SynKB: Semantic Search for Synthetic Procedures

Fan Bai[♣] **Alan Ritter**[♣] **Peter Madrid**[♠] **Dayne Freitag**[◇] **John Niekrasz**[◇]
♣ School of Interactive Computing, Georgia Institute of Technology
♠ Biosciences Division, SRI International
◇ Artificial Intelligence Center, SRI International
{fan.bai, alan.ritter}@cc.gatech.edu
{peter.madrid, daynefreitag, john.niekrasz}@sri.com

Abstract

In this paper we present SynKB,[1] an open-source, automatically extracted knowledge base of chemical synthesis protocols. Similar to proprietary chemistry databases such as Reaxsys, SynKB allows chemists to retrieve structured knowledge about synthetic procedures. By taking advantage of recent advances in natural language processing for procedural texts, SynKB supports more flexible queries about reaction conditions, and thus has the potential to help chemists search the literature for conditions used in relevant reactions as they design new synthetic routes. Using customized Transformer models to automatically extract information from 6 million synthesis procedures described in U.S. and EU patents, we show that for many queries, SynKB has higher recall than Reaxsys, while maintaining high precision. We plan to make SynKB available as an open-source tool; in contrast, proprietary chemistry databases require costly subscriptions.[2]

1 Introduction

Commercial chemistry databases, such as Reaxsys[3] are invaluable tools for chemists, who issue structured SQL-like queries to retrieve precise information about chemical reactions described in the literature. Large, high-quality datasets are also crucial for synthetic route planning (Klucznik et al., 2018), automation (Coley et al., 2019b; Collins et al., 2020), and machine learning approaches to retrosynthesis (Coley et al., 2019a). In addition to proprietary, manually curated databases such as Reaxys, recent work has begun to use automatically extracted data from reactions described in patents (Tetko et al., 2020), however existing databases are limited to basic reaction information, and do not include important details such as concentrations or order of additions (Coley et al., 2019b). The lack of high-quality data has been identified as a key challenge in developing recommendation models for reaction conditions (Struble et al., 2020).

In this paper, we present SynKB, a working system that demonstrates the application of modern NLP methods to extract large quantities of structured information about chemical synthesis procedures from text. SynKB has a number of advantages with respect to existing chemistry databases such as Reaxys: (1) We show that by automatically extracting information from millions of synthesis procedures described in U.S. and European patents using state-of-the-art NLP methods, we can achieve significantly higher recall than existing chemistry databases while maintaining high precision. In §3, we demonstrate SynKB's coverage is complementary to Reaxys; see Figure 2 for details. (2) SynKB's novel graph search supports better coverage of reaction conditions than existing chemistry databases; this includes concentrations, reaction times, order of the addition of reagents, catalysts, etc. (3) We will make SynKB available as open-source software on publication, in contrast, most existing chemistry databases are proprietary, with the notable exception of Lowe (2017), which we compare to in §3.

We have built an online demo, which can be viewed at the following URL: https://tinyurl.com/synkb. We will also release the source code and patent-based extractions used to build SynKB on publication.

