Increasing the adoption of machine learning technologies by chemists with graphical and programmatic interfaces

Alain Vaucher[1,2] , Matteo Manica, Aleksandros Sobczyk, Teodoro Laino
[1]IBM Research Europe, Zurich, Switzerland
[2]National Center for Competence in Research – Catalysis, Zurich, Switzerland

@acvaucher

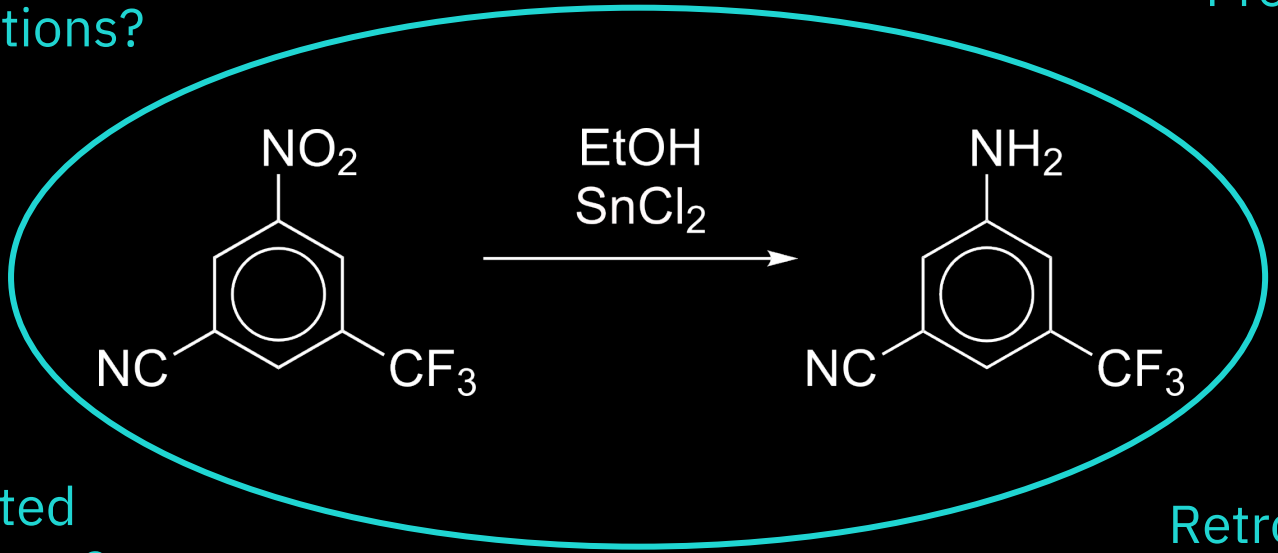
ACS Spring 2023
March 26, 2023

IBM **Research**

AI and chemical reactivity

Experimental conditions?

Product?



Related reactions?

