Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Check for updates

Cite this: *Digital Discovery*, 2024, **3**, 1822

Extracting structured data from organic synthesis procedures using a fine-tuned large language model†

Qianxiang Ai,[a] Fanwang Meng,[a] Jiale Shi,[a] Brenden Pelkie[b] and Connor W. Coley*[a]

Received 6th April 2024
Accepted 30th July 2024

DOI: 10.1039/d4dd00091a

rsc.li/digitaldiscovery

The popularity of data-driven approaches and machine learning (ML) techniques in the field of organic chemistry and its various subfields has increased the value of structured reaction data. Most data in chemistry is represented by unstructured text, and despite the vastness of the organic chemistry literature (papers, patents), manual conversion from unstructured text to structured data remains a largely manual endeavor. Software tools for this task would facilitate downstream applications such as reaction prediction and condition recommendation. In this study, we fine-tune a large language model (LLM) to extract reaction information from organic synthesis procedure text into structured data following the Open Reaction Database (ORD) schema, a comprehensive data structure designed for organic reactions. The fine-tuned model produces syntactically correct ORD records with an average accuracy of 91.25% for ORD "messages" (*e.g.*, full compound, workups, or condition definitions) and 92.25% for individual data fields (*e.g.*, compound identifiers, mass quantities), with the ability to recognize compound-referencing tokens and to infer reaction roles. We investigate its failure modes and evaluate performance on specific subtasks such as reaction role classification.

1 Introduction

Data-driven methods are now routinely employed in the physical sciences. A trend toward the use of supervised machine learning (ML) techniques has increased the need for structured data, *i.e.*, data represented using a standardized data schema. In most scientific communities, however, data is stored and communicated predominantly *via* unstructured documents and prose, with only a few exceptions.[1] Synthetic organic chemistry is not one of those exceptions. Reaction procedures and details are commonly recorded as free text in journal publications, patents, or electronic lab notebooks (ELNs). Manual information extraction and curation are still widely used to construct structured datasets from unstructured texts.[2,3] An automated method to extract structured reaction data from unstructured texts would accelerate efforts to use historical reaction data for data-driven discovery.

As an information extraction task, structured data extraction from text can be considered as a combination of named entity recognition (NER) and relation extraction (RE) between named entities. Challenges in chemical NER include the pervasive usage of abbreviations and aliases, deviations from standard nomenclature, and the ambiguous boundaries between which a chemical entity is defined (*e.g.*, when multiple words describe a single species).[4,5] A variety of methods have been applied for chemical NER tasks. Rule-based or dictionary-based methods, such as LeadMine[6] and ChemicalTagger,[7] have been used to annotate reaction procedure texts or in the text parsing pipeline for constructing synthesis datasets such as SureCHEMBL,[8] Pistachio,[9] and ZeoSyn.[10] While these algorithms are usually computationally efficient, the scope of rules and dictionary items limits their generalizability to new datasets. Various statistical model-based NER algorithms have also been proposed, often as a sequence labeling problem where the tokens in a sentence are assigned most likely tags based on token features. A popular strategy is the use of conditional random fields[11] in combination with expert-selected features[12] or contextualized word embeddings from neural networks (recurrent networks,[13–15] or transformers[16–19]).

Traditionally, RE is formulated as a downstream task to NER and is solved as an ensemble of classification problems for entity pairs.[20,21] More recent efforts aim to solve NER and RE simultaneously by building end-to-end models.[22–25] This trend has persisted as pretrained large language models (LLMs) have become more accessible. LLMs have been used for NER/RE tasks in biomedicine,[26] materials,[27] and clinical trials,[28] showing promise as tools for structured data extraction. For example, Dagdelen *et al.* developed a training pipeline for GPT-3 to extract information from scientific texts about crystalline materials as structured JSON[29] and Walker *et al.* present an iterative scheme to fine-

[a] *Department of Chemical Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA. E-mail: ccoley@mit.edu*

[b] *Department of Chemical Engineering, University of Washington, Seattle, WA, USA*

† Electronic supplementary information (ESI) available. See DOI: **https://doi.org/10.1039/d4dd00091a**

tune LLMs for extracting structured data of gold nanorods synthesis.[30] Recent studies by Zhong *et al.* explored fine-tuned LLMs for reaction data extraction from literature in PDF format.[31,32] The output of these models provides a reasonable coverage of reaction information, with the exception of quantity information. Pretrained LLMs can also be used for this task directly without fine-tuning. For example, a recent preprint by Patiny and Godin explores extracting analytical experiment results from literature solely through prompt engineering.[33] While this method can extract structured data by including in-prompt data schema, it relies on closed-source LLMs and performs poorly when numerical values are involved.

One important use case for extracting structured reaction data is the production of procedural instructions to be used for reproducing experiments. For example, Vaucher *et al.* developed a transformer-based model to translate sentences of experiment procedures into action sequences.[34] While these action sequences contain detailed information for execution, their evaluations focus more on the type of action than the parameters or objects of that action. SynthReader,[35] a rule-based translator developed by Mehr *et al.*,[35] converts natural language procedures to χDL, a data schema designed for chemical operations. Such a rule-based method, despite being computationally efficient, has to be expanded/modified to adapt to a different distribution, *e.g.*, a change in writing style. Various submissions to Cheminformatics Elsevier Melbourne University (ChEMU) evaluation lab[36–38] also aim to solve the NER/RE tasks including reaction/workup steps. Since these campaigns aim at evaluating individual NER/RE tasks, they do not constitute an end-to-end solution for structured data extraction into a specific output data schema.

In this study, we fine-tune an open-source large language model to extract structured reaction information from unstructured text from US patents (Fig. 1). To structure the desired outputs, we adopt the Open Reaction Database (ORD) data format, a comprehensive data schema tailored to organic reactions.[40] The 100 000-reaction dataset we use for fine-tuning is part of a collection originally published by Lowe in Chemical Markup Language (CML) format,[39] so the fine-tuned model essentially pursues the same goal as Lowe's expert natural language processing pipeline, albeit using a different data schema. Extracted records cover information on reactants, products, conditions, and workup steps. We demonstrate that the fine-tuned model produces syntactically correct ORD records from the USPTO with an average accuracy of 91.25% for chemical messages (compounds, workups, conditions) and 92.25% for individual data fields. We also investigate its failure modes and evaluate performance on reaction role classification. We note that a preliminary version of this study was previously disclosed as part of a Perspective article on opportunities for LLMs in chemistry.[42]

2 Methods

2.1 Introduction to the Open Reaction Database (ORD) schema

A reaction record in the ORD is structured as a Reaction message using Google's Protocol Buffers, which can be faithfully converted to and from JSON format without loss of information. For a specific Reaction, we focus on four chemically important fields: inputs, conditions, workups, and outcomes, each of which is also a message or a list of messages defined in ORD schema. An example reaction record is shown in Fig. 2 with representative fields populated. There are more than 600 fields defined in ORD schema, some of which are size-mutable, and an ORD record typically includes many nested messages. There are also strict rules on types and values admitted by data fields. For example, the type field of ReactionWorkup is an enum field that only accepts specific strings, and assigning out-of-vocabulary strings to this field leads to a syntactically invalid ORD record. The full definition of the Reaction message used in this study is available on GitHub.[44]