2 SynKB

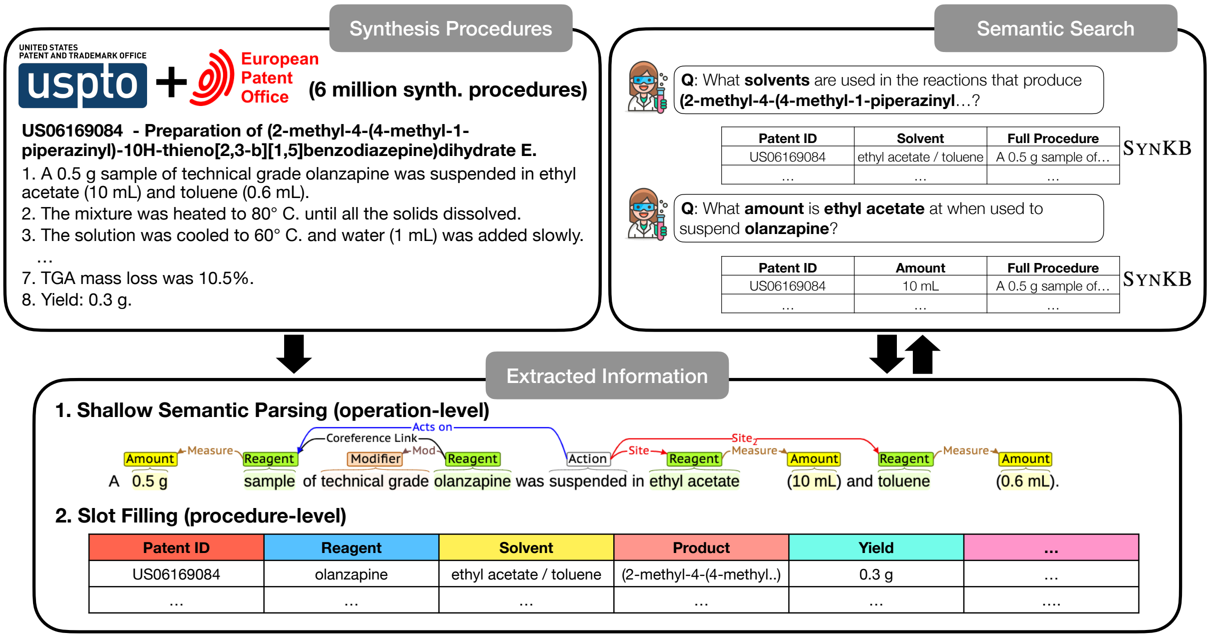
SynKB is an open-source system that allows chemists to perform structured queries over large corpora of synthesis procedures. In this section, we present each component of SynKB, as illustrated in Figure 1. Our corpus collection is first presented in §2.1. Section 2.2 describes how a corpus of six

[1] Demo URL: https://tinyurl.com/synkb
Introduction video: https://screencast-o-matic.com/watch/c3jVQsVZwOV

[2] Code: https://github.com/bflashcp3f/SynKB.

[3] https://www.elsevier.com/solutions/reaxys

Proceedings of the The 2022 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, pages 311 - 318
December 7-11, 2022 ©2022 Association for Computational Linguistics



US06169084 - Preparation of (2-methyl-4-(4-methyl-1-piperazinyl)-10H-thieno[2,3-b][1,5]benzodiazepine)dihydrate E.

1. A 0.5 g sample of technical grade olanzapine was suspended in ethyl acetate (10 mL) and toluene (0.6 mL).
2. The mixture was heated to 80° C. until all the solids dissolved.
3. The solution was cooled to 60° C. and water (1 mL) was added slowly.

...

7. TGA mass loss was 10.5%.
8. Yield: 0.3 g.

Q: What **solvents** are used in the reactions that produce **(2-methyl-4-(4-methyl-1-piperazinyl**...?

Patent ID	Solvent	Full Procedure
US06169084	ethyl acetate / toluene	A 0.5 g sample of...
...	...	...

Q: What **amount** is **ethyl acetate** at when used to suspend **olanzapine**?

Patent ID	Amount	Full Procedure
US06169084	10 mL	A 0.5 g sample of...
...	...	...

1. Shallow Semantic Parsing (operation-level)

A 0.5 g sample of technical grade olanzapine was suspended in ethyl acetate (10 mL) and toluene (0.6 mL).

2. Slot Filling (procedure-level)

Patent ID	Reagent	Solvent	Product	Yield	...
US06169084	olanzapine	ethyl acetate / toluene	(2-methyl-4-(4-methyl..)	0.3 g	...
...	...	...	...	...	....

Figure 1: Overview of our semantic search system SYNKB, which searches over 6 million chemical synthesis procedures collected from patents. Users can enter structured queries to retrieve procedures concerning procedure-level or operation-level information.

million procedures is annotated with sentence-level action graphs, in addition to protocol-level slots relevant to chemical reactions, including starting materials, solvents, reaction products, yields, etc. After automatically annotating and indexing, we experiment with the semantic search capabilities enabled by SYNKB in §2.3.

2.1 Corpus Collection

We extract structured representations of synthetic protocols from a corpus of chemical patents (Bai et al., 2021), which includes over six million chemical synthesis procedures extracted from around 300k U.S. and European patents (written in English). The U.S. portion of this corpus comes from an open-source corpus of chemical synthesis procedures (Lowe, 2017), which covers 2.4 million synthetic procedures extracted from U.S. patents (USPTO[4], 1976-2016). For the European portion, we apply the Lowe (2017) reaction identification pipeline to European patents. Specifically, we download patents from EPO[5] (1978-2020) as XML files and select patents containing the IPC (International Patent Classification) code 'C07' for processing as they are in the category of organic chemistry. Next, the synthesis procedure identifier developed by Lowe (2012), a trained Naive Bayes classifier, is applied to the *Description* section of all selected patents. As a result, we obtain another 3.7 million procedures from European patents.