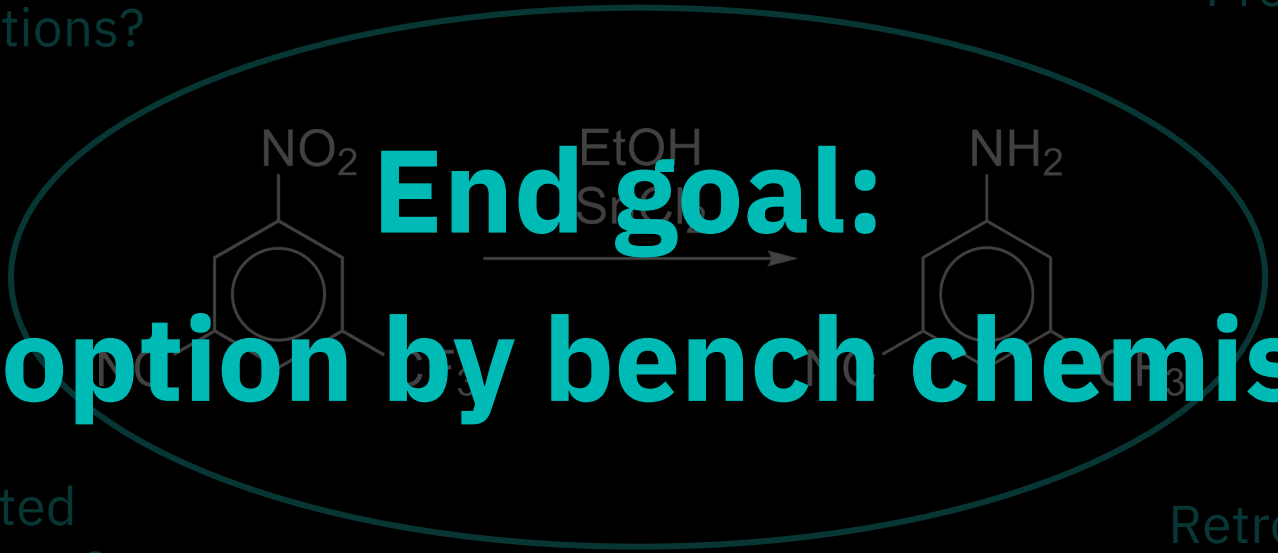
Retrosynthesis?

Yield?

AI and chemical reactivity

Experimental conditions?

Product?



End goal:

Adoption by bench chemists

Related reactions?

Retrosynthesis?

Yield?

Outline

1. AI models for chemistry
2. Building a user interface and API
3. Use cases and special considerations
4. Challenges and outlook

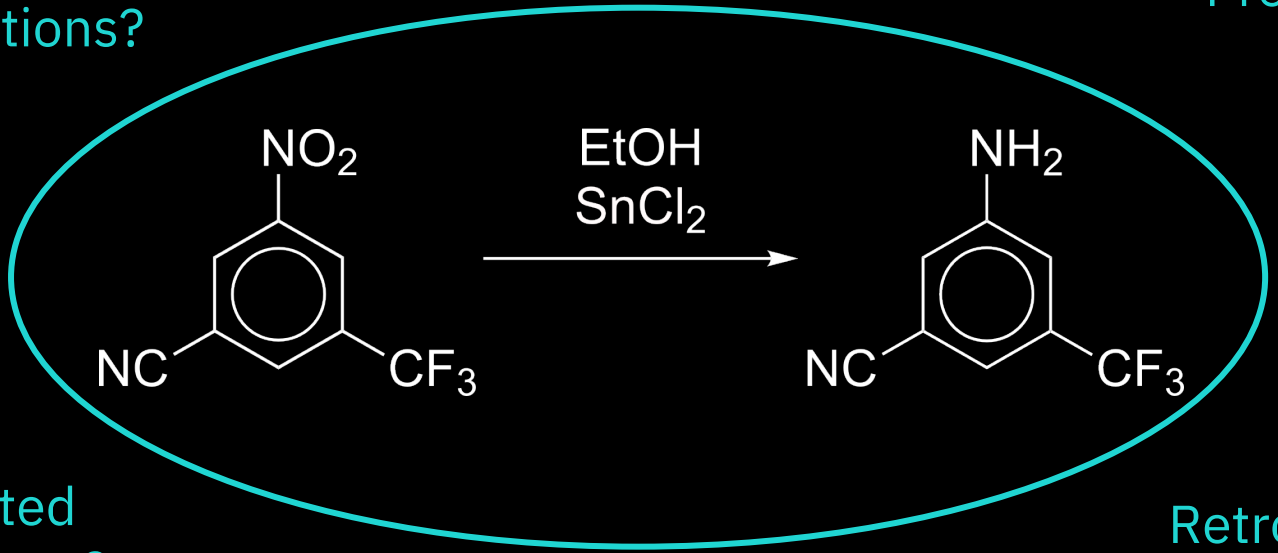
Outline

1. AI models for chemistry
2. Building a user interface and API
3. Use cases and special considerations
4. Challenges and outlook

AI and chemical reactivity

Experimental conditions?

