Digital Discovery

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2893

Extraction of chemical synthesis information using the World Avatar

Simon D. Rihm, †[a] Fabio Saluz,†[ab] Aleksandar Kondinski, [ac] Jiaru Bai, [a] Patrick W. V. Butler, [a] Sebastian Mosbach,[ad] Jethro Akroyd [ad] and Markus Kraft *[ade]

Received 5th May 2025
Accepted 4th August 2025

DOI: 10.1039/d5dd00183h

rsc.li/digitaldiscovery

This work presents a generalisable process that transforms unstructured synthesis descriptions of metal–organic polyhedra (MOPs) – a class of organometallic nanocages – into machine-readable, structured representations, integrating them into The World Avatar (TWA), a universal knowledge representation encompassing physical, abstract, and conceptual entities. TWA makes use of knowledge graphs and semantic agents. While previous work established rational design principles for MOPs in the context of TWA, experimental verification remains a bottleneck due to the lack of accessible and structured synthesis data. However, synthesis information in the literature is often sparse, ambiguous, and embedded with implicit knowledge, making direct translation into structured formats a significant challenge. To achieve this, a synthesis ontology was developed to standardise the representation of chemical synthesis procedures by building on existing standardisation efforts. We then designed an LLM-based pipeline with advanced prompt engineering strategies to automate data extraction and created workflows for seamless integration into a knowledge representation within TWA. Using this approach, we extracted and uploaded nearly 300 synthesis procedures, automatically linking reactants, chemical building units, and MOPs to related entities across interconnected knowledge graphs. Over 90% of publications were processed successfully through the fully automated pipeline without manual intervention. The demonstrated use cases show that this framework supports chemists in designing and executing experiments and enables data-driven retrosynthetic analysis, laying the groundwork for autonomous, knowledge-guided discovery in reticular chemistry.

1 Introduction

Metal–organic polyhedra (MOPs) represent an intriguing class of materials owing to their distinctive structural and chemical characteristics.[1–4] MOPs are porous, highly ordered structures incorporating metallic or multi-metallic centres, whose properties can be precisely tailored for specific applications such as gas separation and catalysis.[4,5] These functionalities align with increasing demands for materials that address global challenges such as greenhouse gas mitigation, with MOPs showing promise in carbon capture[6] and utilisation.[7] The symmetrical polyhedral structures and well-defined pore sizes of MOPS enable a modular approach to material design, wherein complex materials are systematically constructed from simpler, clearly defined building blocks.[8] The self-assembly process of MOPs is driven by the coordination preferences of metal ions and influenced significantly by synthesis conditions; however, current synthesis practices still rely largely on empirical, trial-and-error methods.[1,5,9]

Our recent work has shown that algorithms leveraging explicit knowledge representation can significantly facilitate the design and prediction of novel materials. Kondinski *et al.*[10] introduced a geometric assembly model for MOPs, enabling the systematic prediction of over 1400 previously undocumented structures. This algorithm and its associated data form part of The World Avatar (TWA), a platform supporting semantic representation and interactions between data and computational agents. Additionally, this approach allows for preliminary estimation of the geometric structures and derived properties of these predicted MOPs, such as pore and cavity dimensions.[11]

Although knowledge-based models have successfully predicted novel MOP structures, their experimental validation remains a critical bottleneck, highlighting the necessity for

[a] *Department of Chemical Engineering and Biotechnology, University of Cambridge, Philippa Fawcett Drive, Cambridge, CB3 0AS, UK. E-mail: mk306@cam.ac.uk*
[b] *Department of Mechanical and Process Engineering, ETH Zurich, Rämistrasse 101, Zurich, CH-8092, Switzerland*
[c] *Institute of Physical and Theoretical Chemistry, Graz University of Technology, Stremayrgasse 9, Graz, 8010, Austria*
[d] *CARES, Cambridge Centre for Advanced Research and Education in Singapore, 1 Create Way, CREATE Tower, #05-05, 138602, Singapore*
[e] *Department of Chemical Engineering, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Room 66-350, Cambridge, Massachusetts, 02139, US*

† These authors contributed equally to this work.

more efficient and systematic approaches to synthesis planning and execution.[10] Despite their potential, MOP syntheses currently face substantial challenges. Traditional synthesis approaches typically rely on trial-and-error methods, which are both time-consuming and resource-intensive.[1] Moreover, synthesis procedures reported in the literature are often presented in unstructured formats, complicating standardisation and automation. The absence of structured, machine-readable data significantly restricts the integration of advanced computational tools and limits the scalability of MOP research.[12]

Large language models (LLMs) have shown significant promise in extracting detailed synthesis information from scientific literature. For example, Zhu *et al.*[13] demonstrated an "AI chemist" capable of inferring novel synthesis routes. LLMs and prompt engineering methods have evolved substantially, progressing from basic response-guidance techniques to sophisticated strategies that optimise model performance. Specifically, in-context learning (ICL) enhances outcomes by strategically embedding examples. Techniques such as zero-shot and few-shot prompting enable complex tasks with minimal training examples.[14–16] Furthermore, role prompting and chain-of-thought (CoT) prompting improve logical reasoning through step-by-step guidance.[17,18] Finally, retrieval-augmented generation (RAG) integrates external knowledge, enriching model outputs, while schema-aligned prompting ensures outputs conform precisely to structured data formats.[19]

Despite notable progress in applying LLM-driven techniques for chemical data extraction[20,21] and even efforts to construct structured knowledge representations such as knowledge graphs (KGs) from extracted information,[22,23] the integration of such data into comprehensive, existing knowledge ecosystems comprising multiple interconnected KGs, such as TWA, remains largely unexplored. This underutilisation restricts opportunities to combine LLM-based information extraction with semantic structuring capabilities inherent in KGs and TWA, potentially limiting significant improvements in the accessibility, interoperability, and automation of chemical knowledge. Increasing the adoption of these advanced methodologies could bridge this gap, enabling streamlined workflows that convert unstructured scientific literature into structured, actionable data, seamlessly integrated within dynamic knowledge systems.[24]

The purpose of this paper is to build on these advances by developing an integrated family of computational agents that not only extract synthesis information from scientific literature by utilising LLMs but also embed this information within the semantic framework of TWA, as introduced by Kondinski *et al.*[10] Our focus is on integrating synthesis data for MOPs into TWA to augment knowledge of experimentally known structures and enable automated generation of synthesis routes for newly predicted MOPs. This addresses three critical challenges: automating the extraction of chemical knowledge from unstructured texts, embedding it into a pre-existing knowledge base, and establishing a semantic framework that enables computational agents to process, interpret, and propose novel synthesis pathways. Aligning this ontology with established frameworks further enhances interoperability.

2 Background

This section introduces background on three key areas. First, we review the rational design of MOPs, focusing on chemical and geometric principles for structure prediction, and introduce The World Avatar. Second, we discuss current challenges in documenting synthesis procedures and present standardisation frameworks such as XDL, CML, and SiLA. Third, we examine how large language models support information extraction from scientific literature, highlighting advanced prompt-engineering techniques, including in-context learning, chain-of-thought prompting, retrieval-augmented generation, and structured output generation.

2.1 Rational metal–organic polyhedra design in the World Avatar

MOPs are hybrid nanomolecules composed of repeating organic and inorganic units, forming highly symmetrical, supramolecular cage-like structures.[8] Their intrinsic porosity and internal cavities enable applications in molecular sensing, carbon capture, and synthesis of metal organic frameworks (MOFs).[1,10] In addition, owing to their discrete and well-defined architectures, MOPs exhibit extensive internal and external functionalisation, enhancing their adaptability for biomedical applications, catalysis, and gas separation.[5]

Given the vast number of potential MOP structures, a systematic design approach is essential.[4] Kondinski *et al.*[10] designed MOPs by leveraging chemical and geometric principles to systematically predict novel structures. This work was conducted within The World Avatar, a dynamic collection of virtual knowledge graphs and semantic agents that enables seamless cross-domain data integration and automated knowledge discovery.[24–26] TWA employs a containerised technology stack, including Blazegraph and a Python-based twa library.[27,28]

In a previous work, MOPs were designed using geometric and chemical rules.[10] The approach hinges on chemical complementarity, ensuring stable bonds between organic and inorganic components, and topological compatibility, which governs spatial arrangement. This was implemented *via* a framework (for details, see SI) consisting of chemical building units (CBUs) and their geometric counterparts, generic building units (GBUs). CBUs represent chemical entities (*e.g.*, metal clusters and ligands), while GBUs define geometric roles (*e.g.*, 2-linear or 5-pyramidal). Assembly models then serve as blueprints for constructing MOPs from GBUs: for each GBU a corresponding set of CBUs was identified, and MOPs were generated by systematically recombining those CBUs according to the GBUs contained in the assembly models. Through this approach more than 1000 unreported MOPs were generated.

MOPs exemplify niche research areas where large datasets for bespoke model training or fine-tuning are scarce. The World Avatar specifically addresses such challenges by leveraging modular, lightweight ontologies that encode expert knowledge suitable for smaller datasets, supporting rule-based or hybrid agent workflows, as demonstrated by recent work on question-answering systems for MOPs.[29]

2.2 Synthesis procedures and standardisation

Extracting the desired information and data from these publications is time-consuming and automating the process is therefore highly desirable. The unstructured nature of synthesis descriptions, embedded with domain-specific language and inconsistencies in reporting styles, units, and naming conventions, complicates machine readability.[12,30] Standardised digital formats, such as XDL,[31] Chemical Markup Language (CML),[32] and SiLA standards,[33] have been developed to address this issue and to improve data interoperability in laboratory automation. Moreover, efforts like Suvarna *et al.*[12] emphasise structured reporting to enhance synthesis extraction, fostering advancements in chemistry automation. However, non-semantic approaches still face challenges in data linking and provenance tracking.[34]

Ontology-based approaches have emerged as a robust framework for addressing the limitations of traditional standardisation methods in chemistry, offering semantic interoperability through structured, machine-readable representations.[35,36] Ontologies such as Allotrope Foundation Ontology (AFO), Chemical Entities of Biological Interest (ChEBI),[37] and Royal Society of Chemistry's name reaction ontology (RXNO)[38] facilitate data organisation and retrieval by focusing on entities, reactions, or laboratory instrumentation. However, to our knowledge, no ontology currently exists that is specifically designed to represent stepwise, lab-scale synthesis procedures.[39] As a result, synthesis ontologies remain underdeveloped and underutilised in laboratory automation and digital chemistry workflows.[24] KGs, built upon Semantic Web principles, enhance data integration by linking heterogeneous datasets through ontologies.[40–42] Key technologies such as Resource Description Framework (RDF), Web Ontology Language (OWL), and SPARQL query language underpin these frameworks, ensuring interoperability and reasoning capabilities.[43–45] At the core of these technologies are triples – subject-predicate-object statements – that define relationships between entities, each uniquely identified by an Internationalised Resource Identifier (IRI).

2.3 Information extraction with large language models

Information extraction (IE) involves converting unstructured text into structured data, crucial for chemical analyses by identifying chemical entities and reaction conditions.[46–48] Traditional methods (rule-based and statistical) have limitations in scalability and adaptability,[48,49] highlighting the need for more flexible solutions. Large language models offer significant advancements due to their adaptability across diverse text formats.[15,50,51] However, unlike traditional IE systems, LLMs generate structured outputs that are not necessarily direct substrings of the input text. Following recent community conventions and the success of generative information extraction,[22,49,52] we refer to this approach as information extraction throughout this work.