2.2 Extracting Reaction Details from Synthetic Procedures

To facilitate semantic search, we automatically annotate the corpus of 6 million synthetic procedures described above with semantic action graphs (Kulkarni et al., 2018) in addition to chemical reaction slots (Nguyen et al., 2020) using Transformer models that are pre-trained on a large corpus of scientific procedures (Bai et al., 2021).

Shallow Semantic Parsing. We first perform sentence-level annotation, where each step in the procedure is annotated with a semantic graph (Tamari et al., 2021). Nodes in the graph are experimental operations and their typed arguments, whereas labeled edges specify relations between the nodes (see the example shallow semantic parse in Figure 1). Here we use the CHEMSYN framework (Bai et al., 2021), which covers 24 types of nodes (such as *Action*, *Reagent*, *Amount*, *Equipment*, etc.) and 17 edge types (e.g. *Acts-on* and

[4] https://www.uspto.gov/learning-and-resources/bulk-data-products

[5] https://www.epo.org/searching-for-patents/data/bulk-data-sets.html

Measure). With these annotated semantic graphs, users can search for operation-level information, for example, the amount of `DMF` when used as a solvent to dissolve `HATU` (this will be further discussed in §3). Following Tamari et al. (2021), we split semantic graph annotation into two sub-tasks, Mention Identification (MI) for node prediction and Argument Role Labeling (ARL) for edge prediction. We use the same fine-tuning architectures as in Tamari et al. (2021). Models are fine-tuned on the CHEMSYN corpus, which consists of 992 chemical synthesis procedures extracted from patents, and the resulting performance (averages across five random seeds) is shown in Table 1. We select model checkpoints via the Dev set performance out of five random seeds, and use the selected checkpoint for inference on our 6 million synthetic procedures.

Slot Filling. In the second task, we annotate procedures from a protocol perspective, i.e., identifying key entities playing certain roles in a protocol, which can be queried in a slot-based search. We use the CHEMU training corpus proposed in Nguyen et al. (2020). This dataset includes 10 pre-defined slot types concerning chemical compounds and related entities in chemical synthesis processes such as *Starting Material*, *Solvent*, and *Product*. Similar to the Mention Identification task, we treat Slot Filling as a sequence tagging problem. However, the input in Slot Filling is the entire protocol, rather than a single sentence, as in mention identification. We fine-tune models on the CHEMU dataset (see Table 1 for results), and then run inference on the chemical patent corpus using the learned model.

ProcBERT. We use ProcBERT (Bai et al., 2021), a BERT-based model that is pre-trained on in-domain data (scientific protocols), as the backbone for all of our models, and develop task-specific fine-tuning architectures on top of it. The comparison between ProcBERT and other pre-trained models is presented in Table 1. Because ProcBERT is pre-trained using in-domain data, we find that it outperforms both BERT$_{\text{large}}$ (Devlin et al., 2019) and SciBERT (Beltagy et al., 2019) on all three tasks.

Annotation Task	**Dataset**	**Pre-trained Model**		
		BERT$_{\text{large}}$	SciBERT	ProcBERT
Mention Identification	CHEMSYN	$95.26_{0.1}$	$95.82_{0.2}$	$\mathbf{95.97}_{0.2}$
Argument Role Labeling		$92.87_{0.5}$	$93.27_{0.2}$	$\mathbf{93.57}_{0.2}$
Slot Filling	CHEMU	$95.10_{0.2}$	$95.63_{0.1}$	$\mathbf{96.19}_{0.1}$

Table 1: Test set F_1 scores of fine-tuned models for the three annotation tasks. These numbers, averages across five random seeds with standard deviations as subscripts, are taken from our previous work Bai et al. (2021). Models using ProcBERT for contextual embeddings perform the best on all three tasks and are used for automatic annotations on six million synthesis procedures to construct SYNKB.