Product?

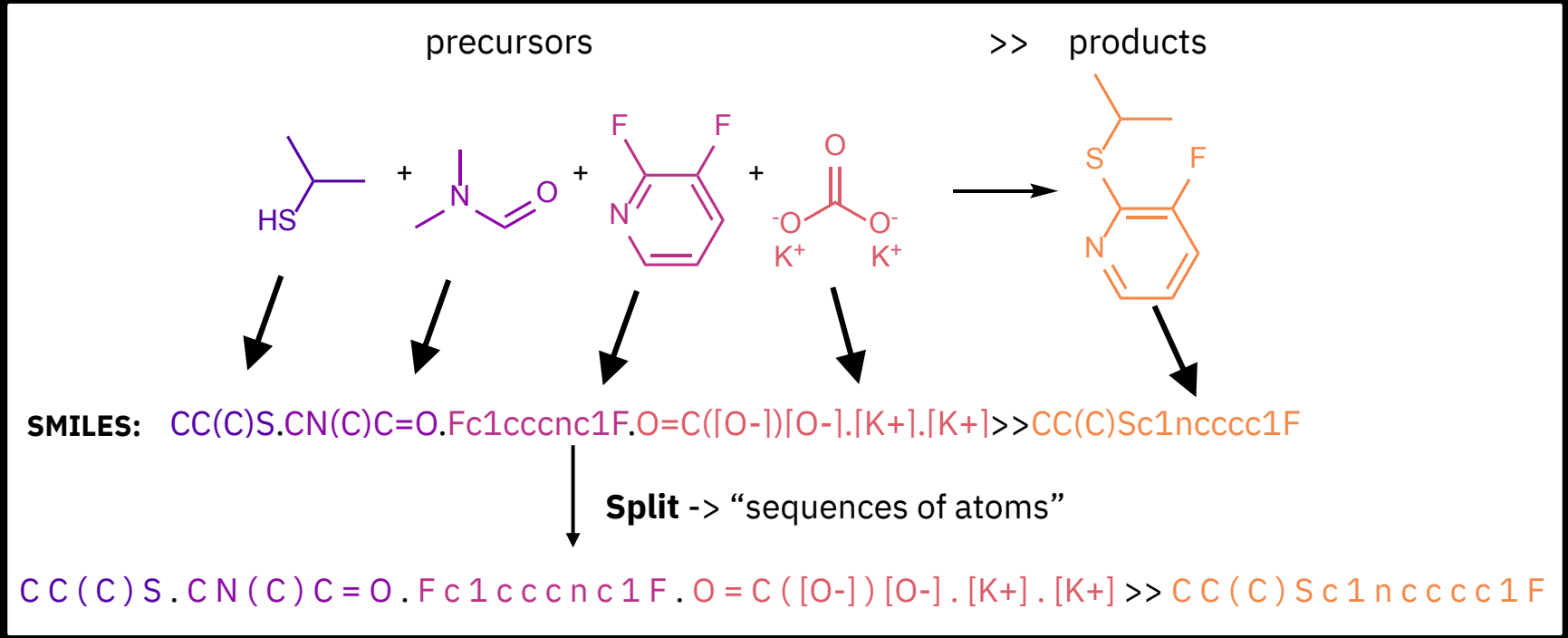


Related reactions?

Retrosynthesis?