Ensuring structured, consistent outputs from LLMs remains challenging, emphasising the necessity of advanced prompt engineering techniques.[53] Effective prompt engineering includes targeted content classification, modular retrieval, error mitigation, and ICL (zero- and few-shot prompting), which enhances adaptability without extensive retraining[14,16,50,54] and tackles the challenge of data scarcity for training in areas such as chemical synthesis.[21] Several advanced prompting strategies can further enhance LLM performance such as role prompting,[17,55] chain-of-thought prompting[18,54] and retrieval-augmented generation.[19,54] In addition, enforcing 'constrained output generation' ensures adherence to predefined schemas, a critical capability for structured data extraction.[49,56]

While LLMs have successfully been applied to extract structured data from tables in scientific papers and populate KGs,[22] extracting detailed information from completely unstructured free text and integrating it into highly structured representations like KGs remains challenging.[49] Recent studies demonstrate the potential yet indicate that iterative prompt refinement and validation are necessary to achieve reliable, ontology-aligned outputs.[23,57,58] OpenAI's introduction of Structured Outputs significantly addresses these issues, enabling responses to strictly adhere to developer-defined JSON schemas, thus enhancing the integration and robustness of AI-driven systems.[53,59] Nonetheless, context window limitations continue to restrict the amount of information that can be processed in a single inference, which is particularly relevant when extracting synthesis data from lengthy procedures or full-text articles.

3 Developing a novel ontology for chemical synthesis

In this section, we present our approach to overcoming aforementioned challenges by creating a modular, lightweight, and XDL-compatible ontology for synthesis procedures. This includes an analysis of synthesis procedures in the literature, the design of a structured ontology for synthesis workflows, its integration with existing ontologies such as OntoMOPs and OntoSpecies, and the implementation of semantic frameworks to facilitate machine-readable data representation and interoperability.

3.1 Preliminary data analysis

A preliminary analysis of the data used in this work helps us to develop competency questions outlining the scope and range of the ontology.[60] As the existing OntoMOPs domain in TWA includes MOPs from 75 publications curated by Kondinski *et al.*,[10] these were also chosen as a test case in this work for extracting synthesis information from and integrating these with existing knowledge in TWA. In a first step, a subset of these publications were screened manually to identify the information stored in the publications. After defining the information that is required, a second analysis using the OpenAI API evaluated what data can be possibly extracted and to identify potential issues. The resulting ontology competency questions are presented in the SI.

In the XDL standard, synthesis information is represented *via* markup language describing mainly three categories:[31] reagents, procedure, and equipment. As the purpose of the synthesis ontology developed in this work goes beyond the execution of synthesis recipe, characterisation data – which is commonly included in synthesis reports – becomes equally important, *e.g.* to verify reproduction or compare potential yields. For this reason, product characterisation was included as a forth category. This organisation into four categories became a fundamental principle for the ontology and pipeline design, ensuring comprehensive coverage of all relevant synthesis information. Each of the four categories presents distinct challenges and often necessitates additional contextual information for accurate interpretation.

These challenges include inconsistent reagent nomenclature, non-standardised procedural descriptions, and insufficiently detailed equipment reporting. Ambiguous references, such as using "1" for MOP product names,[61] hinder automated linkage with OntoMOPs. Customised entity matching strategies will be therefore necessary. Moreover, publications often describe multiple synthesis procedures, including those for precursors, and a single MOP may have several distinct synthesis routes. Therefore, the ontology must support multiple procedures per MOP, include detailed information on precursor synthesis, and accurately track the provenance of each procedure.

3.2 Ontology design

The findings from the preliminary analysis were distilled into the design of the OntoSyn ontology. The OntoSyn ontology models the transformation of input chemicals into their respective outputs, capturing synthesis procedures through structured steps. Fig. 1 shows a simplified version of the new ontology, highlighting key concepts and important connections with other ontologies. This diagram serves as a structural reference for how experimental synthesis data are semantically represented and integrated within TWA. The full ontology can be found in the SI.

Each ChemicalTransformation corresponds to a unique output and may be associated with multiple ChemicalSynthesis instances, as different synthesis procedures can exist for the same transformation across publications. Each ChemicalSynthesis instance captures essential synthesis details, including provenance information, which links experimental procedures to source documents using the bibo ontology.[62] Input chemicals are annotated according to the OntoCAPE ontology,[24] ensuring standardised representation. The synthesis steps detail process conditions and methodologies, while yield data is systematically represented using the "Ontology of units of Measure" (OM)[63] as AmountOfSubstanceFraction. The ontology supports synthesis procedures for MOPs and other materials, linking outputs to the OntoMOPs KG when applicable.[10] Since MOPs are assembled from CBUs – which are technically speaking fragments or moieties and not chemical species – OntoSyn establishes links between CBUs and the corresponding chemical species used as reactant in OntoSpecies *via* the predicate isUsedAsChemical. It should be noted that these are the only two links specific to the MOP use case; aside from this, the ontology is agnostic to material class and can be adapted to other reticular materials or general synthesis workflows.

The synthesis procedure itself is structured as a sequence of unit operations, each classified as a SynthesisStep, with specific step types implemented as subclasses. The ontology allows to specify a vessel, atmosphere, and duration independent of step type, while additional properties specific to certain step types allow the representation of customised information on the performed action. Where possible, step types align with existing XDL step categories to facilitate future interoperability. The predefined subclasses are: Add, HeatChill, Separate, Evaporate, Dry, Crystallize, Transfer, Filter, Stir, Sonicate, and Dissolve, which cover all synthesis procedures considered in this work. All of these sub classes, except Sonicate, correspond to XDL-defined actions.[64] In summary, the ontology builds on XDL's robust framework for describing unit operations for automation purposes,[31] extends it by concepts relevant for reproducibility and reticular chemistry while ensuring clarity for information extraction with LLMs and integration with the overall TWA knowledge base.

The interlinked OntoSpecies ontology includes concepts for Nuclear Magnetic Resonance (NMR) and mass spectrometry data,[26] yet lacks representation for infrared (IR) spectroscopy and elemental analysis, commonly reported in MOP synthesis procedures. To address this, we extended OntoSpecies with IR spectroscopy concepts derived from the Chemical Methods Ontology (CHMO),[65] introducing FourierTransformSpectrum as a subclass of AbsorptionSpectrum which refers to SpectralInformation. This structure enables IR spectra representation akin to NMR, utilising the existing Spectra Graph concept to define axes, units, and peak coordinates. Given OntoSpecies' original focus on emission spectroscopy, the term "peaks" was generalised to CharacteristicPeak, encompassing both peaks and absorption bands. Additionally, we integrate elemental analysis by distinguishing between calculated and experimental data, aligning ElementalAnalysis with subclasses CalculatedElementalAnalysis – derived from molecular formulae – and ExperimentalElementalAnalysis, which includes device specifications. This extension ensures OntoSpecies accommodates the most common characterisation techniques for MOPs: IR and elemental analysis data alongside existing characterisation methods, facilitating more comprehensive material property representation.

4 Building an automated pipeline: transforming scientific literature into structured knowledge

In this work, we introduce a structured pipeline developed to extract, process, and integrate synthesis data into TWA. We utilised OpenAI's GPT-4o model (gpt-4o-2024-08-06) to support information extraction and transformation tasks. This section details prompt engineering strategies and how they are employed to transform unstructured text to KG-compatible triples. Moreover, strategies for uploading and linking extracted information are discussed, which ensure the seamless integration of extracted data with existing knowledge in TWA while avoiding duplication and promoting an interconnected graph.

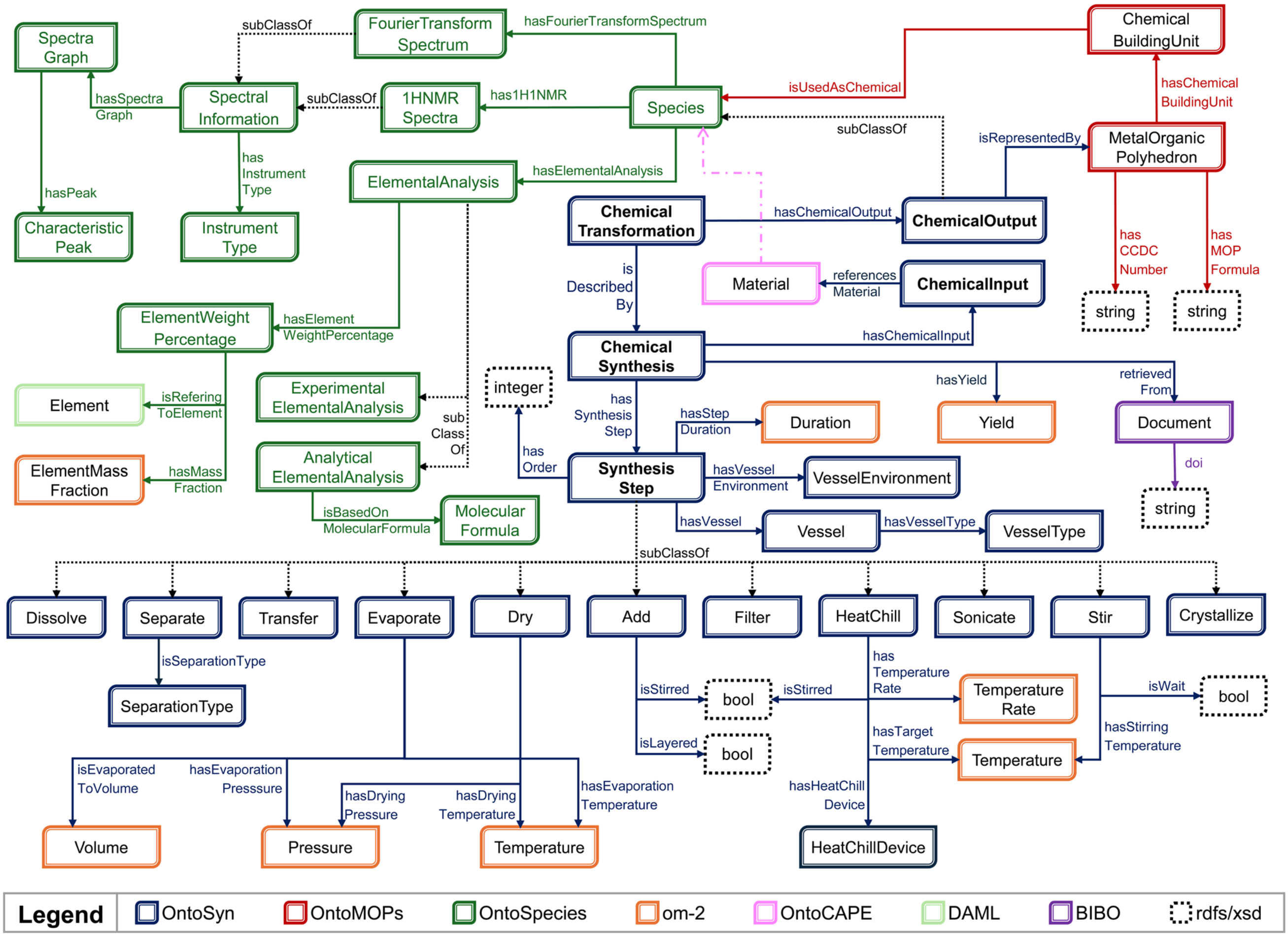


Fig. 1 Overview of the core classes and selected properties defined in the OntoSyn ontology, which formalises key concepts involved in synthetic chemistry workflows. The figure highlights relationships between the most important concepts as well as links to domain-specific classes from external ontologies such as OntoSpecies and OntoMOPs. Dashed lines represent indirect relationships, indicating intermediary concepts that have been omitted in this figure.