	SYNKB **(ours)**	**USPTO-Lowe**	**Reaxys**
License	Open source	Open source	Subscription
# Procedures (mill.)	6	2.4	57
# Entity Types	24	8	10
# Relation Types	17	-	-
Annotation	Automatic	Automatic	Manual

Table 2: Comparison between our SYNKB and two performant databases. Our SYNKB provides more fine-grained annotations (more entity types and unique relation annotations) than the other two systems and covers more procedures than USPTO-Lowe, a database built using the largest open-source synthesis procedure corpus (Lowe, 2017).

2.3 Semantic Search

SYNKB offers search modalities specific to each of these two forms of annotation, i.e., semantic action graphs and chemical reaction slots, along with features designed to support practical use. The first type of query supported by SYNKB is **semantic graph search**, which allows users to search for synthesis procedures based on the semantic parse of the constituent operations. We adapt the graph query formalism proposed originally for syntactic dependencies in Valenzuela-Escárcega et al. (2020).[6] Formally, the input query $G = (V, E)$ is a labeled directed graph. Each node $v_i \in V$ is specified as a set of constraints on matching entities (a single or multi-token span). For example, users can specify the node as `DMF` or `[word=DMF]`, which triggers an exact match on entity mentions containing the word "DMF". They can also constrain the entity type of the node using the expression `[entity=Type]`.[7] Moreover, nodes can be named `captures` when surrounded with `(?<name>...)`, e.g., the query `(?<solvent> DMF)` captures `DMF` as the `solvent`. As for the edge $e = (v_i, v_j, l) \in E$, we need to specify the direction and the semantic relation. Considering the query `(?<solvent> DMF) >measure (?<amount> 1 ml)`, it represents a se-

[6] We refer readers to the tutorial of Odinson query language for more details of this graph query formalism.

[7] We store entity labels with the BIO tagging scheme, so users can match a single token entity with the expression `[entity=B-Type]` and a multi-token entity with the expression `[entity=B-Type][entity=I-Type]*`.

mantic graph containing two entity nodes captured as `solvent` and `amount`, and an edge signaling the `measure` relation and its direction (from `solvent` to `amount`).

In addition, SYNKB supports **slot-based search**, which presents a structured search interface, with entries corresponding to CHEMU slots. A keyword entered into any entry restricts the retrieved set to procedures where the extracted slot contains the indicated keyword. Like the graph search, this returns a set of tuples with elements named with matching slots and containing the matching entity strings. The special token "?" can be used to match *any* slot value.

As for the implementation, the semantic graph search module is powered by Odinson (Valenzuela-Escárcega et al., 2020), an open-source Lucene-based query engine. Odinson pre-indexes the annotated corpus by generating the inverted index for each procedure. Given an input query, Odinson performs a two-step matching process, where it first examines the node constraints via the inverted index; if this step works well, the semantic relations will be verified in the second step. The two-step matching process improves the speed of Odinson, and thus enables interactive querying. As for the slot-based search, it is supported by Elasticsearch[8] with the exception that, when users perform both types of search at the same time, we use the metadata search feature of Odinson for slot filters (we store slot values as metadata) to improve the system's response speed.

3 Empirical Comparison

In §2, we described the design and implementation of SYNKB including the underlying models, data preparation, and semantic search features. To demonstrate the utility of SYNKB for assisting chemists to search the literature for reaction details, we now evaluate its search features on ten example questions (Q1-Q10 in Table 3), which were collected from synthetic chemists working on the design of new synthesis protocols. In §3.1, we evaluate the slot-based search module of SYNKB and compare it with two existing databases which provides similar search features. In §3.2, we demonstrate how to use our novel semantic graph search module to answer operation-specific questions and evaluate its retrieved answers and procedures.

3.1 Slot-based Search Evaluation

We benchmark the slot-based search module of SYNKB against Reaxys, one of the leading proprietary chemistry databases, and USPTO-Lowe, an automatically extracted database built using a large open-source synthesis procedure corpus (Lowe, 2017). Below, we first introduce these two databases briefly, and then evaluate the results of all three systems on the chemist-proposed questions.