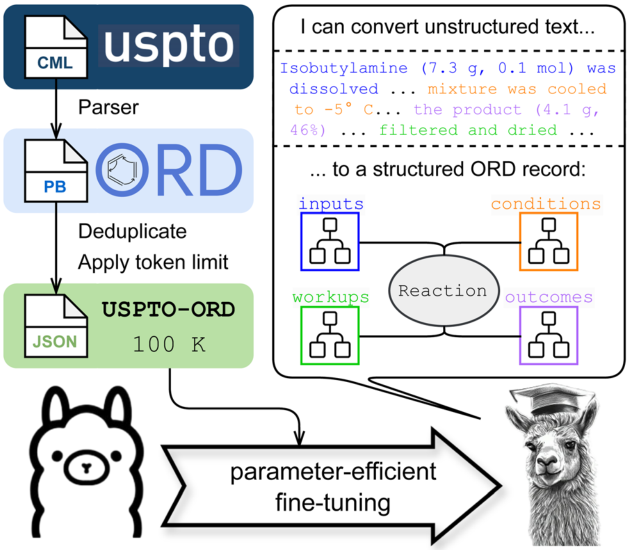
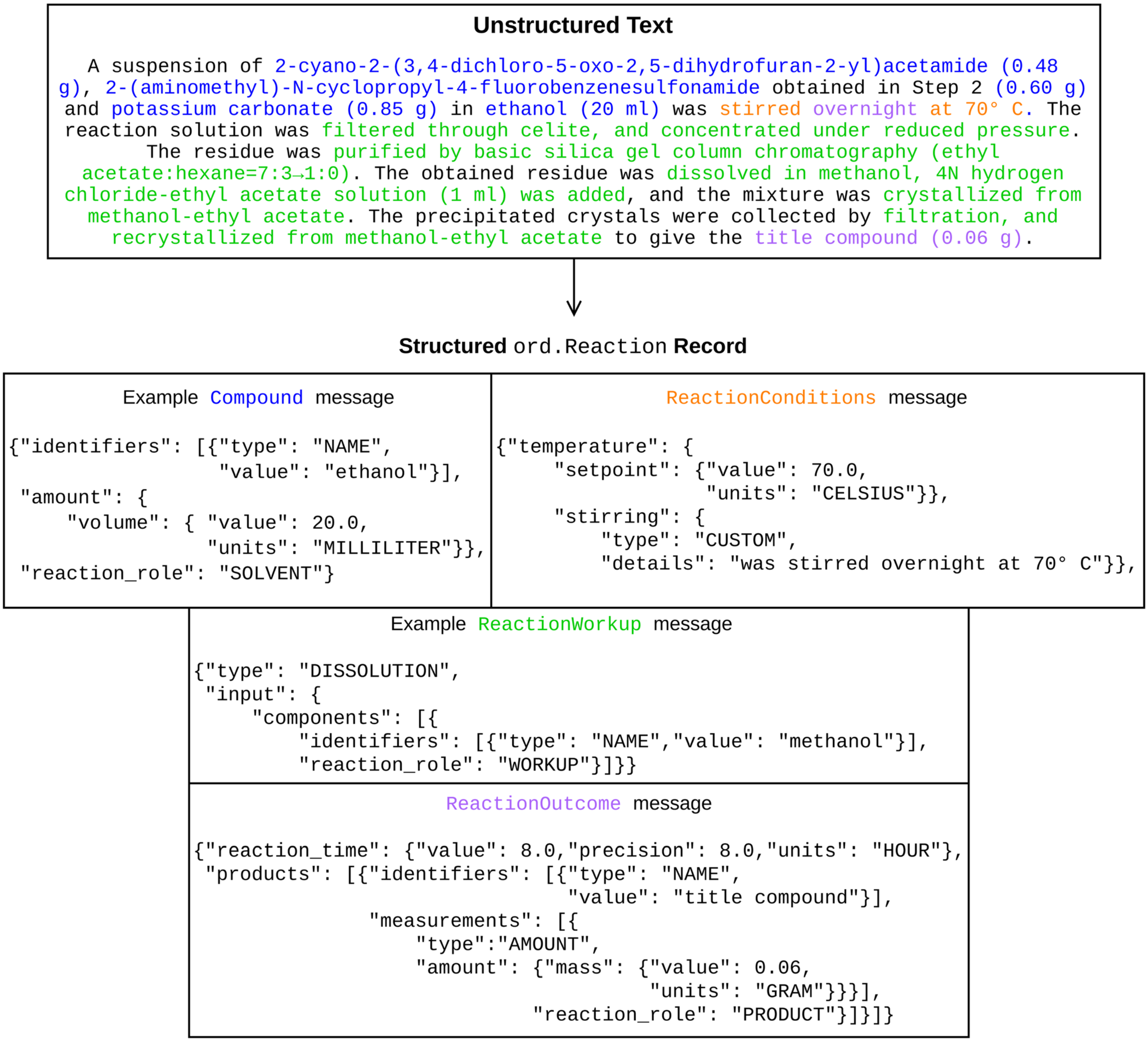

CML
uspto
Parser
PB
ORD
Deduplicate
Apply token limit
JSON
USPTO-ORD
100 K
I can convert unstructured text...
Isobutylamine (7.3 g, 0.1 mol) was dissolved ... mixture was cooled to -5° C... the product (4.1 g, 46%) ... filtered and dried ...
... to a structured ORD record:
inputs
conditions
Reaction
workups
outcomes
parameter-efficient fine-tuning

Fig. 1 Overview of this study's approach to structured reaction data extraction from text. A 100k reaction subset of the United States Patent and Trademark Office (USPTO) reaction data[39] as represented in the Open Reaction Database (ORD)[40] is used to fine-tune and evaluate LLaMa-2-7B. An example of the structured ORD record is included in Section 2.1. The data pipeline (top left) is detailed in Section 2.2. The fine-tuning procedure is described in Section 2.3. The llama with a cap was generated using Craiyon AI.[41]

2.2 Dataset preparation from patents and the ORD

Reaction records from the United States Patent and Trademark Office (USPTO) were collected from the ORD, sharded across 489 datasets. The link to a complete list of dataset IDs can be found in the ESI.† These records were originally published by Lowe in Chemical Markup Language (CML) format[39] and were imported into the ORD using a custom CML-to-ORD translation script.[45] A reaction record is admitted to our dataset if it satisfies the following conditions:

- Each of its ReactionInput messages has non-empty values for its components field. This usually means this reaction input is not the crude product of another reaction and that the chemical information of this reaction's inputs are present in reaction procedure text.



Unstructured Text

A suspension of 2-cyano-2-(3,4-dichloro-5-oxo-2,5-dihydrofuran-2-yl)acetamide (0.48 g), 2-(aminomethyl)-N-cyclopropyl-4-fluorobenzenesulfonamide obtained in Step 2 (0.60 g) and potassium carbonate (0.85 g) in ethanol (20 ml) was stirred overnight at 70° C. The reaction solution was filtered through celite, and concentrated under reduced pressure. The residue was purified by basic silica gel column chromatography (ethyl acetate:hexane=7:3→1:0). The obtained residue was dissolved in methanol, 4N hydrogen chloride-ethyl acetate solution (1 ml) was added, and the mixture was crystallized from methanol-ethyl acetate. The precipitated crystals were collected by filtration, and recrystallized from methanol-ethyl acetate to give the title compound (0.06 g).