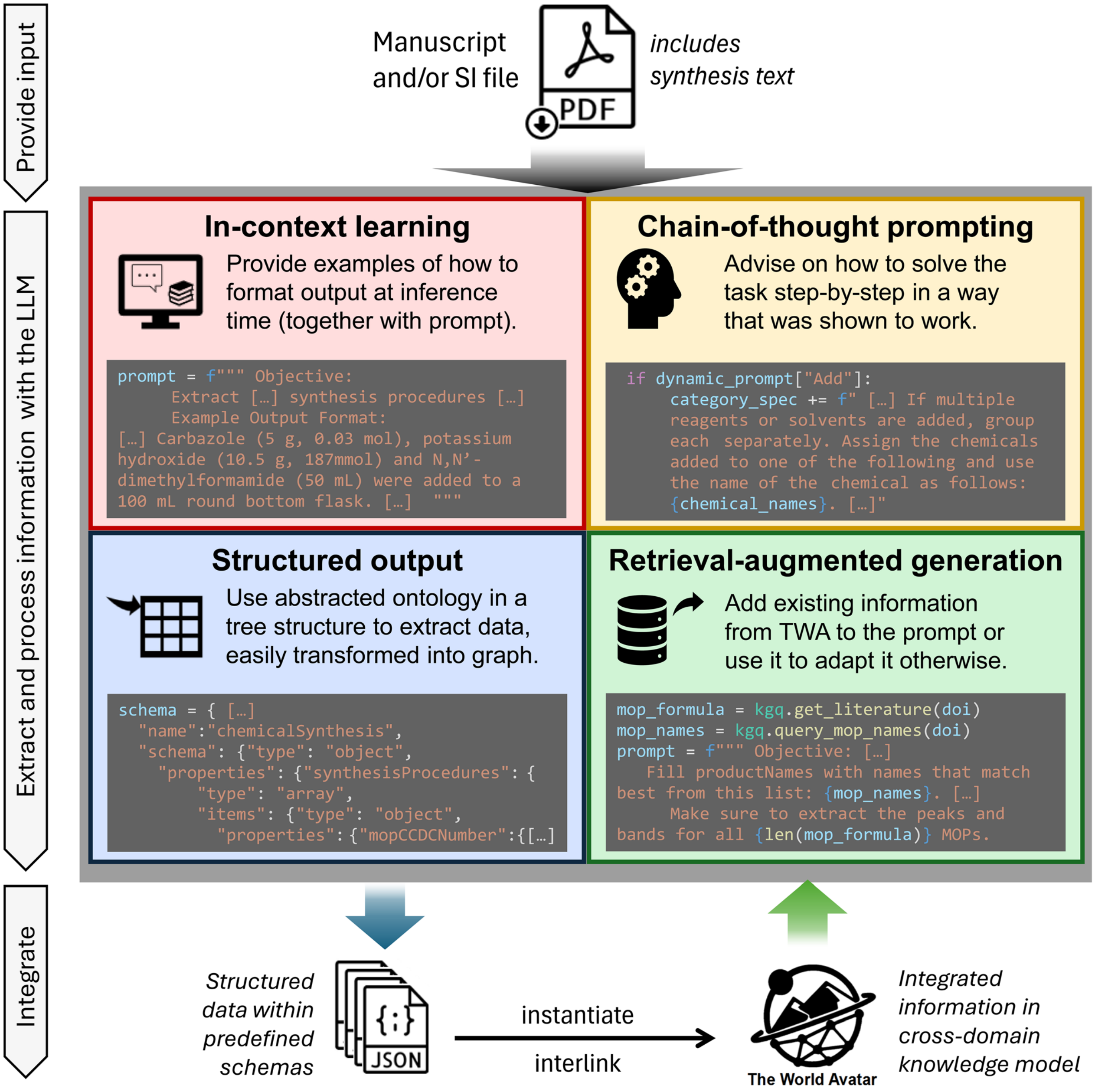
4.1 Pipeline overview

The information extraction pipeline was developed in a modular fashion for a variety of reasons, including:

- The complexity of simultaneous data extraction and KG integration.
- The restrictions on JSON schema size by the OpenAI model used in this work[59]
- Better control over the input and output of each module for debugging and adapting the pipeline.

Moreover, research by Sahoo *et al.*[54] has shown that breaking down tasks into smaller substeps can significantly improve the quality of LLM outputs, a core principle applied in this pipeline design. Based on this principle, the ontology was segmented into three prompting domains with each domain's data being extracted, uploaded, and linked separately. These domains chemicals, step types, and characterisation were selected based on the ontology design outlined in Subsection 3.2. Integrating each domain's data within the existing TWA knowledge base required different upload protocols and necessitated subsequent re-linking of separate data elements. These strategies are discussed in detail in Subsection 4.3.

In order to enforce this modularity throughout the pipeline while ensuring consistency of data extracted, every piece of information related to a specific synthesis procedure needs to be associated with a single unique attribute – a so-called primary key.[66] Therefore, extracted product names were immediately associated with each synthesis, serving as a pseudo-primary key in this structure to link and connect the files when they are uploaded. Linking the extracted data in the different files is an essential step for achieving meaningful integration within TWA. Without this linkage, unconnected subgraphs would be uploaded and the stored data holds limited value. To support the linking process, existing information within TWA was leveraged through RAG: querying pre-existing concepts and prompting the LLM to match exact string specifications significantly enhanced the reliability of these connections. A high-level overview of the transformation from synthesis text in PDF format to instantiated knowledge in TWA is depicted in Fig. 2. The key strategies of ICL, RAG, structured output, and CoT form the pillars of this LLM-based pipeline and are the basis for a reliable, targeted and hallucination-free data extraction and linkage with an LLM.


Provide input
Manuscript and/or SI file
PDF
includes synthesis text
Extract and process information with the LLM
In-context learning
Provide examples of how to format output at inference time (together with prompt).
Chain-of-thought prompting
Advise on how to solve the task step-by-step in a way that was shown to work.
Structured output
Use abstracted ontology in a tree structure to extract data, easily transformed into graph.
Retrieval-augmented generation
Add existing information from TWA to the prompt or use it to adapt it otherwise.
Integrate
Structured data within predefined schemas
{;}
JSON
instantiate
interlink
The World Avatar
Integrated information in cross-domain knowledge model

```
prompt = f""" Objective:
     Extract […] synthesis procedures […]
     Example Output Format:
[…] Carbazole (5 g, 0.03 mol), potassium
hydroxide (10.5 g, 187mmol) and N,N’-
dimethylformamide (50 mL) were added to a
100 mL round bottom flask. […]  """
```

```
if dynamic_prompt["Add"]:
    category_spec += f" […] If multiple
    reagents or solvents are added, group
    each separately. Assign the chemicals
    added to one of the following and use
    the name of the chemical as follows:
    {chemical_names}. […]"
```

```
schema = { […]
  "name":"chemicalSynthesis",
  "schema": {"type": "object",
    "properties": {"synthesisProcedures": {
      "type": "array",
      "items": {"type": "object",
        "properties":{"mopCCDCNumber":{[…]
```

```
mop_formula = kgq.get_literature(doi)
mop_names = kgq.query_mop_names(doi)
prompt = f""" Objective: […]
    Fill productNames with names that match
best from this list: {mop_names}. […]
     Make sure to extract the peaks and
bands for all {len(mop_formula)} MOPs.
```

Fig. 2 Overview of the LLM-based pipeline for synthesis data extraction and TWA integration visualised as abstract process from top to bottom. The modular pipeline calls the LLM seven times and thereby uses different prompt engineering techniques to extract synthesis information from manuscripts or SI Files, of which the four main ones are illustrated in the centre of the figure. Each box representing a key prompting strategy defines the main use case in the pipeline and provides an example below for the specific prompt engineering technique.

While Fig. 2 illustrates the prompt engineering strategies deployed throughout the pipeline, it does not indicate exact sequence or content of prompts used. Fig. 3 presents the complete pipeline workflow as a UML diagram, illustrating the interactions among four primary actors: the user, the pipeline agent, the LLM, and TWA. The user initiates the data extraction process by providing the synthesis text in PDF format. The pipeline agent iteratively constructs prompts for the LLM, incorporating both preexisting knowledge from the TWA and information obtained from earlier prompts. Each prompt is passed to the LLM, which returns structured responses, but not all responses are ultimately integrated into TWA. For example, Prompts 2 and 3 specifically serve to condense the synthesis text, isolating the segments pertinent to the current task, while Prompt 4 identifies and classifies the types of synthesis steps involved in the procedure.

4.2 Prompt strategies

In LLM-based synthesis data extraction, prompt engineering has proven essential, showcasing models' ability to generalise effectively to unseen data through the use of well-designed prompts.[67] The most important strategies used in this synthesis extraction pipeline are ICL, RAG, and CoT prompting.

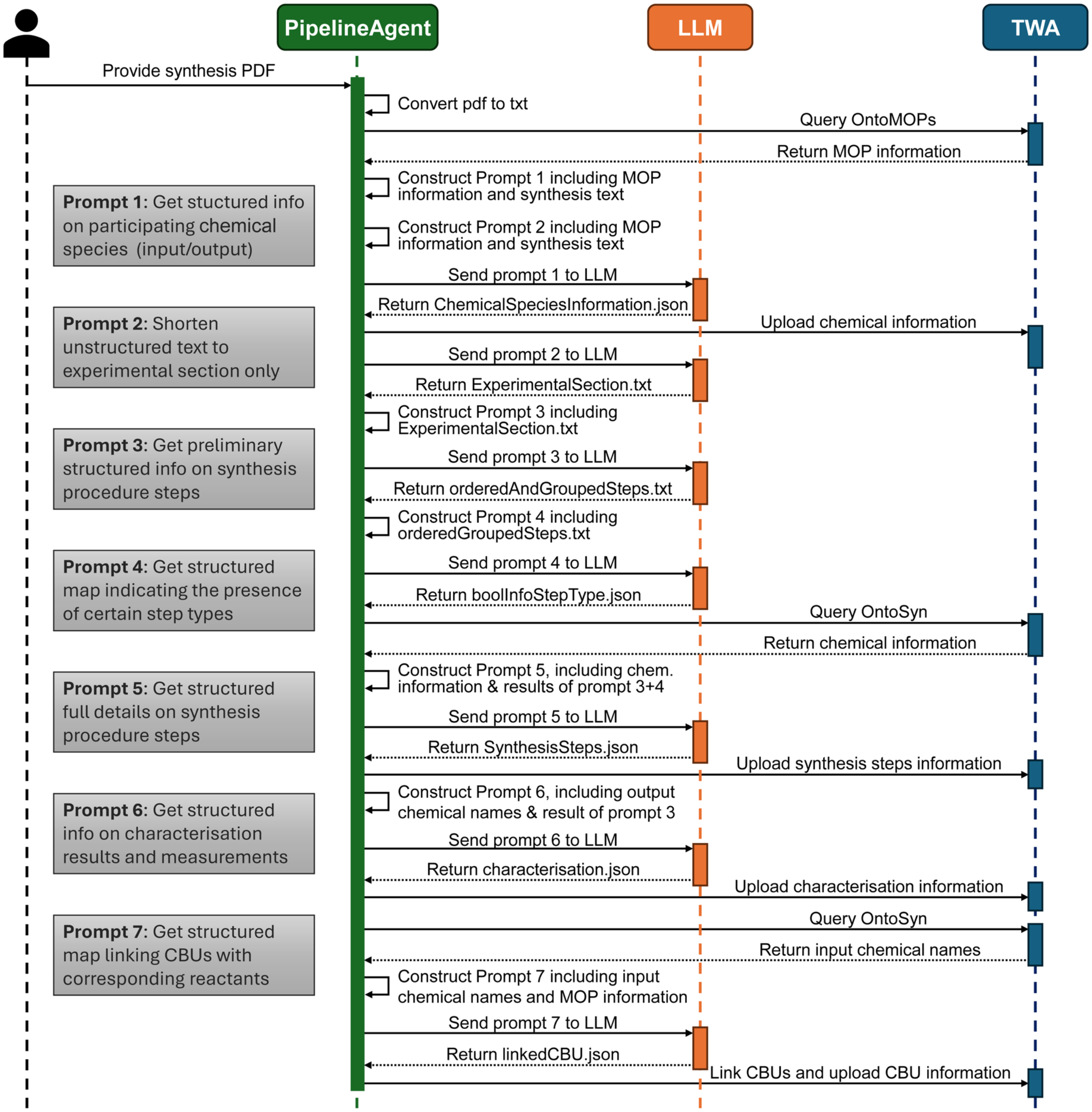

PipelineAgent
LLM
TWA
Provide synthesis PDF
Convert pdf to txt
Query OntoMOPs
Return MOP information
Construct Prompt 1 including MOP information and synthesis text
Construct Prompt 2 including MOP information and synthesis text
Send prompt 1 to LLM
Return ChemicalSpeciesInformation.json
Upload chemical information
Send prompt 2 to LLM
Return ExperimentalSection.txt
Construct Prompt 3 including ExperimentalSection.txt
Send prompt 3 to LLM
Return orderedAndGroupedSteps.txt
Construct Prompt 4 including orderedGroupedSteps.txt
Send prompt 4 to LLM
Return boolInfoStepType.json
Query OntoSyn
Return chemical information
Construct Prompt 5, including chem. information & results of prompt 3+4
Send prompt 5 to LLM
Return SynthesisSteps.json
Upload synthesis steps information
Construct Prompt 6, including output chemical names & result of prompt 3
Send prompt 6 to LLM
Return characterisation.json
Upload characterisation information
Query OntoSyn
Return input chemical names
Construct Prompt 7 including input chemical names and MOP information
Send prompt 7 to LLM
Return linkedCBU.json
Link CBUs and upload CBU information
Prompt 1: Get stuctured info on participating chemical species (input/output)
Prompt 2: Shorten unstructured text to experimental section only
Prompt 3: Get preliminary structured info on synthesis procedure steps
Prompt 4: Get structured map indicating the presence of certain step types
Prompt 5: Get structured full details on synthesis procedure steps
Prompt 6: Get structured info on characterisation results and measurements
Prompt 7: Get structured map linking CBUs with corresponding reactants