3.1.1 Chemistry Databases

The first database we compare with is **Reaxys**, a web-based commercial chemistry database, which contains comprehensive chemistry data, including chemical properties, compound structures, etc. What particularly interests us in Reaxys is that it contains expert-curated reaction procedures collected from extensive published literature such as chemistry-related patents and periodicals.[9] Also, key experimental entities in those reaction procedures, like participating reagents and reaction temperature, are specified. Thus, similar to our slot-based search, Reaxys allows users to search for reaction procedure information by applying text filters. Users can use its *Query Builder* module to specify multiple chemical reaction-specific filters, and then Reaxys returns all matched reaction procedures along with identified entities in those procedures, which are available for download.

Apart from Reaxys, we also build a database using **USPTO-Lowe** (Lowe, 2017), the largest available open-source chemical synthesis procedure corpus as introduced in §2.1, for comparison. Similar to our SYNKB, this corpus includes automatic annotations of experimental entities on 2.4 million contained reaction procedures.[10] However, our SYNKB provides more fine-grained and comprehensive entity annotations (see Table 2 for the statistics of three experimented databases), and also annotates the relations between extracted entities, which constitute semantic graphs (§2.2) enabling operation-specific semantic graph search. As for the implementation, we load USPTO-Lowe's entity annotation into Elasticsearch, so this customized database can be used in the same way as the slot-based search module of our SYNKB.

[8] https://www.elastic.co/elasticsearch/

[9] https://www.elsevier.com/solutions/reaxys/features-and-capabilities/content

[10] https://www.nextmovesoftware.com/leadmine.html

System	Input Query	# Proce.	# Ans.	Ans. Prec.
	Slot-based Search			
Q1 - What are the **solvents** used for reactions containing the reagent **triphosgene**?				
Reaxys	`{"reagent":"triphosgene"}`	35	7	**100%**
USPTO-Lowe	`{"reagent":"triphosgene", "solvent":"?"}`	3157	104	90%
SynKB		**7184**	**127**	94%
Q2 - What are the **yields** (percent) of reactions producing **(5-Methylpyrimidin-2-yl)methanol**?				
Reaxys	`{"product":"(5-Methylpyrimidin-2-yl)methanol"}`	1	1	100%
USPTO-Lowe	`{"product":"(5-Methylpyrimidin-2-yl)methanol", "yield (percent)":"?"}`	1	1	100%
SynKB		1	1	100%
Q3 - What are the **products** of reactions containing the reagent **trimethylsilyldiazomethane**?				
Reaxys	`{"reagent":"trimethylsilyldiazomethane"}`	438	75	**100%**
USPTO-Lowe	`{"reagent":"trimethylsilyldiazomethane", "product":"?"}`	517	335	98%
SynKB		**1033**	**708**	96%
Q4 - What are the **products** of reactions containing the reagent **chlorosulfonic acid** and the solvent **chlorobenzene**?				
Reaxys	`{"reagent":"chlorosulfonic acid"} AND {"solvent":"chlorobenzene"}`	**148**	**65**	100%
USPTO-Lowe	`{"reagent":"chlorosulfonic acid", "solvent":"chlorobenzene", "product":"?"}`	6	2	100%
SynKB		9	4	100%
Q5 - What are the **reaction times** for reactions using reagent **CDI (carbonyldiimidazole)**?				
Reaxys	`{"reagent":"CDI"} OR {"reagent":"carbonyldiimidazole"}`	93	24	**100%**
USPTO-Lowe	`{"reagent": "CDI OR carbonyldiimidazole", "reaction time":"?"}`	3722	339	100%
SynKB		**6377**	**511**	94%
Q6 - What are the **reaction temperatures** for reactions containing reagent **trifluoromethanesulfonic acid**?				
Reaxys	`{"reagent":"trifluoromethanesulfonic acid"}`	104	3	**100%**
USPTO-Lowe	`{"reagent":"trifluoromethanesulfonic acid", "temperature":"?"}`	727	124	100%
SynKB		**1937**	**243**	98%
	Semantic Graph Search			
Q7 - What are the **reagents** used to dilute **plasma**?				
SynKB	`plasma <acts-on diluted >using (?<reagent> [entity=B-Reagent][entity=I-Reagent]*)`	24	16	100%
Q8 - What is the **pH** of a solution after being titrated with **NaOH**?				
SynKB	`(?<ph> [entity=B-pH][entity=I-pH]+) <setting titrated >using NaOH`	39	21	95%
Q9 - What are the common **pore sizes** of **PTFE filters**?				
SynKB	`PTFE filter >measure (?<pore_size> [entity=B-Generic-Measure][entity=I-Generic-Measure]*)`	183	39	92%
Q10 - What **molar concentration** is the reagent **HATU** at when **dissolved** in the solvent **DMF**?				
SynKB	`HATU >measure (?<mole> [] [word=mmol|word=mol]) []{1,10} DMF >measure (?<volume> [] [word=ml|word=l])`	447	289	100%