Structured ord.Reaction **Record**

Example Compound message

```
{"identifiers": [{"type": "NAME",
                  "value": "ethanol"}],
 "amount": {
     "volume": { "value": 20.0,
                 "units": "MILLILITER"}},
 "reaction_role": "SOLVENT"}
```

ReactionConditions message

```
{"temperature": {
     "setpoint": {"value": 70.0,
                  "units": "CELSIUS"}},
     "stirring": {
         "type": "CUSTOM",
         "details": "was stirred overnight at 70° C"}},
```

Example ReactionWorkup message

```
{"type": "DISSOLUTION",
 "input": {
     "components": [{
         "identifiers": [{"type": "NAME","value": "methanol"}],
         "reaction_role": "WORKUP"}]}}
```

ReactionOutcome message

```
{"reaction_time": {"value": 8.0,"precision": 8.0,"units": "HOUR"},
 "products": [{"identifiers": [{"type": "NAME",
                                "value": "title compound"}],
               "measurements": [{
                   "type":"AMOUNT",
                   "amount": {"mass": {"value": 0.06,
                                       "units": "GRAM"}}}],
                   "reaction_role": "PRODUCT"}]}]}
```

Fig. 2 (Top) The original text description of a reaction procedure and (bottom) example messages within the structured ORD reaction record.[43]

- The reaction includes an associated procedure text, *i.e.*, the notes.procedure_details field of this reaction is a paragraph describing the reaction.

Reaction records satisfying these criteria were exported to JSON and deduplicated using OpenAI's data preparation tools (openai tools fine_tunes.prepare_data) to produce 1 339 260 unique records. The use of OpenAI's data preparation tools is free and was used here solely for convenient prompt deduplication. The procedure text and structured JSON are combined using a prompt template (see ESI†) modified from Stanford Alpaca.[46] A sequence length limit of 2048 tokens based on LLaMA tokenizer, is imposed due to memory considerations in fine-tuning the language models. This sequence limit reduces the number of records to 1 300 613 (97.1%) of 1 339 260. The cumulative distribution function of sequence lengths is shown in Fig. S1.† A subset of 100K records, hereinafter referred to as USPTO-ORD-100K, is randomly selected from the 1 300 613 records. Unless otherwise specified, a random 8 : 1 : 1 train : validation : test split is applied to USPTO-ORD-100K to train/evaluate models throughout this study. This data pipeline is schematically shown in Fig. 1.

The information in a structured ORD record is not guaranteed to be a proper subset of its free text description, as some information in the structured ORD record is derived from elsewhere, and in this work denoted "implicit information". For example, the reaction roles of compounds are rarely stated in a reaction's text description. As another example, the text description may indicate a filtration step (mapping to a ReactionWorkup of type FILTRATION in its ORD record) but does not include "filter" or "filtration" explicitly, *e.g.*, "passing through celite". We consider this kind of implicit information learnable and therefore do not exclude them from ORD records. On the other hand, some implicit information is considered unlearnable and thus excluded from the ORD records. Specifically,

- Unspecified outcome: if the name of a product is present in the ORD record and is not explicitly stated in the reaction text, this name is removed from the ORD record. This could happen when the product name is defined only in the title of the corresponding patent and not mentioned explicitly in the procedure text. This can also happen for reactants when they are referred to by compound identifiers or generic names.
- Calculated yield: if the yield value of a product is present in the ORD record and its integer value is not explicitly stated in the reaction text, this value is removed from the ORD record. This can occur when the calculated yield is different from the yield reported in the procedure text.

2.3 LLaMA fine-tuning procedure

LLaMA is a collection of decoder-only models first released in February 2023 by Meta AI,[47] with an updated version LLaMA-2 (released in July 2023).[48] LLaMA models are convenient foundational models for scientific communities because they are pre-trained using publicly available data only, have parameter sizes ranging from 7 billion to 70 billion, and are distributed with both model weights and training code under an open-source license. We select LLaMA-2-7B in this study for fine-tuning due to memory considerations. We note the pretrain-finetune paradigm is not exclusive to LLaMA nor the decoder-only models, and other large language models are also amenable to this task. Further performance improvements are likely possible by adopting a different pretrained model.

To avoid tuning the entire 7 billion parameters in LLaMA-2-7B, we adopt LLaMA-Adapter in our fine-tuning procedure.[49] LLaMA-Adapter achieves parameter-efficient fine-tuning using learnable adaption prompts: for each of the topmost L transformer layers, a learnable prompt of length K is prepended to the (embedded) word tokens. This procedure reduces the total number of trainable parameters to $K \times L \times C$, where C is the token embedding dimensions, set to 4096 by default in LLaMA. Throughout this study, $K = 10$ and $L = 30$, giving 1.2 million trainable parameters that can fit in a GPU of 24 GB memory in half precision.

The train and validation datasets from the aforementioned random split are used for fine-tuning LLaMA-2-7B. The validation set is used to monitor the training process and to determine the number of training epochs with early stopping. Fine-tuning LLaMA-2-7B for 15 epochs with an initial learning rate of 7×10^{-5} was completed in approximately 70 hours using 2 NVIDIA RTX 4090 GPUs. In contrast, preparing the ORD datasets (in .pb.gz format) to obtain USPTO-ORD-100K took approximately 4 hours using our scripts with a 16-core 4.70 GHz CPU (Intel® i7-1260P). The average inference speed was roughly 37 token per second as estimated over 100 generations on one RTX 4090 GPU with batch size set to 1. This model is referred to as "the fine-tuned model" throughout this study. The hyperparameters for fine-tuning were not optimized.

2.4 Evaluation protocol and metrics

Text descriptions of reaction records from the test set of USPTO-ORD-100K are passed to the fine-tuned LLaMA-2-7B to generate structured data as text completions for model evaluation. Because a Reaction message consists of nested sub-messages (or "objects" in JSON terminology), such as Compound and ReactionWorkup, we can define evaluation tasks based on the comparison between the ground truth and LLM-inferred Reaction at the message level: Evaluation Metric 1. For a given message type, how many messages of this message type are accurately extracted or erroneously added, removed, or altered?

Fig. 3 shows an example of Evaluation Metric 1 when comparing two ReactionInput messages given the message type of Compound messages. To distinguish the three failure modes, we first define a distance function for the given message type based on DeepDistance,[50] an edit distance similar to Levenshtein distance designed for nested objects. When comparing two lists of messages (the shorter list is padded with empty messages such that two lists are of equal sizes), a bijective mapping between messages from two lists is found by minimizing the distance sum of all pairs, which is then used to identify the aforementioned failure modes.

Since a message always has a tree structure, we can also define evaluation tasks at the leaf level, where a leaf corresponds to an unstructured, literal field: Evaluation Metric 2. For a given message type, how many leaf fields of messages of this message type are accurately extracted or erroneously added, removed, or altered?