Fig. 3 Detailed UML activity diagram, illustrating the interactions between the four main actors: the user, the pipeline agent, the LLM, and TWA. The flowchart outlines the step-by-step process for uploading data, extracting chemical and synthesis information *via* prompt-based LLM interactions, and integrating the structured outputs into TWA. On the left side, a short synopsis on the purpose of each prompt is provided.

Furthermore, the structured output mode of the OpenAI API allows to reliable generate output that complies with a given JSON schema.[59] While ICL and CoT improve the reliability of information extraction from text, RAG and JSON schema-constrained outputs are critical for aligning that information with KG integration workflows.

As discussed in Subsection 2.3, in-context learning leverages examples placed within the prompt to guide the LLM in generating the desired response.[16,54] In this pipeline, each prompt incorporates examples to define the output structure. For text-generating prompts (Prompt 2 and Prompt 3), examples shape the expected format of free-text outputs, ensuring a consistent and structured response. Since plain text is inherently more ambiguous than structured formats, ICL plays a crucial role in maintaining uniformity in output format and structure. For structured output prompts that return JSON files, examples clarify the format and expectations of specific entries. An example of text formatting for ICL is shown in Fig. 2, while examples of JSON-formatted ICL are shown for most prompts in the SI. CoT prompting is used for prompts that require multiple reasoning steps, as it has been shown to improve the output quality in such instances.[54] In all three prompting domains we make use of this strategy to a degree, describing the information extraction procedure step by step – usually, by guiding the model from recognising certain text passages to categorising them and filling in specific parameters dependent on it. Fig. 2 includes an excerpt of Prompt 6 demonstrating CoT prompting. A chain of thought can also be established across multiple prompts: Prompt 3 for example groups, orders, and assigns step types in preparation for subsequent Prompts 4 and 5, which extract details on equipment, parameters, and reactants involved based on the step type.

The structured output mode introduced by OpenAI ensures compliance with predefined JSON schemas, providing a robust framework for data extraction and integration into KGs.[59] JSON inherently organises data in a hierarchical tree structure, where objects and arrays form nested parent–child relationships. Although effective for capturing hierarchical data, tree structures impose constraints that limit the representation of complex interconnections.[68] In contrast, graph structures, composed of nodes and edges, better model intricate, interconnected data.[69] The synthesis information in this case is used to populate KGs, but generating graphs data directly remains challenging:[57] while graph extraction as RDF model directly (*e.g.*, in "Turtle" file format) is possible and has previously been demonstrated by Caufield *et al.*,[70] integrating such outputs into a KG using multiple predefined ontologies and correctly linking entities is extremely difficult. The LLM-generated output likely is not perfectly formatted and still needs to be post-processed, there is no guarantee that the output complies with a given schema, and the output file structure may vary and potentially be wrong. Furthermore, the IRIs used in the extracted Turtle file still need to be mapped to previously instantiated instances to avoid duplicates. The pipeline circumvents these problems by relying on the structured output mode of the OpenAI API to generate reliable JSON files that follow the provided schema. These JSON files are designed to closely match the ontology in their structure and contain the values used to populate TWA. The entity linkages that transform the tree-like JSON file into a graph are fixed by the ontology and applied during the upload process, which ensures proper linking and instantiation.

RAG improves LLM responses by querying external data and adding it to the prompt.[19] This is especially helpful for knowledge extraction with TWA, where an existing and constantly updated knowledge base can be queried to improve prompts. In this case, the pipeline connects three different triplestores and even more ontologies. Five out of seven prompts are extended by existing data from either OntoMOPs or previously uploaded data from OntoSyn, allowing semantic knowledge to predict links and connect multiple outputs. Fig. 2 highlights how the chemical names are queried, saved to the variable chemical_names, and embedded in the prompt. Four specialised prompt strategies were developed to integrate knowledge from TWA, leverage information from the designed ontology, and tune prompts based on previous responses:

1. Knowledge-augmented prompting: Supplements prompts with relevant information queried from TWA, which serve as a set of reference values to guide the model. For instance, chemical names from input data can be matched with existing entries, such as in the OntoMOPs KG, or with newly generated information derived from extracted and uploaded data in OntoSyn.

2. Response-adaptive prompting: Dynamically adjusts prompts based on prior LLM outputs, without uploading intermediate data to TWA. Sub-prompts and schemas are composed on-the-fly using information from preceding prompts. For example, the LLM is instructed to generate a JSON file with boolean entries for each present step type in Prompt 4, based on which Prompt 5 is constructed from predefined sub-prompts for every step type, incorporating only relevant information to ensure efficient and targeted extraction.

3. Lookup table-driven extraction: Uses predefined value sets (*via* the enum keyword in JSON schemas) to restrict LLM responses. This forces selection from a fixed list including "unknown", reducing hallucinations and improving extraction reliability.

4. Prompt-based link generation: Queries instances from different classes and tasks the LLM with identifying specific links between them. Unlike knowledge-augmented prompting, this approach incorporates multiple KGs and focuses on generating connections. For example, Prompt 7 uses pre-queried instances from OntoMOPs (*e.g.*, CBUs) and OntoSpecies (*e.g.*, chemical species) KGs to establish links between them.

In addition to prompt design, the architecture of the pipeline itself plays a key role in overcoming limitations imposed by LLM context windows. Experimental sections are isolated from irrelevant content in Prompt 2, and different information categories are processed in a modular, sequential fashion. This allows long documents to be handled in smaller, coherent chunks and ensures that context-sensitive reasoning can still be performed effectively, even when full-text input exceeds model limits.

4.3 Uploading strategies

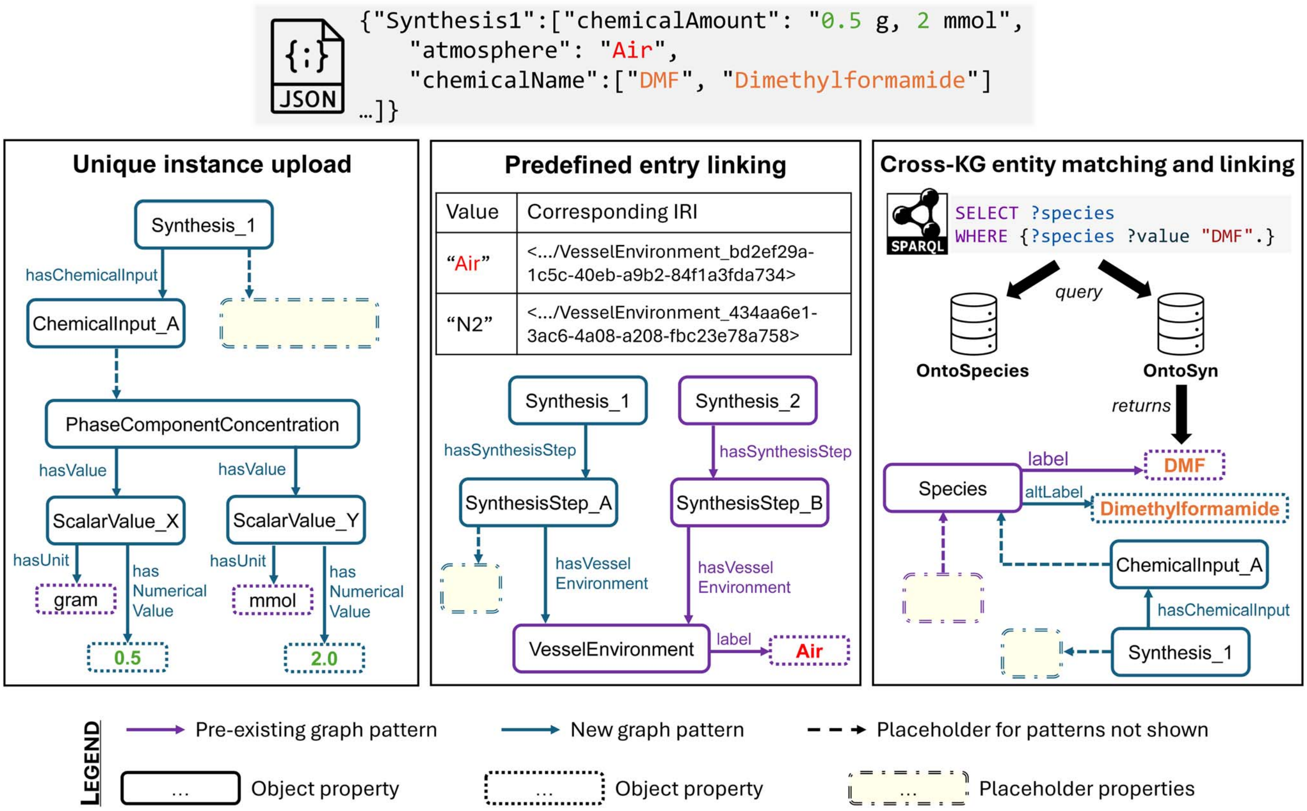
Transforming JSON data into a graph structure is critical for enabling flexible and interconnected data analysis. JSON inherently follows a hierarchical tree model, which, while useful for structured data, limits complex relationship representations.[68] In contrast, graph structures facilitate the modelling of complex, interconnected data beyond hierarchical limitations.[69] Converting JSON data into a graph structure involves mapping JSON objects and arrays to graph nodes, and their relationships to edges, thereby preserving the original data's semantics while enabling richer interconnections. As detailed in the SI, the JSON files closely resemble the ontology. Using the TWA Python package, those JSON objects and hierarchy are instantiated within the program logic and seamlessly pushed to their respective knowledge graphs *via* the object-graph mapper.[28] This structured approach allows us to programmatically link the different entities with each other beyond the tree structure.