Table 3: Search queries and resulting performance on 10 chemist-proposed questions for Reaxys, USPTO-Lowe, and SynKB (ours). **# Proc.** is the number of returned procedures containing valid answers, and **# Ans.** refers to the number of distinct answer slots or captures in these procedures. The first six questions (Q1-Q6) are answerable for all three databases as they only require entity annotation while the last four questions (Q7-Q10) can only be answered by our SynKB using our unique semantic action graph annotation. SynKB consistently shows better recall than two compared databases while being highly accurate.

3.1.2 Comparison with Examples

We now compare three systems on six questions that were proposed by chemists (Q1-Q6) as these questions only require annotations on experimental entities and thus can be answered in all three systems. For example, Q1 ("What solvents are used in reactions involving triphosgene?") can be answered by the SynKB query `{"reagent":"triphosgene", "solvent":"?"}`, as *reagent* and *solvent* are query-able ChEMU slots. Similarly, for Reaxys, experimental entities are specified for corresponding text filters.

We evaluate the output of each system from two perspectives: 1) recall, which is measured by the number of returned procedures containing valid answers and the number of distinct answer slots or captures in these procedures; and 2) precision, the proportion of correct answers among all predicted answers. In cases where the number of answers exceeds 50, we sample 50 answers from the full set to estimate precision.

The search queries and performance on each question for the three systems are shown in Table 3. We can see that, SynKB consistently retrieves a larger number of relevant procedures and answers than Reaxys (5 out of 6 questions) while maintaining high precision. USPTO-Lowe, which uses a rule-based annotation model, shows competitive performance on precision but trails behind our SynKB in terms of recall for all 6 questions. This comparison clearly shows the strength of our system: by leveraging state-of-the-art NLP for chemical synthesis procedures (Bai et al., 2021), we can provide chemists with abundant information, which is non-proprietary and delivered with high precision. Furthermore, we plot the Venn diagram (Figure 2) over the retrieved answers, which shows the percentage of unique and shared answers for each system out of all retrieved answers (we do macro-average across six questions.) Interestingly, only 18.1% of retrieved answers are shared among all three systems, and both our SYNKB and Reaxys contain a large number of unique answers, which take 31.5% and 17.4% of retrieved answers respectively. This shift in answer distribution suggests that our open-source SYNKB can be a good complement to proprietary chemistry databases like Reaxys, and it is better for users to use both of them if possible instead of choosing one over the other.

3.2 Semantic Graph Search Evaluation

We evaluate our novel semantic graph search on four operation-specific questions (Q7-Q10). Unlike the six questions introduced above, these questions place constraints on the relations between mentioned entities, and thus are not answerable for Reaxys and USPTO-Lowe (due to the lack of relation annotation). For instance, to answer Q7 “What are the reagents used to dilute `plasma`?”, a system needs to first locate the particular operation in a procedure where `plasma` is diluted, and then identify the reagent, which facilitates this dilution operation. This whole process can be realized in our semantic graph search module. Concretely, the graph-based query we use for Q7 is: “`plasma <acts-on diluted >using (?<reagent> [entity=B-Reagent][entity=I-Reagent]*)`”, which matches procedures containing “`plasma`” and “`diluted`” connected in the same semantic graph and returns used reagents in the form of named captures. We evaluate the performance of the semantic graph search module by manually inspecting predicted answers (randomly sampling 50 answers for Q10), and show results in Table 3. Similar to the findings in the slot-based search evaluation, SYNKB shows good coverage while maintaining high precision.

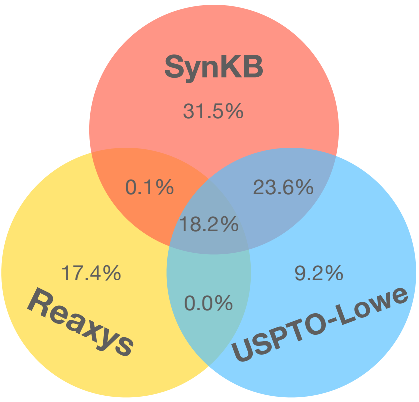


Figure 2: Venn diagram on the answer distribution of six slot-based search questions (macro-average) for all three databases. We can see that both our SYNKB and Reaxys cover high percentage of unique answers, suggesting that users should use them together if possible.