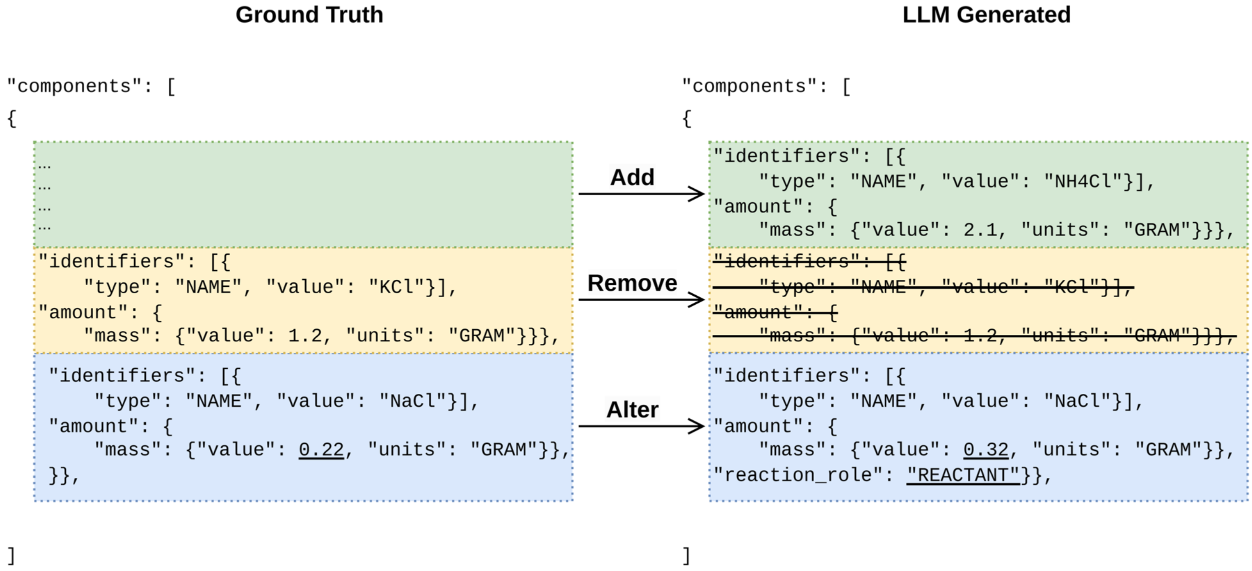
We note that Evaluation Metric 1 is defined at a lower granularity and is more stringent than Evaluation Metric 2, as summarized in Table 1. For example, in the case shown in Fig. 3, an entire compound message (blue) is marked as altered, while only two leaf fields (underscored) are considered as "Alteration" (value), and "Addition" (reaction_role), respectively. Assigning "Addition" and "Removal" to leaf fields also depends on the assignment at the message level, for example, when a message is assigned "Removal", all of its leaf fields are assigned "Removal".

It could be reasonable to use a numerical error measure to evaluate field-level extraction. This is because for certain downstream tasks, such as reaction condition recommendation, one could argue that mis-extracted fields containing floating point numbers will have a less deleterious effect on performance if they are close to the true value. However, we prefer the strict evaluation of exact-match accuracy for the information extraction task used here as sometimes missing or misplacing a number can happen more frequently than extracting a wrong number. This is reflected in an analysis on extracting reaction temperature values (ESI Section S7†).

3 Results and discussion

3.1 Quantitative model evaluation

The fine-tuned LLaMA-2-7B model is evaluated against the test set from the random 8 : 1 : 1 train–validation–test split of USPTO-ORD-100K. Out of the 10K model outputs (completions), only 42 (0.4%) of them are invalid JSON records, and 59 (0.6%) of them are invalid ORD records. Note the former is a sufficient condition for the latter. All of the 42 syntactically JSON invalid completions can be "repaired" by heuristic string operations,



```
"components": [
{
  …
  …
  …
  …
  "identifiers": [{
      "type": "NAME", "value": "KCl"}],
  "amount": {
      "mass": {"value": 1.2, "units": "GRAM"}}},
   "identifiers": [{
       "type": "NAME", "value": "NaCl"}],
   "amount": {
       "mass": {"value": 0.22, "units": "GRAM"}},
   }},

]
```

```
"components": [
{
  "identifiers": [{
      "type": "NAME", "value": "NH4Cl"}],
  "amount": {
      "mass": {"value": 2.1, "units": "GRAM"}}},
  ~~"identifiers": [{~~
      ~~"type": "NAME", "value": "KCl"}],~~
  ~~"amount": {~~
      ~~"mass": {"value": 1.2, "units": "GRAM"}}},~~
  "identifiers": [{
      "type": "NAME", "value": "NaCl"}],
  "amount": {
      "mass": {"value": 0.32, "units": "GRAM"}},
  "reaction_role": "REACTANT"}},

]
```

Fig. 3 An example of Evaluation Metric 1 for when comparing two lists of Compound messages, where the ground truth denotes the already structured JSON from the test set of USPTO-ORD-100K. Three failure modes at the Compound message level, "Addition", "Removal", and "Alteration" are colored green, yellow, and blue, respectively. Underscored fields denote failures at the leaf fields level (Evaluation Metric 2, *vide infra*). Data shown is for illustration purposes only.

Table 1 Comparison between Evaluation Metric 1 and 2

Metric	Specific to a message type	Specific to a field type	What is being counted?	Granularity
1	Yes	No	Added/removed/altered messages	Low
2	Yes	Yes	Added/removed/altered leaf fields	High

such as adding missing quotes or commas, using jsonrepair.[51] After repairing, 9963 (99.6%) valid ORD records are collected. These results indicate that the fine-tuned model successfully learns the syntax of the ORD's structured data schema during training.

Table 2 summarizes the evaluation results at the message level (Evaluation Metric 1). The fine-tuned model is able to extract compound information for ReactionInput entries reliably with an accuracy of 85.6%. Compared with missing compound information in ReactionInput (5.0%, failure mode "Removal"), it is relatively rare (2.3%) for the model to include excess compounds (failure mode "Addition"), and almost all of the excess compounds come from misplacement (*e.g.*, a ProductCompound is placed in ReactionInput) instead of hallucination.

Errors in extracting ProductCompound entries are more frequent, as indicated by a lower accuracy of 71.3%. Upon inspection, we noticed the errors mainly originate from implicit information: some fields of a ProductCompound message are not explicitly stated in the text description and are instead derived or inferred. One example is the "calculated" reaction yield, in contrast to the "reported" reaction yield which the model can capture successfully (Table S2†). To alleviate this effect, we also report the accuracy using a more lenient routine for identifying equivalent ProductCompound messages that considers two ProductCompound messages identical if all of their identifiers and amount fields are identical. These fields often capture all important chemical information about reaction outcomes. After applying this less strict equivalence definition, the accuracy for extracting ProductCompound messages increases from 71.3% to 87.1%, indicating that the model is capable of chemical entity/relation extraction even if it struggles with implicit calculation of yields. This routine also results in an increased accuracy (91.5%) for Compound messages in ReactionInput by excluding errors in reaction role classification (*vide infra*).

High accuracies of 95.7% and 90.7% are measured for ReactionConditions and ReactionWorkup, respectively. Since the ORD schema defines ReactionConditions as one single message rather than a list of messages, no "Addition" or "Removal" of this type of message is applicable.