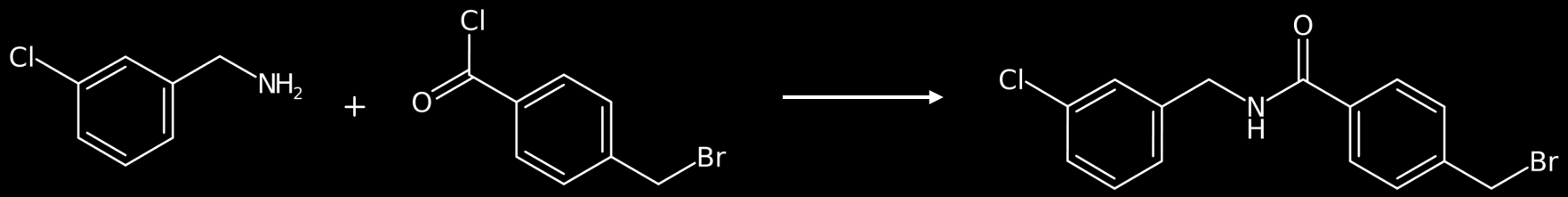
Yield?

Atoms as letters, molecules as words



→ Borrow AI methods developed for human languages

Reaction prediction



Textual representation (SMILES)

NCc1cccc(Cl)c1 O=C(Cl)c1ccc(CBr)cc1 O=C(NCc1cccc(Cl)c1)c1ccc(CBr)cc1

“Sentence of atoms”

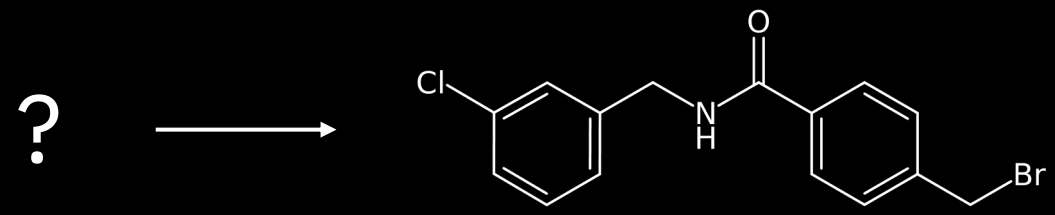
“Translation”

N C c 1 c c c c (Cl) c 1 . O = C (Cl) c 1 c c c (C Br) c c 1 → O = C (N C c 1 c c c c (Cl) c 1) c 1 c c c (C Br) c c 1

Molecular Transformer

Schwaller, P.; Laino, T.; Gaudin, T.; Bolgar, P.; Hunter, C. A.; Bekas, C. & Lee, A. A., *ACS Cent. Sci.*, **2019**, *5*, 1572-1583.

Retrosynthetic analysis

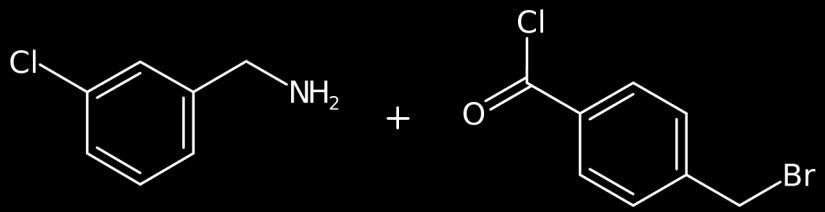


Similar approach, both sides switched

“Translation”

`O = C ( N C c 1 c c c c ( Cl ) c 1 ) c 1 c c c ( C Br ) c c 1` ⟹ `N C c 1 c c c c ( Cl ) c 1 . O = C ( Cl ) c 1 c c c ( C Br ) c c 1`