4 Related Work

Lowe (2012) was the first to develop a complete information extraction pipeline for chemical synthesis procedures, using a mostly rule-based approach. Subsequently, there have been several efforts to extract information from experimental procedures by either developing more performant extraction models (Vaucher et al., 2020; Guo et al., 2021) or designing extraction frameworks for other types of scientific literature, like wet-lab protocols (Kulkarni et al., 2018) and material science publications (Mysore et al., 2019; Kuniyoshi et al., 2020; Olivetti et al., 2020; O’Gorman et al., 2021). In this paper, we use the state-of-the-art NLP models for chemical synthesis procedures (Bai et al., 2021) to build the largest open-source knowledge base that searches synthetic procedure details. Our system is complementary to many proprietary chemistry databases, such as Reaxys, SciFinder[11], and Pistachio[12], in terms of contained information and search modalities.

Recent work has also developed slot-based classifiers to extract structured representations of events (from social media), supporting structured queries (Zong et al., 2020). In contrast, we present a semantic search system, which is customized for chemical synthesis procedures with specialized search features. In addition, recent work has explored *extractive search* systems (Ravfogel et al., 2021) that allow experts to specify syntactic pat-

[11] https://scifinder.cas.org

[12] https://www.nextmovesoftware.com/pistachio.html

terns, including syntactic structures of the input and capture slots. The graph-based queries in our SYNKB enable a similar capability in the domain of synthetic procedures, however SYNKB's queries are defined over semantic graphs that encode actions in synthetic protocols and associated semantic arguments.

5 Conclusion

In this paper, we present SYNKB, a system for large-scale extraction and querying of chemical synthesis procedures. SYNKB provides efficient searches against semantic action graphs and chemical reaction slots derived from 6 million synthesis procedures contained in chemical patents. A quantitative comparison with Reaxys, one of the leading commercial databases of reaction information, demonstrates the competence and versatility of our freely accessible system.

Ethical Considerations and Broader Impacts

Proprietary chemistry databases, such as Reaxys require costly subscriptions, limiting scientific inquiry for those who do not have the means to access this valuable source of information. In this paper, we presented an open-source semantic search system, SYNKB, which demonstrates state-of-the-art NLP methods can enable automatically extracted databases of synthetic procedure operational details that are competitive with Reaxys in terms of recall. We will make our code and data freely available.

The data contained in SYNKB is based on automatic extraction from both European and U.S. patents that are in the public domain. Our use complies with the terms of service of the U.S. Patent and Trademark Office and the European Patent Office.

Acknowledgements

We thank Wei Xu for helpful discussion and comments on a previous draft. This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under Contract No. HR001119C0108. The views, opinions, and/or findings expressed are those of the author(s) and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

References

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5002–5015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A Pretrained Language Model for Scientific Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China.

Connor W Coley, Wengong Jin, Luke Rogers, Timothy F Jamison, Tommi S Jaakkola, William H Green, Regina Barzilay, and Klavs F Jensen. 2019a. A graph-convolutional neural network model for the prediction of chemical reactivity. *Chemical science*, 10(2):370–377.

Connor W Coley, Dale A Thomas III, Justin AM Lummiss, Jonathan N Jaworski, Christopher P Breen, Victor Schultz, Travis Hart, Joshua S Fishman, Luke Rogers, Hanyu Gao, et al. 2019b. A robotic platform for flow synthesis of organic compounds informed by ai planning. *Science*, 365(6453):eaax1566.

Nathan Collins, David Stout, Jin-Ping Lim, Jeremiah P Malerich, Jason D White, Peter B Madrid, Mario Latendresse, David Krieger, Judy Szeto, Vi-Anh Vu, et al. 2020. Fully automated chemical synthesis: toward the universal synthesizer. *Organic Process Research & Development*, 24(10):2064–2077.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4171–4186, Minneapolis, Minnesota.