To further understand how the fine-tuned model performs in extracting different types of chemical information, the completions are examined with finer granularity at the leaf level (Evaluation Metric 2), as shown in Table 3. The fine-tuned model shows excellent recognition capability for chemical entities such as compound identifiers (accuracy 93.5%) and amounts (95.2%), and it can infer reaction roles that are usually not explicitly stated in procedure texts (Section 3.3). Errors at

Table 2 Evaluation results at the message level (Evaluation Metric 1) for structured records extracted using the fine-tuned LLaMA-2-7B model. For each record in the test set of USPTO-ORD-100K, an ORD-formatted JSON record is extracted from the unstructured text and evaluated against the ground truth using Evaluation Metric 1. The "Path" column denotes the root path of the corresponding messages in a reaction message. * These values were calculated using a more lenient routine detailed in the main text

Message type	Path	Accurate	Removal	Addition	Alteration	Total
Compound	Inputs	38 470 (85.6%) 41 138* (91.5%)	2242 (5.0%)	1015 (2.3%)	4242 (9.4%) 1574* (3.5%)	44 954
ProductCompound	Outcomes	7450 (71.3%) 9105* (87.1%)	345 (3.3%)	58 (0.6%)	2656 (25.4%) 1001* (9.6%)	10 451
ReactionConditions	Conditions	9524 (95.7%)	N/A	N/A	433 (4.4%)	9957
ReactionWorkup	Workups	44 165 (90.7%)	1713 (3.5%)	1719 (3.5%)	2807 (5.8%)	48 685

Table 3 Evaluation results at the leaf field level (Evaluation Metric 2) for structured records extracted using the fine-tuned LLaMA-7B model. For each record in the test set of USPTO-ORD-100K, an ORD-formatted JSON record is extracted from the unstructured text and evaluated against the ground truth using Evaluation Metric 2. * These fields do not belong to any of the five field types (identifiers, amount, reaction role, condition, workup). In this dataset, all of them are leaf fields of ProductCompound, including texture, isolated_color, and yield-related measurements

Message type	Field type	Accurate	Removal	Addition	Alteration	Total
ProductCompound & Compound	Identifiers	100 958 (93.5%)	5490 (5.1%)	2590 (2.4%)	1566 (1.5%)	108 014
	Amount	74 209 (95.2%)	3434 (4.4%)	2182 (2.8%)	300 (0.4%)	77 943
	Reaction role	48 262 (89.3%)	2797 (5.2%)	1264 (2.3%)	2978 (5.5%)	54 037
ReactionConditions	Condition	26 782 (98.3%)	298 (1.1%)	391 (1.4%)	176 (0.7%)	27 256
ReactionWorkup	Workup	178 733 (94.0%)	8360 (4.4%)	10 189 (5.4%)	3156 (1.7%)	190 249
Other*		31 794 (84.80%)	5261 (14.0%)	2240 (6.0%)	439 (1.2%)	37 494

the field-level mainly come from implicit information in ProductCompound messages, such as calculated yields (Table S1†).

As an alternate approach and point of comparison, we explored extracting structured data with pretrained LLMs directly using the chain-of-thought prompting method,[52] a few-shot training method by engineering the prompts such that they mimic the thought processes of a human when solving a complicated task. This method is easier to deploy compared to the fine-tuning methods; however, it could only produce syntactically correct ORD data in 408 out of 500 cases after repairing with accuracies of 61.2% and 31.3% for Compound and ProductCompound, respectively, indicating that chain-of-thought prompting without fine-tuning is likely insufficient for this task. This prompting method is also limited by human-crafted instructions and the context window of the model, and, considering there are more than 600 different fields defined in ORD schema, preparing examples and steps to extract a full Reaction record seems impractical. Enabling JSON mode through OpenAI API in this process does not improve the model performance (Table S4†). Details of our implementation and evaluation can be found in ESI.†

3.2 Comparison to previous studies

As a smart chemical NER tool, the fine-tuned model learned to recognize cross-referencing tokens and to ignore unwanted chemical entities. This is reflected in the comparison (Table 4) between the fine-tuned model and ChemDataExtractor (version 2.1.0),[53,54] a toolkit for extracting chemical information mainly from scientific literature. Specifically, the comparison is made for the task of compound name recognition, which evaluates the list of compounds (entities) extracted from a reaction. This list is directly available both from the output of ChemDataExtractor and from the ORD-formatted structured data from the fine-tuned model. While ChemDataExtractor is capable of recognizing many chemical entities, it frequently fails to identify referencing tokens, such as "desired product" or "compound 322" (the "Removal" column). It also captures excess chemical entities, such as "1H" from NMR reports ("the "Addition" column). These errors are at least partially attributable to the distribution shift in how procedures are described in our source text paragraphs. We also compare our fined-tuned model with a NER model based on a pre-trained BERT model, MatSciBERT.[55] This NER model is trained and evaluated using the same USPTO-ORD-100K dataset and is marginally better than the fine-tuned LLaMA model. Considering the significantly lower training cost of the BERT model compared to fine-tuning the LLaMa model (∼10×), the former may be preferable for pure NER tasks.

Table 4 Compound name recognition results of the fine-tuned model, ChemDataExtractor, and the MatSciBert model from the test set of USPTO-ORD-100K. In this task, a set of compound names (entities) is extracted from the unstructured text and is then evaluated against the ground truth

Model	Accurate	Removal	Addition	Alteration	Total
Fine-tuned	94.9%	4.1%	2.2%	1.0%	78 408
ChemDataExtractor	76.1%	16.0%	22.7%	8.0%	
MatSciBert	96.6%	2.2%	2.4%	1.2%	

We further test the fine-tuned model on uniproduct reactions from the ChemRxnExtractor[16] dataset, a set of 123 records with labeled tokens for compound names. All records from this dataset were collected from individual literature passages. These passages can be considered an out-of-distribution challenge to our fine-tuned model: they tend to be defined by general chemical transformations (*e.g.*, "oxidation of A gave B" or "cyclization of A afforded B") instead of specific actions in synthesis procedures, chemical amount information is rarely present, and named entities in these passages are frequently represented by externally referencing tokens. As expected, the fine-tuned model performs poorly on this dataset, with an accuracy of 62.6% and a tendency to include unwanted tokens (Table S1†). Such a tendency often results from prioritizing chemical entities above referencing tokens. For example, in "by heating tryptophan methyl ester (9) at 140 °C for 3 h" the token "9" is the correct token to extract, while the fine-tuned model only recognizes "tryptophan methyl ester" which is a chemical entity in a more general sense. These results suggest the ChemRxnExtractor dataset differs significantly from USPTO-ORD-100K, which justifies fine-tuning the base LLaMA-2-7B model for the ChemRxnExtractor dataset. Unfortunately, the small size of the ChemRxnExtractor dataset makes it insufficient for fine-tuning and subsequent evaluation (ESI Section S2†).