As detailed in Subsection 4.1, the pipeline extracts and integrates synthesis data from three general domains, with information for each domain stored in separate JSON files. This results in the creation of multiple JSON files per literature source, necessitating the linkage of subgraphs across these domains. Within each domain, the chemical output name is extracted and serves as a primary key, enabling the integration of subgraphs corresponding to specific chemical transformations. By connecting each domain-specific subgraph through the chemical output class, these fragments are effectively linked. Previously extracted names and information on uploaded output chemicals are added to the prompt with RAG and serve as selectable options during language model extraction. Providing a selection of possible names for a certain chemical increases the chance of assigning the name used in the paper to the previously extracted name. For example, if a paper describes a synthesis using the abbreviation "DMF", the previously instantiated entity with the label "*N*,*N*-dimethylformamide" can be leveraged so that the LLM is able to recognise that they are the same chemical and extracts both names. This allows to further expand the "*N*,*N*-dimethylformamide" instance with the new label "DMF". Another helpful tool is the existing OntoSpecies TWA subgraph that stores information on commonly used species.[26] Even when the LLM does not correctly assign the name "DMF" in the text to TWA instance with label "*N*,*N*-dimethylformamide", the OntoSpecies KG is queried while uploading the species. Querying "DMF" in OntoSpecies retrieve the stored knowledge on it being the abbreviation for "*N*,*N*-dimethylformamide" and the instances will be linked during uploading.

Beyond inter-file linkages, avoiding duplicate entity creation is crucial. Three uploading strategies are employed based on instance characteristics. Fig. 4 illustrates these strategies. Unique instance upload is the simplest case, where each extracted instance is newly instantiated, independent of the source ontology. No duplicate checking is required, as instances do not recur. Scalar values are good examples of this, as they uniquely represent a number-unit combination within a paper as shown for the two values representing chemical amounts in Fig. 4.

Certain ontology classes contain a finite, predefined set of instances, often derived from OntoSyn and originally defined through manual literature review. Examples include JSON entries from "lookup-table driven extraction" (Subsection 4.2) such as atmospheres (*e.g.*, "air") and temperature units. These predefined instances guide LLM outputs by restricting extractions to known values – *e.g.*, a predefined set of vessel types. Predefined entry linking requires these instances to be uploaded to TWA at the start of the pipeline. When extracted, values are matched against stored IRIs *via* a lookup table, ensuring consistency and preventing duplicates.

Some classes have an open set of possible values, allowing for an infinite number of variations, yet certain values recur throughout the data. To avoid duplicates they need to be uploaded only the first time and linked otherwise. The rightmost uploading workflow depicted in Fig. 4 named cross-KG entity matching and linking handles these cases. When uploading such instances, it is essential to first check the OntoSyn KG to verify whether an instance of the same entity already exists, thereby avoiding duplicate entries. A significant challenge arises when the same entity is extracted with different labels, making consistent linking difficult. To address this, we include alternative labels and attempt to extract as many



```
{"Synthesis1":["chemicalAmount": "0.5 g, 2 mmol",
   "atmosphere": "Air",
   "chemicalName":["DMF", "Dimethylformamide"]
…]}
```

Value	Corresponding IRI
"Air"	<…/VesselEnvironment_bd2ef29a-1c5c-40eb-a9b2-84f1a3fda734>
"N2"	<…/VesselEnvironment_434aa6e1-3ac6-4a08-a208-fbc23e78a758>

```
SELECT ?species
WHERE {?species ?value "DMF".}
```

Fig. 4 Illustrative example of uploading a JSON-formatted output of LLM-based IE, demonstrating three key techniques: unique instance instantiation, predefined entry linking, and cross TWA-subgraph entity matching and linking. The figure shows how each technique generates new graph patterns and establishes links between newly instantiated entities and existing.

variants and relevant data as possible. The process becomes more complex when handling instances from external ontologies with entries outside the TWA stack. To avoid duplicates – where the same entity may have different IRIs – it is necessary to query all relevant KGs *via* their SPARQL endpoints to check for existing instances of the entity. This ensures consistent representation and minimises duplicate entries.

4.4 Performance evaluation

The pipeline presented in this work does not just extract synthesis information and convert it into a structured data format – it integrates it within a body of pre-existing knowledge. This makes quantitative performance evaluation inherently challenging, as results depend not only on extraction quality but also on the current state of related knowledge in TWA. To assess the pipeline's effectiveness in extracting relevant information, we manually curated schemas for a test set of 9 papers and compared both the schemas and the resulting knowledge graphs to those produced automatically by the pipeline. Comparison was based on exact value matches, yielding standard performance metrics (precision, recall, *F*1 score), detailed in the SI.

Analysing the JSON schema results, the pipeline achieves an overall *F*1 of 84.7% with 4224 true positives and only 768 false negatives. A breakdown by category shows strong performance in extracting chemical entities, synthesis steps, and characterisation details, but somewhat lower performance for CBU components. This discrepancy is primarily due to integration challenges: the pipeline often extracts initial reagents, while OntoMOPs expects curated CBU formulas—highlighting that the main difficulty lies not in extraction *per se*, but in aligning extracted data with structured domain knowledge. For the knowledge graph comparison, which is based on exact predicate-literal matches, performance is even higher, with an *F*1 score of 94.2%. This reinforces the reliability of the pipeline in both extracting and linking meaningful information. Unlike the JSON schema comparison, this does not depend on the order of data and thus we observe higher performance. These results are on par with state-of-the-art literature extraction methods,[23,58] underscoring the effectiveness and reliability of the pipeline to accurately extract and link information from the papers.

In this work, a total dataset of 75 publications was analysed, based on the selection by Kondinski *et al.*,[10] with few exclusions due to access restrictions and data processing errors. Of the 75 initially selected articles, 69 were successfully processed without manual intervention. One of the main achievements in this dataset is the linkage of 102 out of 151 MOPs present in TWA to chemical outputs, with 78 of the total 127 CBUs successfully linked to reactants. This connectivity is crucial for future developments and an understanding of the relations between OntoMOPs concepts and actual chemical species and provides the foundation for future synthesis predictions. Of 565 unique species instantiated within the synthesis procedures, 88 were detected in the OntoSpecies KG and linked accordingly.

5 Applications

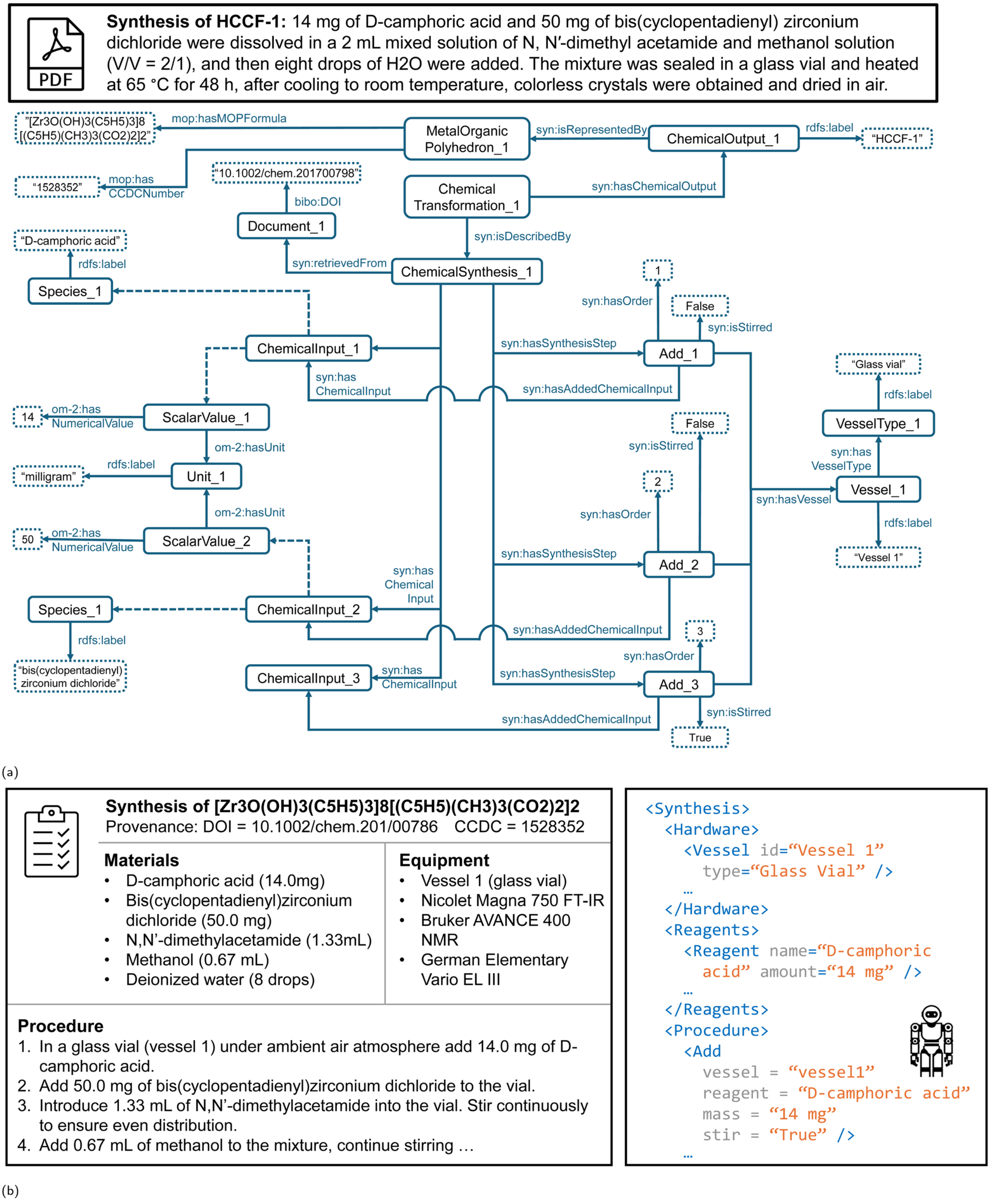
In this work, we used the OpenAI API[59] for prompting tasks when running the pipeline presented in section 4 on 75 preselected articles.[10] It is important to emphasise that the pipeline itself is largely independent of the LLM used and could be changed to any general purpose LLM API. In this section we demonstrate the extracted knowledge's practical applications and explore broader implications for the field.

5.1 Knowledge graph-based assembly of synthesis procedures

Unlike traditional static synthesis procedure formats, the presented TWA-based framework provides a flexible and dynamic structure that can adapt to both human and machine requirements. As shown in Fig. 5, this adaptability ensures that synthesis data can be easily interpreted by researchers while simultaneously offering the structured, queryable format necessary for automation and computational analysis. Custom-tailored synthesis recipes or other output formats can now be generated by querying all steps and characterisation data of a specific synthesis and reassembling it into a new "synthesis generation prompt" that gets again fed to an LLM API. Similar to the pipeline prompts described in section 4, these prompts would employ RAG and ICL techniques.

This workflow allows chemists to query the TWA and retrieve synthesis protocols in a semi-structured, readable, and standardised format. Due to the TWA's structured nature, outputs can be flexibly formatted based on specific requirements. An exemplary output is illustrated on the left side of Fig. 5b while the original synthesis text as well as an excerpt of the representation in TWA can be found in Fig. 5a for comparison. The overall workflow of pipeline extraction and reassembling the synthesis procedure worked very well and even outperformed the original text by presenting the information in a more concise and structured manner. The data in TWA correctly resembles the used vial, the chemicals, and even captured the mixing relationship of methanol and DMF and adjusted the specified amount, yet failed to add them in one step. Even though the output is well-structured and captures most of the important information presented in the literature procedure, there are minor flaws in the LLM-generated description.

In the structured output shown on the left side of Fig. 5b, data are categorised into the four main categories identified in Subsection 3.1: chemicals, equipment, step types, and characterisation. This organised approach differs from traditional documentation of synthesis procedures, where these domains are often intertwined, requiring additional reading and interpretation. The text is written in a generalised and consistent style, eliminating ambiguities and minimising room for interpretation. For example, implicit details such as "room temperature" are explicitly stated in the new format, providing clarity. The structured output also offers a significant advantage by concisely listing chemicals and equipment before the synthesis process begins, aiding chemists in identifying essential materials more efficiently.



Synthesis of HCCF-1: 14 mg of D-camphoric acid and 50 mg of bis(cyclopentadienyl) zirconium dichloride were dissolved in a 2 mL mixed solution of N, N′-dimethyl acetamide and methanol solution (V/V = 2/1), and then eight drops of H2O were added. The mixture was sealed in a glass vial and heated at 65 °C for 48 h, after cooling to room temperature, colorless crystals were obtained and dried in air.

