTHOR INFORMATION

### Corresponding Author

**Regina Barzilay** – Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, Massachusetts 02139, United States; Email: [regina@csail.mit.edu](mailto:regina@csail.mit.edu)

### Authors

**Jiang Guo** – Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, Massachusetts 02139, United States; [orcid.org/0000-0002-9816-805X](https://orcid.org/0000-0002-9816-805X)

**A. Santiago Ibanez-Lopez** – Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, Massachusetts 02139, United States

**Hanyu Gao** – Department of Chemical Engineering, MIT, Cambridge, Massachusetts 02139, United States; [orcid.org/0000-0002-6346-0739](https://orcid.org/0000-0002-6346-0739)

**Victor Quach** – Computer Science and Artificial Intelligence Laboratory, MIT, Cambridge, Massachusetts 02139, United States

**Connor W. Coley** – Department of Chemical Engineering, MIT, Cambridge, Massachusetts 02139, United States; [orcid.org/0000-0002-8271-8723](https://orcid.org/0000-0002-8271-8723)

**Klavs F. Jensen** – Department of Chemical Engineering, MIT, Cambridge, Massachusetts 02139, United States

Complete contact information is available at:

<https://pubs.acs.org/10.1021/acs.jcim.1c00284>

## Author Contributions

<sup>§</sup>J. Guo and A. S. Ibanez-Lopez contributed equally to this work.

## Notes

The authors declare no competing financial interest.

## ACKNOWLEDGMENTS

This work was supported by the DARPA Accelerated Molecular Discovery (AMD) program under contract HR00111920025, the Machine Learning for Pharmaceutical Discovery and Synthesis Consortium (MLPDS), and the Defence Threat Reduction Agency under contract HDTRA12110013. We specially thank the American Chemical Society for sharing the journal articles, and Elsevier for the access to Reaxys. We also thank all the students and postdocs who helped with the data annotation. In addition, we thank Juan Ortiz for his work on our Web-based user interface.<sup>42</sup>

## REFERENCES

- (1) Reaxys. <https://www.reaxys.com> (accessed May 12, 2021).
- (2) SciFinder. <https://scifinder.cas.org> (accessed May 12, 2021).
- (3) Larsen, P.; Von Ins, M. The rate of growth in scientific publication and the decline in coverage provided by Science Citation Index. *Scientometrics* **2010**, *84*, 575–603.
- (4) Jessop, D. M.; Adams, S. E.; Willighagen, E. L.; Hawizy, L.; Murray-Rust, P. OSCAR4: a flexible architecture for chemical text-mining. *J. Cheminf.* **2011**, *3*, 1–12.
- (5) Swain, M. C.; Cole, J. M. ChemDataExtractor: a toolkit for automated extraction of chemical information from the scientific literature. *J. Chem. Inf. Model.* **2016**, *56*, 1894–1904.
- (6) Hawizy, L.; Jessop, D. M.; Adams, N.; Murray-Rust, P. ChemicalTagger: A tool for semantic text-mining in chemistry. *J. Cheminf.* **2011**, *3*, 17.
- (7) Lowe, D. M. Extraction of Chemical Structures and Reactions from the Literature. Ph.D. Thesis, University of Cambridge, 2012.
- (8) Mehr, S. H. M.; Craven, M.; Leonov, A. I.; Keenan, G.; Cronin, L. A universal system for digitization and automatic execution of the chemical synthesis literature. *Science* **2020**, *370*, 101–108.
- (9) Vaucher, A. C.; Zipoli, F.; Geluykens, J.; Nair, V. H.; Schwaller, P.; Laino, T. Automated extraction of chemical synthesis actions from experimental procedures. *Nat. Commun.* **2020**, *11*, 1–11.
- (10) Herndon, J. W.; Chatterjee, G.; Patel, P. P.; Matasi, J. J.; Tumer, S. U.; Harp, J. J.; Reid, M. D. Cyclopropylcarbene-tungsten complexes. alkynes: a [4+ 2+ 1] cycloaddition route for the construction of seven-membered rings. *J. Am. Chem. Soc.* **1991**, *113*, 7808–7809.
- (11) Coley, C. W.; Barzilay, R.; Jaakkola, T. S.; Green, W. H.; Jensen, K. F. Prediction of organic reaction outcomes using machine learning. *ACS Cent. Sci.* **2017**, *3*, 434–443.
- (12) Jin, W.; Coley, C. W.; Barzilay, R.; Jaakkola, T. Predicting Organic Reaction Outcomes with Weisfeiler-Lehman Network. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017; pp 2604–2613.