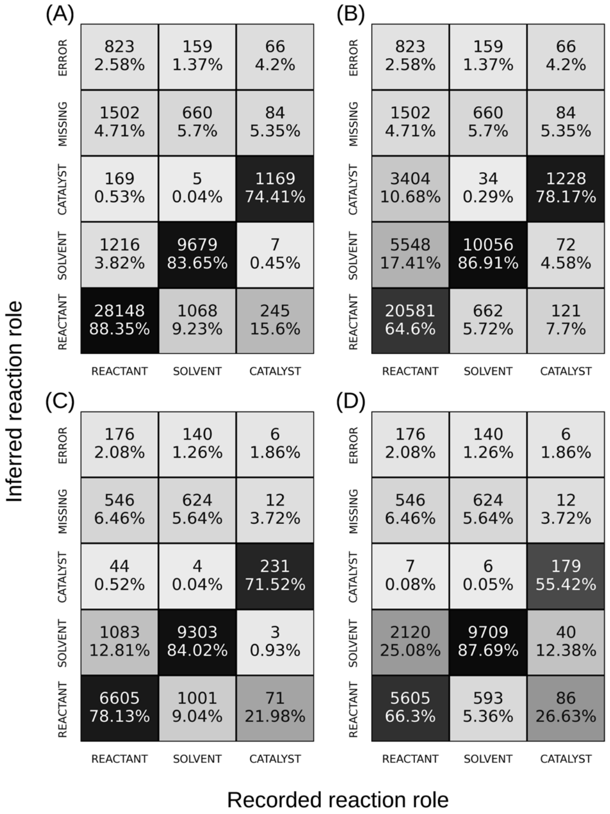


Fig. 4 Confusion matrices of reaction role classification for the compounds in the test dataset using (A) the fine-tuned model and (B) the popularity baseline. The results for compounds whose role in the dataset varies from reaction to reaction are shown for (C) the fine-tuned model and (D) the baseline model. Percentage values were normalized using the number of true instances. In addition to three reaction role classes, prediction results can also be labeled as "MISSING" – when the corresponding compound is absent in the extracted ORD record, and "ERROR" – when the name of the extracted compound is incorrect. Note that because the reaction role classification depends on correct extraction of compound names, the first two rows of Fig. 4A and B share identical values. The same applies to the first two rows of Fig. 4C and D.

3.3 Reaction role classification

Reaction role assignments that distinguish reactants, reagents, catalysts, and solvents are sometimes used in downstream tasks such as reaction condition recommendation.[56–58] The reaction role of a compound is context-dependent, *e.g.*, a chemical can serve as a solvent or a reactant in different reactions, and not explicitly stated in procedure text, so this is also not a pure information extraction task. However, since this implicit information is included in fine-tuning, the fine-tuned model learns the conventions about role assignment in a generalizable way, and the inferred assignment is directly available in the reaction_role field. Since each Compound message is allowed to have only one reaction_role, the reaction role assignment is a standard classification problem. While the ORD data schema has more than 10 types of reaction roles defined to cover a variety of situations, in this dataset only three are used for input compounds (CATALYST, REACTANT, SOLVENT). The prediction of one of these three labels for each compound defines the reaction role classification problem (and corresponding definitions of accuracy) discussed in this section We exclude ProductCompound messages in this section because they always have a reaction_role of PRODUCT in this dataset. We also evaluate a popularity baseline that makes classification decisions based on the role frequency of compounds in the training dataset; roles are uniformly randomly assigned in the case of ties or unseen compounds.

Fig. 4A shows the confusion matrix of reaction role assignment from the fine-tuned model for all compounds in ReactionInput from the test dataset. The classification accuracy decreases from REACTANT to SOLVENT to CATALYST, with a tendency to mislabel SOLVENT or CATALYST as REACTANT, as expected based on class populations. Compared to extracting compounds of other roles (2.6% for REACTANT, 1.4% for SOLVENT), the model failed more frequently (4.2%) when extracting catalysts. Fig. 4B shows the results from the popularity baseline with similar accuracies for SOLVENT and CATALYST, and lower accuracy for REACTANT compared to the fine-tuned model. A macro-average F1 score of 86.1% is calculated for the fine-tuned model, while the popularity baseline gives 63.5%. For compounds whose reaction role in the dataset varies from reaction to reaction, the difference between the fine-tuned model (Fig. 4C) and the popularity baseline (Fig. 4D) becomes more pronounced: the former exhibits better performance for both REACTANT and CATALYST. These results suggest through fine-tuning the model learned to make role classifications based on reaction context.

4 Conclusion

We have demonstrated the application of a fine-tuned LLaMA model for the extraction of structured reaction information from unstructured reaction texts from the USPTO. The fine-tuned model can consistently (99.6%) produce JSON records complying with the highly structured ORD data schema. The fine-tuned model exhibits average accuracies of 91.3% for message level, and 92.3% for field-level extractions. The fine-tuned model can also infer reaction roles that are not explicitly stated in texts, modestly beating the popularity baseline for role classification. While the model may not be accurate enough to be directly used in dataset preparation, it may greatly accelerate information extraction compared to manual extraction, and simplify the job of human curators, especially for detailed, nested data schemas.

As reaction data can include additional non-textual elements, such as reaction schemes and tables for reporting conditions/yields, multi-modality models will be needed to fully organize unstructured data. For reaction schemes, recent developments in the field of optical chemical structure recognition have enabled open-source tools to accurately capture chemical entities from raster images. Notable examples include MolScribe[59] and RxnScribe[60] developed by Barzilay and coworkers, as well as ReactionDataExtractor[61,62] by Wilary and Cole. Table parsing/extraction tools have also been developed for chemistry literature, such as the table parsing module in ChemDataExtractor[54] and OpticalTable-SQA,[63] a fine-tuned question-answering language model for table extraction. As multimodal foundation models become increasingly available in fields beyond chemistry, it will be worth exploring their suitability for reaction data extraction.

The obvious use of the fine-tuned model is to support reaction data import to ORD with proper expert validation of the LLM-generated output. For example, as a postprocessing tool to convert unstructured ELN reports to structured data, or a reviewing/proofreading tool to expose as structured data what would otherwise be unsearchable, such as the procedure details buried in supplementary materials of a journal article. Tools presented in this study should contribute to answering the call for standardization in reaction informatics.[1,64] As aligning reaction text with molecular representation has been demonstrated to be helpful in prediction tasks, the tool developed in this study could also serve as an auxiliary to inform reaction predictive models.[65]

Data availability

The source code for data processing, the fine-tuning and evaluation scripts, and all fine-tuning/evaluation datasets used in this study can be found at **https://github.com/qai222/LLM_organic_synthesis**. The fine-tuned model is available at **https://doi.org/10.6084/m9.figshare.25485973**.

Author contributions

Qianxiang Ai: conceptualization, data curation, formal analysis, investigation, methodology, software, writing – original draft preparation. Fanwang Meng: data curation, formal analysis. Jiale Shi: conceptualization, methodology, software. Brenden Pelkie: data curation, formal analysis. Connor W. Coley: funding acquisition, supervision, writing – review & editing.

Conflicts of interest

There are no conflicts of interest to declare.