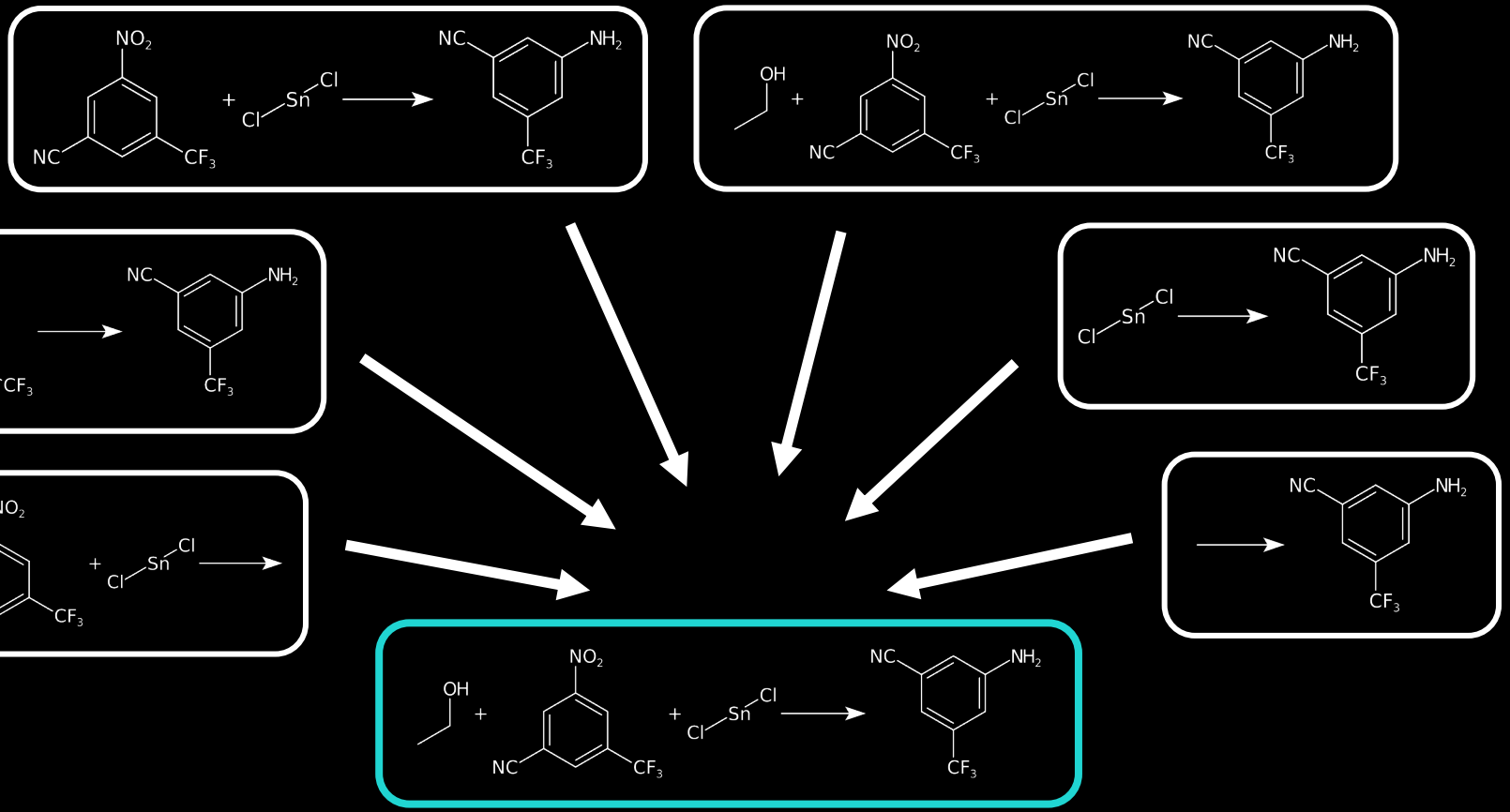
Transformer



Schwaller, P.; Petraglia, R.; Zullo, V.; Nair, V. H.; Haeuselmann, R. A.; Pisoni, R.; Bekas, C.; Iuliano, A. & Laino, T., *Chem. Sci.*, **2020**, *11*, 3316-3325.