Synthesis of [Zr3O(OH)3(C5H5)3]8[(C5H5)(CH3)3(CO2)2]2
Provenance: DOI = 10.1002/chem.201/00786 CCDC = 1528352

Materials
- D-camphoric acid (14.0mg)
- Bis(cyclopentadienyl)zirconium dichloride (50.0 mg)
- N,N'-dimethylacetamide (1.33mL)
- Methanol (0.67 mL)
- Deionized water (8 drops)

Equipment
- Vessel 1 (glass vial)
- Nicolet Magna 750 FT-IR
- Bruker AVANCE 400 NMR
- German Elementary Vario EL III

Procedure
1. In a glass vial (vessel 1) under ambient air atmosphere add 14.0 mg of D-camphoric acid.
2. Add 50.0 mg of bis(cyclopentadienyl)zirconium dichloride to the vial.
3. Introduce 1.33 mL of N,N'-dimethylacetamide into the vial. Stir continuously to ensure even distribution.
4. Add 0.67 mL of methanol to the mixture, continue stirring …

```
<Synthesis>
  <Hardware>
    <Vessel id=“Vessel 1”
      type=“Glass Vial” />
    …
  </Hardware>
  <Reagents>
    <Reagent name=“D-camphoric
      acid” amount=“14 mg” />
    …
  </Reagents>
  <Procedure>
    <Add
      vessel = “vessel1”
      reagent = “D-camphoric acid”
      mass = “14 mg”
      stir = “True” />
    …
```

Fig. 5 Illustration of different stages of synthesis knowledge representation – from instructions in PDF article to instances in a KG to tailor-made output formats – demonstrate the flexibility and versatility of the TWA-based synthesis representation developed in this work. (a) Original synthesis text from journal article PDF and illustration of corresponding instantiated triples in TWA after synthesis extraction. (b) Exemplary output formats: one human-readable (bill of materials and recipe-style instructions), one machine-readable (*i.e.* XDL).

Alternatively, given the strong alignment between the ontology and XDL format detailed in section 3.2, KG data can be translated into an XDL file. Most synthesis steps share class attributes with XDL properties (see SI for details), allowing for programmatic population of a template XDL file from corresponding triples. This offers a major advantage over previous approaches, where the creation of XDL schemes from literature remained largely manual due to the linguistic challenges of parsing free-text procedures.[71] Because of the fact that XDL was developed with a focus on organic chemistry and lately flow chemistry in particular while OntoSyn is for reticular material synthesis, minor differences in step types and attributes require future harmonisation. The sonication step, unique to OntoSyn, is absent in XDL.[64] Additionally, some information is not reliably extractable using the LLM approach, though advancements in XDL and LLMs will mitigate this. Key step types such as Add, HeatChill, and Filter are already supported. Extracting vessel information remains challenging. OntoSyn currently represents only the main vessel per step, except in transfer steps, whereas XDL requires explicit details for all involved vessels. Lastly, some XDL steps, such as Separate, necessitate both target and waste vessels. However, synthesis texts seldom specify vessels, complicating perfect alignment.

5.2 Data analysis and retrosynthesis

The collection of highly structured and interlinked MOP synthesis information opens up a unique opportunity for analysing procedures, parameters, equipments, and yields to identify trends that can inform future synthesis planning. Such a structured and comprehensive knowledge base has been envisioned as a critical enabler for digital reticular chemistry, as suggested by Lyu *et al.*[72]. Particularly, linking synthesis procedures to the building units and assembly models of the resulting MOP structures allows us to uncover design rules. These rules can be applied for retrosynthesis – working backword from a desired MOP structure to suggest plausible synthetic routes as suggested by Kondinski *et al.*[10]. Fig. 6 illustrates two examples of our preliminary data analysis that provides insights potentially applicable to the retrosynthesis of novel MOPs.

As shown in Fig. 6a, the mapping between reactants and CBUs is not always one-to-one. This is particularly true for transition metals such as vanadium, which can form various complexes depending on their oxidation states and spin configurations. In such cases, the choice of solvent and the heating regime have a substantial impact on determining the final CBU as well. Multiple reactant combinations are typically possible and even multiple reactants are sometimes used as vanadium source, making synthesis prediction for these systems especially complex. For closed-shell metal CBUs, the mapping is generally more straightforward and often follows a one-to-one pattern. In contrast, organic CBUs can be usually mapped to a specific single reactant, simplifying predictive modelling.

Fig. 6b demonstrates a strong correlation between the type of metal CBU and the heating regime employed during synthesis. Specific metal centres are consistently associated with distinct thermal profiles, leading to visible clustering patterns in the analysis. Interestingly, when the same analysis is performed using organic CBUs as the categorisation criterion, no such correlation is observed. This observation aligns well with chemical intuition: heating primarily serves to activate the metal complex, after which the self-assembly of the structure takes place. From a retrosynthesis standpoint, this suggests a useful heuristic: when designing a novel MOP based on a known structure, retaining the metal core while substituting the organic ligand is likely to preserve the required thermal conditions for synthesis.

Lastly, analysis of the extracted synthesis data can also reveal insights into common reporting practices of such procedures apart from retrosynthesis applications. For example, the structured data enables analysis of characterisation trends across the synthesis literature. Among 272 documented chemical outputs, elemental analysis is the most frequently used verification method, appearing in 213 cases. Other techniques, such as IR spectroscopy and NMR, are comparatively less common. Furthermore, synthesis yield is reported in only 115 of the 291 documented procedures, highlighting a general lack of emphasis on quantitative output in the literature. These insights not only inform future synthesis reporting practices but also guide data prioritisation for predictive modelling and automated synthesis planning.

5.3 Discussion

Despite segmenting the workflow into seven prompts (see Fig. 3), synthesis data extraction remains challenging. Therefore, as suggested by prior work,[12,31] a standardised way of reporting synthesis procedures could greatly benefit the computational community by facilitating more efficient and precise data extraction. Even small changes – such as the standardisation of the language used to describe certain actions performed in the lab, or structuring the synthesis procedure in a generally agreed-upon clear and compact way – could reduce ambiguity and improve LLMs ability to parse this data. In the longer term, such standardisation could even enable reliable data extraction without the need for large language models, reducing computational cost and environmental impact. At the moment, the way synthesis procedures are reported is often constrained by the formatting guidelines and section structures imposed by the journals in which articles are published. Therefore, publishers play a critical role and should take the lead in developing and promoting unified standards for reporting experimental procedures.

In the synthesis example of MOP $[Zr_3O(OH)_3(C_5H_5)_3]_8[(C_5H_5)(CH_3)_3(CO_2)_2]_2$ partially illustrated in Fig. 5, TWA data correctly resembles the used vial, the chemicals, and even captured the mixing relationship of methanol and DMF. It adjusted the specified amounts, yet failed to add them in one step. Even though the output is well-structured and captures most of the important information presented in the literature procedure, there are minor flaws in the LLM-generated description: Water and deionised water are confused in the uploading process as the same entity causing

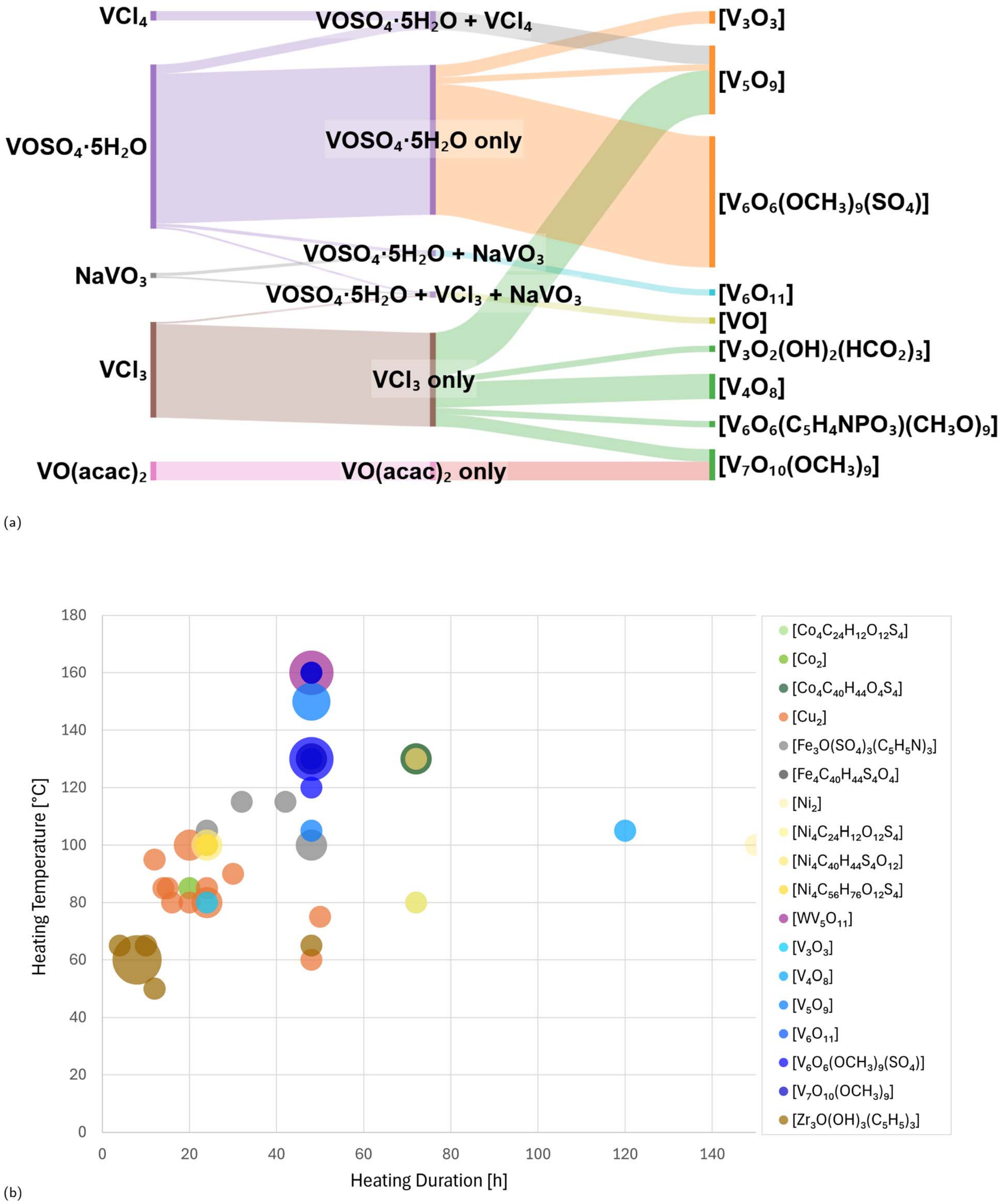


Fig. 6 Exemplary data analyses of the recorded MOP synthesis procedures reveal correlations between building units and synthesis conditions that might be used for retrosynthesis. (a) Sankey diagram of metal CBUs on the right and their corresponding reactants on the left used in recorded synthesis procedures for selected vanadium-based MOPs. Reactant mixtures are indicated in the middle where applicable. (b) Heating temperature and duration used for crystallisation of selected MOPs in recorded synthesis procedures. Data points are colour-coded by the metal CBU present, larger points indicate multiple MOPs with a certain metal CBU are synthesised with the same heating regime.

the confusion in the output text. Furthermore, the LLM often extracted isStirred as required even though the text does not explicitly mention it. While it is often sensible to imply stirring in synthesis steps, even when not explicitly mentioned, instructions that involve stirring a single reactant or mixture of powders might confuse researchers. As ontology complexity increases, refining prompts will be critical. Restricting outputs to predefined values effectively mitigates hallucination, though human oversight in data screening remains a bottleneck.