Acknowledgements

Research reported in this publication was supported by the National Institutes of Health under award number U18TR004149 and by the U.S. National Science Foundation through the UW Molecular Engineering Materials Center (MEM-C), a Materials Research Science and Engineering Center (DMR-2308979). The content is solely the responsibility of the authors and does not necessarily represent the official views of the National Institutes of Health or the National Science Foundation. The authors thank Dr Ben Blaiszik for seed funding. The authors also thank Zhengkai Tu, Dr Hassan Harb, Dr Jacob No. Sanders, Dr Stefan Bringuier, and Marcus Schwarting for helpful discussions.

Notes and references

1 R. Mercado, S. M. Kearnes and C. W. Coley, *J. Chem. Inf. Model.*, 2023, **63**, 4253–4265.
2 S. W. Gabrielson, *J. Med. Libr. Assoc.*, 2018, **106**, 588–590.
3 A. J. Lawson, J. Swienty-Busch, T. Géoui and D. Evans, in *The Future of the History of Chemical Information, American Chemical Society*, ACS Symposium Series, 2014, vol. 1164, pp. 127–148.
4 M. Krallinger, O. Rabal, F. Leitner, M. Vazquez, D. Salgado, Z. Lu, R. Leaman, Y. Lu, D. Ji, D. M. Lowe, R. A. Sayle, R. T. Batista-Navarro, R. Rak, T. Huber, T. Rocktäschel, S. Matos, D. Campos, B. Tang, H. Xu, T. Munkhdalai, K. H. Ryu, S. Ramanan, S. Nathan, S. Žitnik, M. Bajec, L. Weber, M. Irmer, S. A. Akhondi, J. A. Kors, S. Xu, X. An, U. K. Sikdar, A. Ekbal, M. Yoshioka, T. M. Dieb, M. Choi, K. Verspoor, M. Khabsa, C. L. Giles, H. Liu, K. E. Ravikumar, A. Lamurias, F. M. Couto, H.-J. Dai, R. T.-H. Tsai, C. Ata, T. Can, A. Usié, R. Alves, I. Segura-Bedmar, P. Martínez, J. Oyarzabal and A. Valencia, *J. Cheminf.*, 2015, **7**, S2.
5 M. Krallinger, O. Rabal, A. Lourenço, J. Oyarzabal and A. Valencia, *Chem. Rev.*, 2017, **117**, 7673–7761.
6 D. M. Lowe and R. A. Sayle, *J. Cheminf.*, 2015, **7**, S5.
7 L. Hawizy, D. M. Jessop, N. Adams and P. Murray-Rust, *J. Cheminf.*, 2011, **3**, 17.
8 G. Papadatos, M. Davies, N. Dedman, J. Chambers, A. Gaulton, J. Siddle, R. Koks, S. A. Irvine, J. Pettersson, N. Goncharoff, A. Hersey and J. P. Overington, *Nucleic Acids Res.*, 2016, **44**, D1220–D1228.
9 NextMove Software|Pistachio, **https://www.nextmovesoftware.com/pistachio.html**.
10 E. Pan, S. Kwon, Z. Jensen, M. Xie, R. Gómez-Bombarelli, M. Moliner, Y. Román-Leshkov and E. Olivetti, *ACS Cent. Sci.*, 2024, 729–743.

11 J. Lafferty, A. McCallum, F. Pereira, *et al.*, *Icml*, 2001, 3.

12 T. Rocktäschel, M. Weidlich and U. Leser, *Bioinformatics*, 2012, **28**, 1633–1640.

13 L. Luo, Z. Yang, P. Yang, Y. Zhang, L. Wang, H. Lin and J. Wang, *Bioinformatics*, 2018, **34**, 1381–1388.

14 W. Hemati and A. Mehler, *J. Cheminf.*, 2019, **11**, 3.

15 Z. Zhai, D. Q. Nguyen, S. Akhondi, C. Thorne, C. Druckenbrodt, T. Cohn, M. Gregory and K. Verspoor, *Proceedings of the 18th BioNLP Workshop and Shared Task*, Florence, Italy, 2019, pp. 328–338.

16 J. Guo, A. S. Ibanez-Lopez, H. Gao, V. Quach, C. W. Coley, K. F. Jensen and R. Barzilay, *J. Chem. Inf. Model.*, 2022, **62**, 2035–2045.

17 T. Isazawa and J. M. Cole, *J. Chem. Inf. Model.*, 2022, **62**, 1207–1213.

18 T. Almeida, R. Antunes, J. F. Silva, J. R. Almeida and S. Matos, *Database*, 2022, **2022**, baac047.

19 A. Trewartha, N. Walker, H. Huo, S. Lee, K. Cruse, J. Dagdelen, A. Dunn, K. A. Persson, G. Ceder and A. Jain, *Patterns*, 2022, **3**, 100488.

20 R. Hoffmann, C. Zhang, X. Ling, L. Zettlemoyer and D. S. Weld, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, Portland, Oregon, USA, 2011, pp. 541–550.

21 S. Riedel, L. Yao and A. McCallum, *Machine Learning and Knowledge Discovery in Databases*, Berlin, Heidelberg, 2010, pp. 148–163.

22 X. Zeng, D. Zeng, S. He, K. Liu and J. Zhao, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, volume 1: Long Papers*, Melbourne, Australia, 2018, pp. 506–514.

23 M. Miwa and M. Bansal, End-to-End Relation Extraction using LSTMs on Sequences and Tree Structures, in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Berlin, Germany, 2016, pp. 1105–1116, DOI: 10.18653/v1/P16-1105.

24 P.-L. Huguet Cabot and R. Navigli, *Findings of the Association for Computational Linguistics: EMNLP 2021*, Punta Cana, Dominican Republic, 2021, pp. 2370–2381.

25 M. Eberts and A. Ulges, *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, 2021, pp. 3650–3660.

26 R. Luo, L. Sun, Y. Xia, T. Qin, S. Zhang, H. Poon and T.-Y. Liu, *Briefings Bioinf.*, 2022, **23**, bbac409.

27 M. Ansari and S. M. Moosavi, Agent-based Learning of Materials Datasets from Scientific Literature, *arXiv*, 2023, preprint, arXiv:2312.11690 [cs], http://arxiv.org/abs/2312.11690.

28 S. Datta, K. Lee, H. Paek, F. J. Manion, N. Ofoegbu, J. Du, Y. Li, L.-C. Huang, J. Wang, B. Lin, H. Xu and X. Wang, *J. Am. Med. Inform. Assoc.*, 2024, **31**, 375–385.

29 J. Dagdelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson and A. Jain, *Nat. Commun.*, 2024, **15**, 1418.

30 N. Walker, S. Lee, J. Dagdelen, K. Cruse, S. Gleason, A. Dunn, G. Ceder, A. Paul Alivisatos, K. A. Persson and A. Jain, *Digital Discovery*, 2023, **2**, 1768–1782.

31 M. Zhong, S. Ouyang, M. Jiang, V. Hu, Y. Jiao, X. Wang and J. Han, *Findings of the Association for Computational Linguistics: ACL 2023*, Toronto, Canada, 2023, pp. 12120–12130.

32 M. Zhong, S. Ouyang, Y. Jiao, P. Kargupta, L. Luo, Y. Shen, B. Zhou, X. Zhong, X. Liu, H. Li, J. Xiao, M. Jiang, V. Hu, X. Wang, H. Ji, M. Burke, H. Zhao and J. Han, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, Singapore, 2023, pp. 389–402.