Jiang Guo, A Santiago Ibanez-Lopez, Hanyu Gao, Victor Quach, Connor W Coley, Klavs F Jensen, and Regina Barzilay. 2021. Automated chemical reaction extraction from scientific literature. *Journal of chemical information and modeling*.

Tomasz Klucznik, Barbara Mikulak-Klucznik, Michael P McCormack, Heather Lima, Sara Szymkuć, Manishabrata Bhowmick, Karol Molga, Yubai Zhou, Lindsey Rickershauser, Ewa P Gajewska, et al. 2018. Efficient syntheses of diverse, medicinally relevant targets planned by computer and executed in the laboratory. *Chem*, 4(3):522–532.

Chaitanya Kulkarni, Wei Xu, Alan Ritter, and Raghu Machiraju. 2018. An annotated corpus for machine reading of instructions in wet lab protocols. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational*

Linguistics: Human Language Technologies, Volume 2 (Short Papers), pages 97–106.

Fusataka Kuniyoshi, Kohei Makino, Jun Ozawa, and Makoto Miwa. 2020. Annotating and extracting synthesis process of all-solid-state batteries from scientific literature. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 1941–1950.

Daniel Lowe. 2017. Chemical reactions from US patents (1976-2016).

Daniel M. Lowe. 2012. Extraction of Chemical Structures and Reactions from the Literature (Doctoral Thesis).

Sheshera Mysore, Zachary Jensen, Edward Kim, Kevin Huang, Haw-Shiuan Chang, Emma Strubell, Jeffrey Flanigan, Andrew McCallum, and Elsa Olivetti. 2019. The materials science procedural text corpus: Annotating materials synthesis procedures with shallow semantic structures. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 56–64.

Dat Quoc Nguyen, Zenan Zhai, Hiyori Yoshikawa, Biaoyan Fang, Christian Druckenbrodt, Camilo Thorne, Ralph Hoessel, S. Akhondi, Trevor Cohn, Timothy Baldwin, and K. Verspoor. 2020. ChEMU: Named Entity Recognition and Event Extraction of Chemical Reactions from Patents. *Advances in Information Retrieval*, pages 572 – 579.

Elsa A Olivetti, Jacqueline M Cole, Edward Kim, Olga Kononova, Gerbrand Ceder, Thomas Yong-Jin Han, and Anna M Hiszpanski. 2020. Data-driven materials research enabled by natural language processing and information extraction. *Applied Physics Reviews*, 7(4).

Tim O'Gorman, Zach Jensen, Sheshera Mysore, Kevin Huang, Rubayyat Mahbub, Elsa Olivetti, and Andrew McCallum. 2021. Ms-mentions: Consistently annotating entity mentions in materials science procedural text. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1337–1352.

Shauli Ravfogel, Hillel Taub-Tabib, and Yoav Goldberg. 2021. Neural extractive search. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 210–217.

Thomas J Struble, Juan C Alvarez, Scott P Brown, Milan Chytil, Justin Cisar, Renee L DesJarlais, Ola Engkvist, Scott A Frank, Daniel R Greve, Daniel J Griffin, et al. 2020. Current and future roles of artificial intelligence in medicinal chemistry synthesis. *Journal of medicinal chemistry*, 63(16):8667–8682.

Ronen Tamari, Fan Bai, Alan Ritter, and Gabriel Stanovsky. 2021. Process-level representation of scientific protocols with interactive annotation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2190–2202, Online. Association for Computational Linguistics.

Igor V Tetko, Pavel Karpov, Ruud Van Deursen, and Guillaume Godin. 2020. State-of-the-art augmented nlp transformer models for direct and single-step retrosynthesis. *Nature communications*, 11(1):1–11.

Marco A. Valenzuela-Escárcega, Gus Hahn-Powell, and Dane Bell. 2020. Odinson: A fast rule-based information extraction framework. In *Proceedings of the 12th Language Resources and Evaluation Conference*, pages 2183–2191, Marseille, France. European Language Resources Association.

Alain C Vaucher, Federico Zipoli, Joppe Geluykens, Vishnu H Nair, Philippe Schwaller, and Teodoro Laino. 2020. Automated extraction of chemical synthesis actions from experimental procedures. *Nature communications*, 11(1):1–11.

Shi Zong, Ashutosh Baheti, Wei Xu, and Alan Ritter. 2020. Extracting a knowledge base of covid-19 events from social media. *arXiv preprint arXiv:2006.02567*.