RAG prompting worked well and guided the LLM to the expected values in the desired format. In this context, data quality was found to be tremendously important as wrongly extracted and uploaded values that are reused later on propagate through the RAG approach. One example of this we encountered once was a mixture that the LLM did not properly recognise as such and instead instantiated a single new species of the name "["Ethanol", "Water"]". As a result, all the synthesis steps that were connected to either ethanol or water were connected to the mixture instance even though only either of the species was used. To address this issue, the JSON schema responsible for chemical name extraction and the uploading procedure were updated, and the extraction process was repeated to ensure accuracy. Expanding RAG to integrate IRI queries during prompt generation could further streamline entity mapping, potentially allowing direct JSON-to-triple conversion. The current approach delays IRI mapping until data upload, avoiding LLM misinterpretation of lengthy IRIs with random character sequences.

Designing JSON schemas that closely align with the ontology and subdividing the ontology into different JSON files allows to extend the data pipeline to very large ontologies as previously observed by Meyer *et al.*[57]. In our experience, joining different extracted JSON files as described in Subsection 4.3 works well but still occasionally fails and generates disconnected subgraphs that are of little value. The more the ontology is split up into different files, the more linking is required which increases the risk of faulty links or uploads. Therefore, finding the right balance between modularising ontology subdomains for the pipeline and preserving overall integrity and interlinkedness remains a critical design challenge. Overall, refining standardisation, ontology expansion, and prompt engineering will enhance pipeline efficiency and extraction accuracy while minimising manual intervention.

While this work focuses on MOPs, the OntoSyn ontology is designed to generalise across other material classes, particularly reticular materials. With minor adjustments to prompts and program logic, and by linking to a material-specific ontology, the pipeline can be adapted for broader applications. Nonetheless, some limitations remain. The ontology does not yet capture complex experimental setups, and handling very large datasets may introduce scalability challenges. Despite prompt engineering and schema constraints, occasional hallucinations persist and require human oversight.

6 Conclusion

Building on advancements in reticular chemistry and LLMs while leveraging the capabilities of TWA, a universal digital twin based on interlinked dynamic KGs, this work demonstrates a comprehensive framework to automate synthesis procedure extraction. One major contribution of this work lies in developing the OntoSyn ontology and verifying its utility for representing and comparing chemical synthesis procedures. By addressing the drawbacks of traditional synthesis documentation, this ontology aims to bridge the gap between unstructured textual data and rigidly structured, machine-readable formats. The ontology's alignment with standards such as XDL and its integration with existing domain ontologies, including OntoMOPs and OntoSpecies, exemplifies its interoperability. The key achievement was the creation of a fully automated pipeline for the extraction of synthesis information from scientific literature and integration within a dynamic system of pre-existing knowledge and structure requirements.

Through systematic prompting strategies, such as chain-of-thought reasoning, retrieval-augmented generation, and in-context learning, the pipeline successfully extracts and structures detailed synthesis information, demonstrating high accuracy and consistency. Four strategies were developed to leverage TWA to support the synthesis data extraction: knowledge-augmented prompting integrates existing knowledge into prompts to refine data selection; response-adaptive prompting helps designing iterative prompts and schemas to systematically analyse publication content and guide future prompt construction; lookup table-driven extraction leverages JSON schema capabilities for structured data extraction; lastly, prompt-based link generation employs LLMs to establish entity linking by embedding instances directly into prompts, enabling seamless integration and alignment within the TWA framework.

Applying this LLM-based pipeline to a considerable number of MOP synthesis publications provided several unique insights. First, the highly structured and detailed representation of synthesis procedures enables the creation of tailor-made lists and protocols that are more practical and efficient for various stakeholders in chemistry labs. Notably, the alignment of the OntoSyn ontology with robotic execution standards such as XDL emphasises its potential to support autonomous laboratories and machine-executable chemistry. Second, the ability to semantically link experimental synthesis procedures with corresponding MOP structures allows for the exploration of correlations between structure and synthesis conditions. These correlations can be leveraged to predict synthesis pathways for novel MOPs with desirable properties, reinforcing the importance of knowledge–driven approaches in hypothesis generation and experimental design. Finally, promoting greater consensus among researchers and publishers on how synthesis instructions should be structured and documented would significantly advance the field. Beyond improving reproducibility, such standardisation would enable more accurate and efficient data extraction, potentially reducing reliance on large language models and facilitating the development of large-scale knowledge bases essential for uncovering complex relationships between molecular structure and synthetic strategy.

Looking ahead, this work lays the foundation for a range of promising developments. From enhancing automation by enabling autonomous literature discovery and data extraction, to leveraging structured formats like XDL for more streamlined knowledge integration, many directions remain open. The interoperability built into OntoSyn allows not only for broader data sourcing but also for generating predictive models to support synthesis planning. Continued advances in large language models—including improved reasoning and larger context windows—are expected to further enhance extraction accuracy and may enable simpler, more efficient workflows. Expanding the ontology to new chemical domains and integrating with robotic platforms could further unlock applications in autonomous laboratories. Moreover, refining the pipeline and establishing connections with other autonomous agents will be key to building more cohesive and intelligent systems. For instance, automatically running the PubChem agent[26] on newly instantiated species within the knowledge graph would retrieve additional physico- and thermochemical properties, enriching the contextual information available for synthesis planning and validation.

In conclusion, this publication provides a robust framework for automating synthesis discovery and material design beyond reticular chemistry, addressing key challenges in scalability, reproducibility, and data integration. By combining AI-driven automation, semantic knowledge representation, and data modelling, it lays a solid foundation for a future transformation of synthetic chemistry from an empirical to a data- and knowledge-driven discipline.

Conflicts of interest

There are no conflicts to declare.

Data availability

All codes and ontologies developed are available on Github under MIT license: **https://github.com/TheWorldAvatar/MOPTools**. The underlying repository contains additional tools and applications related to metal–organic polyhedral and can be permanently accessed on Zenodo *via* DOI: **https://doi.org/10.5281/zenodo.16410991**. Prompts and JSON templates used for LLM interactions were generated at runtime *via* the available code: text blocks and logic for prompt generation can be found under /MOP_Literature_Extraction/llm_prompts.py. Text blocks and logic for template generation are available under /MOP_Literature_Extraction/json_schemas.py. A detailed description of each software module can be found in /MOP_Literature_Extraction/docs/readme.md. Furthermore, the SI contains an example of each prompt and template as well as a list of all extracted articles. The full data set including the ontology, all extracted knowledge graph triples, as well as test and reference data sets are available in the Cambridge repository *via* DOI: **https://doi.org/10.17863/CAM.118147**

Supplementary information is available. See DOI: **https://doi.org/10.1039/d5dd00183h**.

Acknowledgements

This research was supported by the National Research Foundation, Prime Minister's Office, Singapore under its Campus for Research Excellence and Technological Enterprise (CREATE) programme. S. D. Rihm acknowledges financial support from Fitzwilliam College, Cambridge, and the Cambridge Trust. Financial support from the UK Engineering and Physical Sciences Research Council (EPSRC) grant EP/Y016076/1 is also gratefully acknowledged. For the purpose of open access, the author has applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising. During the preparation of this work the authors used generative AI tools in order to enhance the readability and language of the manuscript. After using these tools, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Notes and references

1 A. O. Adeola, J. O. Ighalo, P. I. Kyesmen and P. N. Nomngongo, *J. CO2 Util.*, 2024, **80**, 102664, DOI: **10.1016/j.jcou.2023.102664**.

2 S. P. Argent, I. Da Silva, A. Greenaway, M. Savage, J. Humby, A. J. Davies, H. Nowell, W. Lewis, P. Manuel, C. C. Tang, A. J. Blake, M. W. George, A. V. Markevich, E. Besley, S. Yang, N. R. Champness and M. Schröder, *Inorg. Chem.*, 2020, **59**, 15646–15658, DOI: **10.1021/acs.inorgchem.0c01935**.

3 G. R. Lorzing, A. J. Gosselin, B. S. Lindner, R. Bhattacharjee, G. P. Yap, S. Caratzoulas and E. D. Bloch, *Chem. Commun.*, 2019, **55**, 9527–9530, DOI: **10.1039/C9CC05002G**.

4 S. Lee, H. Jeong, D. Nam, M. S. Lah and W. Choe, *Chem. Soc. Rev.*, 2021, **50**, 528–555, DOI: **10.1039/D0CS00443J**.

5 H. Vardhan, M. Yusubov and F. Verpoort, *Coord. Chem. Rev.*, 2016, **306**, 171–194, DOI: **10.1016/J.CCR.2015.05.016**.

6 W.-H. Xing, H.-Y. Li, X.-Y. Dong and S.-Q. Zang, *J. Mater. Chem. A*, 2018, **6**, 7724–7730, DOI: **10.1039/C8TA00858B**.

7 A. C. Ghosh, A. Legrand, R. Rajapaksha, G. A. Craig, C. Sassoye, G. Balázs, D. Farrusseng, S. Furukawa, J. Canivet and F. M. Wisser, *J. Am. Chem. Soc.*, 2022, **144**, 3626–3636, DOI: **10.1021/jacs.1c12631**.

8 D. J. Tranchemontagne, Z. Ni, M. O'Keeffe and O. M. Yaghi, *Angew. Chem., Int. Ed.*, 2008, **47**, 5136–5147, DOI: **10.1002/ANIE.200705008**.

9 H. Park, Y. Kang, W. Choe and J. Kim, *J. Chem. Inf. Model.*, 2022, **62**, 1190–1198, DOI: **10.1021/acs.jcim.1c01297**.

10 A. Kondinski, A. Menon, D. Nurkowski, F. Farazi, S. Mosbach, J. Akroyd and M. Kraft, *J. Am. Chem. Soc.*, 2022, **144**, 11713–11728, DOI: **10.1021/JACS.2C03402**.

11 A. Kondinski, A. M. Oyarzún, S. D. Rihm, J. Bai, S. Mosbach, J. Akroyd and M. Kraft, Automated Assembly Modelling of Metal-Organic Polyhedra, *Eur. J. Inorg. Chem.*, 2025, e202500115, DOI: **10.1002/ejic.202500115**.

12 M. Suvarna, A. C. Vaucher, S. Mitchell, T. Laino and J. Pérez-Ramírez, *Nat. Commun.*, 2023, (14), 1–11, DOI: **10.1038/s41467-023-43836-5**.

13 Q. Zhu, F. Zhang, Y. Huang, H. Xiao, L. Y. Zhao, X. C. Zhang, T. Song, X. S. Tang, X. Li, G. He, B. C. Chong, J. Y. Zhou, Y. H. Zhang, B. Zhang, J. Q. Cao, M. Luo, S. Wang, G. L. Ye, W. J. Zhang, X. Chen, S. Cong, D. Zhou, H. Li,

J. Li, G. Zou, W. W. Shang, J. Jiang and Y. Luo, *Natl. Sci. Rev.*, 2022, **9**, nwac190, DOI: 10.1093/NSR/NWAC190.

14 X. Wei, X. Cui, N. Cheng, X. Wang, X. Zhang, S. Huang, P. Xie, J. Xu, Y. Chen, M. Zhang, Y. Jiang and W. Han, *ChatIE: Zero-Shot Information Extraction via Chatting with ChatGPT*, 2024, https://arxiv.org/abs/2302.10205.

15 T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever and D. Amodei, *Proceedings of the 34th International Conference on Neural Information Processing Systems*, 2020, DOI: 10.48550/arXiv.2005.14165.

16 M. Luo, X. Xu, Y. Liu, P. Pasupat and M. Kazemi, *In-context Learning with Retrieved Demonstrations for Language Models: A Survey*, 2024, https://arxiv.org/abs/2401.11624.

17 A. Kong, S. Zhao, H. Chen, Q. Li, Y. Qin, R. Sun, X. Zhou, E. Wang and X. Dong, *Better Zero-Shot Reasoning with Role-Play Prompting*, 2024, https://arxiv.org/abs/2308.07702.