- (13) Schwaller, P.; Gaudin, T.; Lanyi, D.; Bekas, C.; Laino, T. Found in Translation: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chem. Sci.* **2018**, *9*, 6091–6098.
- (14) Coley, C. W.; Jin, W.; Rogers, L.; Jamison, T. F.; Jaakkola, T. S.; Green, W. H.; Barzilay, R.; Jensen, K. F. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* **2019**, *10*, 370–377.
- (15) Gao, H.; Struble, T. J.; Coley, C. W.; Wang, Y.; Green, W. H.; Jensen, K. F. Using machine learning to predict suitable conditions for organic reactions. *ACS Cent. Sci.* **2018**, *4*, 1465–1476.
- (16) Maser, M. R.; Cui, A. Y.; Ryou, S.; DeLano, T. J.; Yue, Y.; Reisman, S. E. Multilabel classification models for the prediction of cross-coupling reaction conditions. *J. Chem. Inf. Model.* **2021**, *61*, 156–166.
- (17) Coley, C. W.; Green, W. H.; Jensen, K. F. Machine learning in computer-aided synthesis planning. *Acc. Chem. Res.* **2018**, *51*, 1281–1289.
- (18) Segler, M. H.; Preuss, M.; Waller, M. P. Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **2018**, *555*, 604–610.
- (19) Genheden, S.; Thakkar, A.; Chadimová, V.; Reymond, J.-L.; Engkvist, O.; Bjerrum, E. AiZynthFinder: a fast, robust and flexible open-source software for retrosynthetic planning. *J. Cheminf.* **2020**, *12*, 1–9.
- (20) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019; 4171–4186.
- (21) ChemRxnExtractor. <https://github.com/jiangfeng1124/ChemRxnExtractor> (accessed May 12, 2021).
- (22) Chen, X.; Ge, F.-F.; Lu, T.; Zhou, Q.-F. Stereoselective synthesis of 3-methyleneisindolin-1-ones via base-catalyzed intermolecular reactions of electron-deficient alkynes with N-hydroxyphthalimides. *J. Org. Chem.* **2015**, *80*, 3295–3301.
- (23) ChemIE-Turk. <https://github.com/asibanez/chemie-turk> (accessed May 12, 2021).
- (24) Ramshaw, L. A.; Marcus, M. P. *Natural Language Processing Using Very Large Corpora*; Springer, 1999; pp 157–176.
- (25) Wu, Y.; Schuster, M.; Chen, Z.; Le, Q. V.; Norouzi, M.; Macherey, W.; Krikun, M.; Cao, Y.; Gao, Q.; Macherey, K.; Klingner, J.; Shah, A.; Johnson, M.; Liu, X.; Kaiser, L.; Gouws, S.; Kato, Y.; Kudo, T.; Kazawa, H.; Stevens, K.; Kurian, G.; Patil, N.; Wang, W.; Young, C.; Smith, J.; Riesa, J.; Rudnick, A.; Vinyals, O.; Corrado, G.; Hughes, M.; Dean, J. Google's Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv preprint*, arXiv:1609.08144, 2016.
- (26) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; Polosukhin, I. Attention Is All You Need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, 2017; pp 6000–6010.
- (27) Lafferty, J. D.; McCallum, A.; Pereira, F. C. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, 2001; pp 282–289.
- (28) Forney, G. D. The viterbi algorithm. *Proc. IEEE* **1973**, *61*, 268–278.
- (29) Zhu, Y.; Kiros, R.; Zemel, R.; Salakhutdinov, R.; Urtasun, R.; Torralba, A.; Fidler, S. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. *Proceedings of the IEEE international conference on computer vision* **2015**, 19–27.
- (30) Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A Robustly Optimized Bert Pretraining Approach. *arXiv preprint*, arXiv:1907.11692, 2019.
- (31) Wold, S.; Esbensen, K.; Geladi, P. Principal component analysis. *Chemom. Intell. Lab. Syst.* **1987**, *2*, 37–52.
- (32) Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; Kang, J. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240.
- (33) Ma, X.; Hovy, E. End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* **2016**, 1064–1074.
- (34) Bojanowski, P.; Grave, E.; Joulin, A.; Mikolov, T. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* **2017**, *5*, 135–146.
- (35) Sharma, P.; Rohilla, S.; Jain, N. Copper acetate-DMSO promoted methylthiolation of arenes and heteroarenes. *J. Org. Chem.* **2015**, *80*, 4116–4122.
- (36) Leete, E.; Bjorklund, J. A.; Couladis, M. M.; Kim, S. H. Late intermediates in the biosynthesis of cocaine: 4-(1-Methyl-2-pyrrolidinyl)-3-oxobutanoate and methyl ecgonine. *J. Am. Chem. Soc.* **1991**, *113*, 9286–9292.
- (37) Tobita, H.; Izumi, H.; Ohnuki, S.; Ellerby, M. C.; Kikuchi, M.; Inomata, S.; Ogino, H. Silylene-Bridged Dinuclear Complexes Having a Triplet Ground State: Photochemical Synthesis and Structural Characterization of Cp<sub>2</sub>Fe<sub>2</sub> (μ-CO)<sub>2</sub> (μ-SiR)<sub>2</sub> (Cp = η<sup>5</sup>-C<sub>5</sub>H<sub>5</sub>; R = 2, 4, 6-C<sub>6</sub>H<sub>2</sub>iPr<sub>3</sub>, 2, 6-C<sub>6</sub>H<sub>3</sub>Et<sub>2</sub>, and Mesityl). *J. Am. Chem. Soc.* **1995**, *117*, 7013–7014.
- (38) Nicolaou, K.; Dai, W. M.; Hong, Y.; Baldrige, K.; Siegel, J.; Tsay, S. Molecular design, chemical synthesis, kinetic studies, calculations, and biological studies of novel enediynes equipped with triggering, detection, and deactivating devices. Model dynemicin A epoxide and cis-diol systems. *J. Am. Chem. Soc.* **1993**, *115*, 7944–7953.
- (39) Filippov, I. V.; Nicklaus, M. C. Optical Structure Recognition Software to Recover Chemical Information: OSRA, an Open Source Solution. *J. Chem. Inf. Model.* **2009**, *49*, 740–743.
- (40) Oldenhof, M.; Arany, A.; Moreau, Y.; Simm, J. ChemGrapher: optical graph recognition of chemical compounds by deep learning. *J. Chem. Inf. Model.* **2020**, *60*, 4506–4517.
- (41) Staker, J.; Marshall, K.; Abel, R.; McQuaw, C. M. Molecular structure extraction from documents using deep learning. *J. Chem. Inf. Model.* **2019**, *59*, 1017–1029.
- (42) ChemRxnExtractor: Chemical Reaction Extraction from Scientific Literature. <http://chemie.csail.mit.edu> (accessed June 10, 2021).