33 L. Patiny and G. Godin, Automatic extraction of FAIR data from publications using LLM, *ChemRxiv*, 2023, preprint, DOI: 10.26434/chemrxiv-2023-05v1b-v2.

34 A. C. Vaucher, F. Zipoli, J. Geluykens, V. H. Nair, P. Schwaller and T. Laino, *Nat. Commun.*, 2020, **11**, 3601.

35 S. H. M. Mehr, M. Craven, A. I. Leonov, G. Keenan and L. Cronin, *Science*, 2020, **370**, 101–108.

36 J. He, D. Q. Nguyen, S. A. Akhondi, C. Druckenbrodt, C. Thorne, R. Hoessel, Z. Afzal, Z. Zhai, B. Fang and H. Yoshikawa, *Proceedings of the CLEF 2020 conference*, 2020.

37 Y. Li, B. Fang, J. He, H. Yoshikawa, S. A. Akhondi, C. Druckenbrodt, C. Thorne, Z. Afzal, Z. Zhai and T. Baldwin, *CLEF (Working Notes)*, 2021, 693–709.

38 Y. Li, B. Fang, J. He, H. Yoshikawa, S. A. Akhondi, C. Druckenbrodt, C. Thorne, Z. Afzal, Z. Zhai and K. Machi, *CLEF (Working Notes)*, 2022, pp. 758–781.

39 D. Lowe, Chemical reactions from US patents (1976-Sep2016), 2017, https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873.

40 S. M. Kearnes, M. R. Maser, M. Wleklinski, A. Kast, A. G. Doyle, S. D. Dreher, J. M. Hawkins, K. F. Jensen and C. W. Coley, *J. Am. Chem. Soc.*, 2021, **143**, 18820–18826.

41 Craiyon, A llama with a square academic cap, https://www.craiyon.com/.

42 K. M. Jablonka, Q. Ai, A. Al-Feghali, S. Badhwar, J. D. Bocarsly, A. M. Bran, S. Bringuier, L. C. Brinson, K. Choudhary, D. Circi, S. Cox, W. A. d. Jong, M. L. Evans, N. Gastellu, J. Genzling, M. V. Gil, A. K. Gupta, Z. Hong, A. Imran, S. Kruschwitz, A. Labarre, J. Lála, T. Liu, S. Ma, S. Majumdar, G. W. Merz, N. Moitessier, E. Moubarak, B. Mouriño, B. Pelkie, M. Pieler, M. C. Ramos, B. Ranković, S. G. Rodriques, J. N. Sanders, P. Schwaller, M. Schwarting, J. Shi, B. Smit, B. E. Smith, J. V. Herck, C. Völker, L. Ward, S. Warren, B. Weiser, S. Zhang, X. Zhang, G. A. Zia, A. Scourtas, K. J. Schmidt, I. Foster, A. D. White and B. Blaiszik, *Digital Discovery*, 2023, **2**, 1233–1250.

43 R. Hammond Jr, Apparatus detachably attachable to fishing poles for holding and dispensing semi-liquids, 1993, https://patents.google.com/patent/US5242088A/en?oq=US07985863B2A.

44 Open Reaction Database, ord-schema, https://github.com/open-reaction-database/ord-schema/blob/

ec1ac7965e79e0165ecc3549af7ee8a31c2725a0/proto/reaction.proto.

45 S. Kearnes, CML to ORD parser, https://github.com/open-reaction-database/ord-schema/blob/81ff0943538364722c4ca82d66b24c4361644b56/ord_schema/scripts/parse_uspto.py.

46 R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang and T. B. Hashimoto, Stanford Alpaca: An Instruction-following LLaMA model, Publication Title: GitHub repository, 2023, https://github.com/tatsu-lab/stanford_alpaca.

47 H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, A. Rodriguez, A. Joulin, E. Grave and G. Lample, LLaMA: Open and Efficient Foundation Language Models, *arXiv*, 2023, preprint, arXiv:2302.13971 [cs], http://arxiv.org/abs/2302.13971.

48 H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov and T. Scialom, Llama 2: Open Foundation and Fine-Tuned Chat Models, *arXiv*, 2023, preprint, arXiv:2307.09288 [cs], http://arxiv.org/abs/2307.09288.

49 R. Zhang, J. Han, C. Liu, P. Gao, A. Zhou, X. Hu, S. Yan, P. Lu, H. Li and Y. Qiao, LLaMA-Adapter: Efficient Fine-tuning of Language Models with Zero-init Attention, *arXiv*, 2023, preprint, arXiv:2303.16199 [cs], http://arxiv.org/abs/2303.16199.

50 S. Dehpour, seperman/deepdiff, 2024, https://github.com/seperman/deepdiff, original-date: 2014-09-26T03:21:47Z.

51 J. d. Jong, josdejong/jsonrepair, 2024, https://github.com/josdejong/jsonrepair, original-date: 2020-11-02T16:05:02Z.

52 J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le and D. Zhou, Chain-of-Thought Prompting Elicits Reasoning in Large Language Models, *arXiv*, 2023, preprint, arXiv:2201.11903 [cs], http://arxiv.org/abs/2201.11903.

53 M. C. Swain and J. M. Cole, *J. Chem. Inf. Model.*, 2016, **56**, 1894–1904.

54 J. Mavračić, C. J. Court, T. Isazawa, S. R. Elliott and J. M. Cole, *J. Chem. Inf. Model.*, 2021, **61**, 4280–4289.

55 T. Gupta, M. Zaki, N. A. Krishnan and Mausam, *npj Comput. Mater.*, 2022, **8**, 102.

56 H. Gao, T. J. Struble, C. W. Coley, Y. Wang, W. H. Green and K. F. Jensen, *ACS Cent. Sci.*, 2018, **4**, 1465–1476.

57 A. M. Żurański, J. I. Martinez Alvarado, B. J. Shields and A. G. Doyle, *Acc. Chem. Res.*, 2021, **54**, 1856–1865.

58 V. Voinarovska, M. Kabeshov, D. Dudenko, S. Genheden and I. V. Tetko, *J. Chem. Inf. Model.*, 2024, **64**, 42–56.

59 Y. Qian, J. Guo, Z. Tu, Z. Li, C. W. Coley and R. Barzilay, *J. Chem. Inf. Model.*, 2023, **63**, 1925–1934.

60 Y. Qian, J. Guo, Z. Tu, C. W. Coley and R. Barzilay, *J. Chem. Inf. Model.*, 2023, **63**, 4030–4041.

61 D. M. Wilary and J. M. Cole, *J. Chem. Inf. Model.*, 2021, **61**, 4962–4974.

62 D. M. Wilary and J. M. Cole, *J. Chem. Inf. Model.*, 2023, **63**, 6053–6067.

63 J. Zhao, S. Huang and J. M. Cole, *J. Chem. Inf. Model.*, 2023, **63**, 1961–1981.

64 P. Baldi, *J. Chem. Inf. Model.*, 2022, **62**, 2011–2014.

65 Y. Qian, Z. Li, Z. Tu, C. Coley and R. Barzilay, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore, 2023, pp. 12731–12745.