18 J. Wei, X. Wang, D. Schuurmans, M. Bosma, b. ichter, F. Xia, E. Chi, Q. V. Le and D. Zhou, *Adv. Neural Inf. Process Syst.*, 2022, 24824–24837, DOI: 10.5555/3600270.3602070.

19 Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang and H. Wang, *Retrieval-Augmented Generation for Large Language Models: A Survey*, 2024, https://arxiv.org/abs/2312.10997.

20 Y. Luo, S. Bag, O. Zaremba, A. Cierpka, J. Andreo, S. Wuttke, P. Friederich and M. Tsotsalas, *Angew. Chem., Int. Ed.*, 2022, **61**, e202200242, DOI: 10.1002/anie.202200242.

21 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, DOI: 10.1021/acs.jcim.1c00284.

22 M. Dreger, K. Malek and M. Eikerling, Large language models for knowledge graph extraction from tables in materials science, *Digital Discovery*, 2025, **4**(5), 1221–1231, DOI: 10.1039/d4dd00362d.

23 S. X. Leong, S. Pablo-garcía, B. Wong and A. Aspuru-Guzik, MERMaid: Universal multimodal mining of chemical reactions from PDFs using vision-language models, *ChemRxiv*, 2025, preprint, DOI: 10.26434/chemrxiv-2025-8z6h2.

24 J. Bai, S. Mosbach, C. J. Taylor, D. Karan, K. F. Lee, S. D. Rihm, J. Akroyd, A. A. Lapkin and M. Kraft, *Nat. Commun.*, 2024, **15**, 1–14, DOI: 10.1038/s41467-023-44599-9.

25 A. Eibeck, M. Q. Lim and M. Kraft, *Comput. Chem. Eng.*, 2019, **131**, 106586, DOI: 10.1016/j.compchemeng.2019.106586.

26 L. Pascazio, S. Rihm, A. Naseri, S. Mosbach, J. Akroyd and M. Kraft, *J. Chem. Inf. Model.*, 2023, **63**, 6569–6586, DOI: 10.1021/ACS.JCIM.3C00820.

27 M. Kraft and C. CARES, *The World Avatar*, C. M. G. at University of Cambridge and C. M. C. Ltd, 2025, https://github.com/cambridge-cares/TheWorldAvatar, last accessed April 22, 2025.

28 J. Bai, S. D. Rihm, A. Kondinski, F. Saluz, X. Deng, G. Brownbridge, S. Mosbach, J. Akroyd and M. Kraft, *Digital Discovery*, 2025, **4**(8), 2123–2135, DOI: 10.1039/D5DD00069F.

29 S. D. Rihm, D. N. Tran, A. Kondinski, L. Pascazio, F. Saluz, X. Deng, S. Mosbach, J. Akroyd and M. Kraft, *Data-Centric Eng*, 2025, **6**, e22, DOI: 10.1017/dce.2025.12.

30 L. Hawizy, D. M. Jessop, N. Adams and P. Murray-Rust, *J. Cheminf.*, 2011, **3**, 1–13, DOI: 10.1186/1758-2946-3-17/TABLES/6.

31 A. J. Hammer, A. I. Leonov, N. L. Bell and L. Cronin, *JACS Au*, 2021, **1**, 1572–1587, DOI: 10.1021/JACSAU.1C00303.

32 P. Murray-Rust, H. S. Rzepa and M. Wright, *New J. Chem.*, 2001, **25**, 618–634, DOI: 10.1039/B008780G.

33 H. Bär, R. Hochstrasser and B. Papenfuß, *J. Lab. Autom.*, 2012, **17**, 86–95, DOI: 10.1177/2211068211424550.

34 J. Bai, L. Cao, S. Mosbach, J. Akroyd, A. A. Lapkin and M. Kraft, *JACS Au*, 2022, **2**, 292–309, DOI: 10.1021/jacsau.1c00438.

35 T. R. Gruber, *Knowledge Acquisition*, 1993, **5**, 199–220, DOI: 10.1006/knac.1993.1008.

36 K. K. Breitmann, M. A. Casanova and W. Truszkowski, in *Ontology in Computer Science*, Springer London, 2007, pp. 17–34, DOI: 10.1007/978-1-84628-710-7_2.

37 K. Degtyarenko, J. Hastings, P. de Matos and M. Ennis, *Curr. Protoc. Bioinf.*, 2009, **26**, 14.9.1–14.9.20, DOI: 10.1002/0471250953.bi1409s26.

38 C. Batchelor, *RXNO: reaction ontologies*, 2023, https://github.com/rsc-ontologies/rxno, last accessed April 22, 2025.

39 P. Strömert, J. Hunold, A. Castro, S. Neumann and O. Koepler, *Pure Appl. Chem.*, 2022, **94**, 605–622, DOI: 10.1515/PAC-2021-2007.

40 V. K. Chaudhri, C. Baru, N. Chittar, X. L. Dong, M. Genesereth, J. Hendler, A. Kalyanpur, D. B. Lenat, J. Sequeda, D. Vrandečić and K. Wang, *AI Mag.*, 2022, **43**, 17–29, DOI: 10.1002/AAAI.12033.

41 J. Akroyd, S. Mosbach, A. Bhave and M. Kraft, *Data-Centric Eng*, 2021, **2**, e14, DOI: 10.1017/dce.2021.10.

42 M. Q. Lim, X. Wang, O. Inderwildi and M. Kraft, The World Avatar - A world model for facilitating interoperability, in *Intelligent Decarbonisation*, Lecture Notes in Energy, Springer, 2022, vol. 86, pp. 39–53, DOI: 10.1007/978-3-030-86215-2_4.

43 D. Brickley and R. V. Guha, *RDF Schema 1.1*, 2014, https://www.w3.org/TR/2014/REC-rdf-schema-20140225/, last accessed April 22, 2025.

44 W3C OWL Working Group, *OWL 2 Web Ontology Language*, 2012, https://www.w3.org/TR/2012/REC-owl2-overview-20121211/, last accessed April 22, 2025.

45 S. Harris and A. Seaborne, *SPARQL 1.1 Query Language*, 2013, https://www.w3.org/TR/sparql11-query/, last accessed April 22, 2025.

46 R. Grishman, *IEEE Intell. Syst.*, 2015, **30**, 8–15, DOI: 10.1109/MIS.2015.68.

47 D. Papakyriakou and I. S. Barbounakis, *Int. J. Comput. Appl.*, 2022, **183**, 975–8887, DOI: 10.5120/ijca2022921884.

48 M. Krallinger, O. Rabal, A. Lourenço, J. Oyarzabal and A. Valencia, *Chem. Rev.*, 2017, **117**, 7673–7761, DOI: 10.1021/ACS.CHEMREV.6B00851.

49 M. Schilling-Wilhelmi, M. Ríos-García, S. Shabih, M. V. Gil, S. Miret, C. T. Koch, J. A. Márquez and K. M. Jablonka, From text to insight: large language models for chemical data extraction, *Chem. Soc. Rev.*, 2025, **54**(3), 1125–1150, DOI: 10.1039/D4CS00913D.

50 Z. Zheng, O. Zhang, H. L. Nguyen, N. Rampal, A. H. Alawadhi, Z. Rong, T. Head-Gordon, C. Borgs, J. T. Chayes and O. M. Yaghi, *ACS Cent. Sci.*, 2023, **9**, 2161–2170, DOI: 10.1021/ACSCENTSCI.3C01087.

51 K. M. Jablonka, P. Schwaller, A. Ortega-Guerrero and B. Smit, *Nat. Mach. Intell.*, 2024, **6**, 161–169, DOI: 10.1038/s42256-023-00788-1.

52 D. Xu, W. Chen, W. Peng, C. Zhang, T. Xu, X. Zhao, X. Wu, Y. Zheng, Y. Wang and E. Chen, *Front. Comput. Sci.*, 2024, **18**, 186357, DOI: 10.1007/s11704-024-40555-y.

53 M. X. Liu, F. Liu, A. J. Fiannaca, T. Koo, L. Dixon, M. Terry and C. J. Cai, *Extended Abstracts of the CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–9, DOI: 10.1145/3613905.3650756.

54 P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal and A. Chadha, *A Systematic Survey of Prompt Engineering in Large Language Models: Techniques and Applications*, 2025, https://arxiv.org/abs/2402.07927.

55 D. Kepel and K. Valogianni, *Autonomous Prompt Engineering in Large Language Models*, 2024, https://arxiv.org/abs/2407.11000.

56 X. Chen and X. Wan, *Evaluating, Understanding, and Improving Constrained Text Generation for Large Language Models*, 2024, https://arxiv.org/abs/2310.16343.

57 L.-P. Meyer, C. Stadler, J. Frey, N. Radtke, K. Junghanns, R. Meissner, G. Dziwis, K. Bulert and M. Martin, *First Working conference on Artificial Intelligence Development for a Resilient and Sustainable Tomorrow*, 2024, pp. 103–115, DOI: 10.1007/978-3-658-43705-3_8.

58 J. Dagdelen, A. Dunn, S. Lee, N. Walker, A. S. Rosen, G. Ceder, K. A. Persson and A. Jain, *Nat. Commun.*, 2024, **15**, 1418, DOI: 10.1038/s41467-024-45563-x.

59 M. Pokrass, *OpenAI: Introducing Structured Outputs in the API*, 2024, https://openai.com/index/introducing-structured-outputs-in-the-api/, last accessed April 22, 2025.

60 G. K. Q. Monfardini, J. S. Salamon and M. P. Barcellos, *Conceptual Modeling*, 2023, pp. 45–64, DOI: 10.1007/978-3-031-47262-6_3.

61 M. J. Prakash, Y. Zou, S. Hong, M. Park, M.-P. N. Bui, G. H. Seong and M. S. Lah, *Inorg. Chem.*, 2009, **48**, 1281–1283, DOI: 10.1021/ic802382p.

62 F. Giasson and B. D'Arcu, *The Bibliographic Ontology*, 2016, https://www.dublincore.org/specifications/bibo/, last accessed April 22, 2025.

63 H. Rijgersberg, M. Wigham and J. L. Top, *Adv. Eng. Inform.*, 2011, **25**, 276–287, DOI: 10.1016/J.AEI.2010.07.008.

64 L. Cronin and Cronin Group at University of Glasgow, *XDL Documentation*, 2022, https://croningroup.gitlab.io/chemputer/xdl.

65 C. Batchelor, *CHMO*, 2023, https://github.com/rsc-ontologies/rsc-cmo, last accessed April 22, 2025.

66 C. J. Date, *SIGMOD Rec.*, 1982, **13**, 18–29, DOI: 10.1145/984514.984515.

67 X. Liu, J. Wang, J. Sun, X. Yuan, G. Dong, P. Di, W. Wang and D. Wang, *Prompting Frameworks for Large Language Models: A Survey*, 2023, https://arxiv.org/abs/2311.12785.

68 M. Salehpour and J. G. Davis, *The Effects of Different JSON Representations on Querying Knowledge Graphs*, 2020, https://arxiv.org/abs/2004.04286.

69 T. Cormen, C. Leiserson, R. Rivest and C. Stein, *Introduction to Algorithms*, 4th edn, MIT Press, 2022.

70 J. H. Caufield, H. Hegde, V. Emonet, N. L. Harris, M. P. Joachimiak, N. Matentzoglu, H. Kim, S. Moxon, J. T. Reese, M. A. Haendel, P. N. Robinson and C. J. Mungall, *Bioinformatics*, 2024, **40**, btae104, DOI: 10.1093/bioinformatics/btae104.

71 A. C. Vaucher, F. Zipoli, J. Geluykens, V. H. Nair, P. Schwaller and T. Laino, *Nat. Commun.*, 2020, **11**, 1–11, DOI: 10.1038/s41467-020-17266-6.

72 H. Lyu, Z. Ji, S. Wuttke and O. M. Yaghi, *Chem*, 2020, **6**, 2219–2241, DOI: 10.1016/j.chempr.2020.08.008.