Completing partial chemical equations

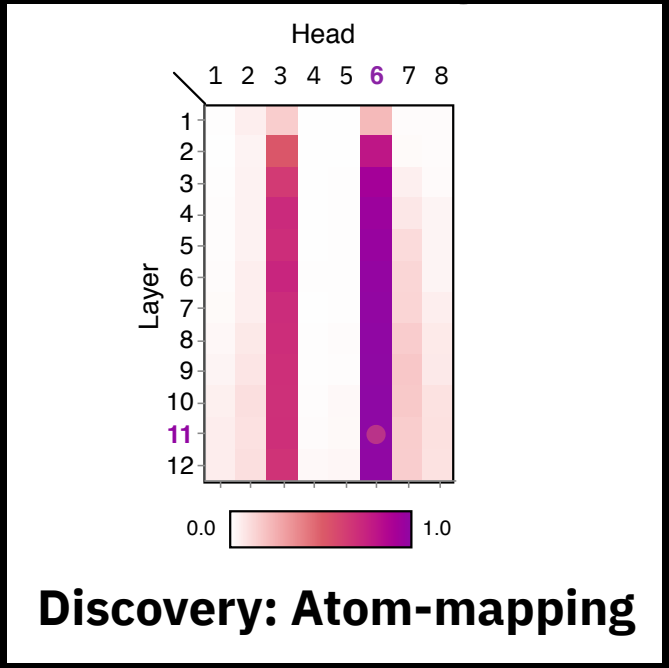
Example 3/5



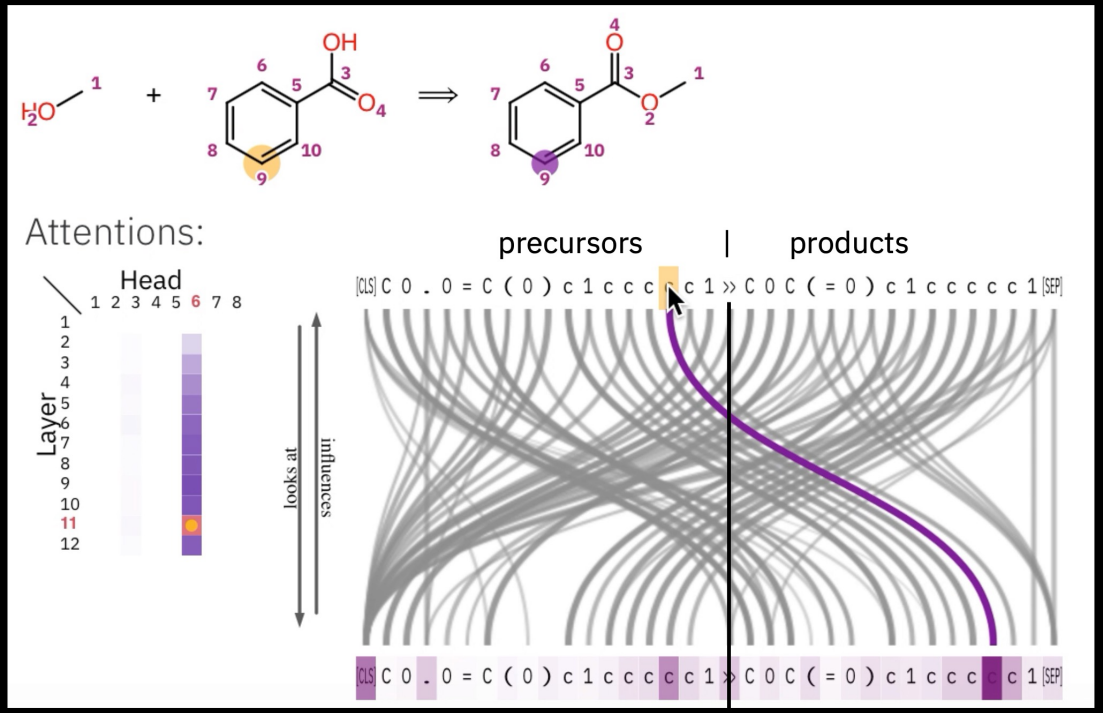
Vaucher, A. C.; Schwaller, P. & Laino, T., Chemrxiv.13273310, **2020**.

Atom mapping: RXNMapper

Example 4/5



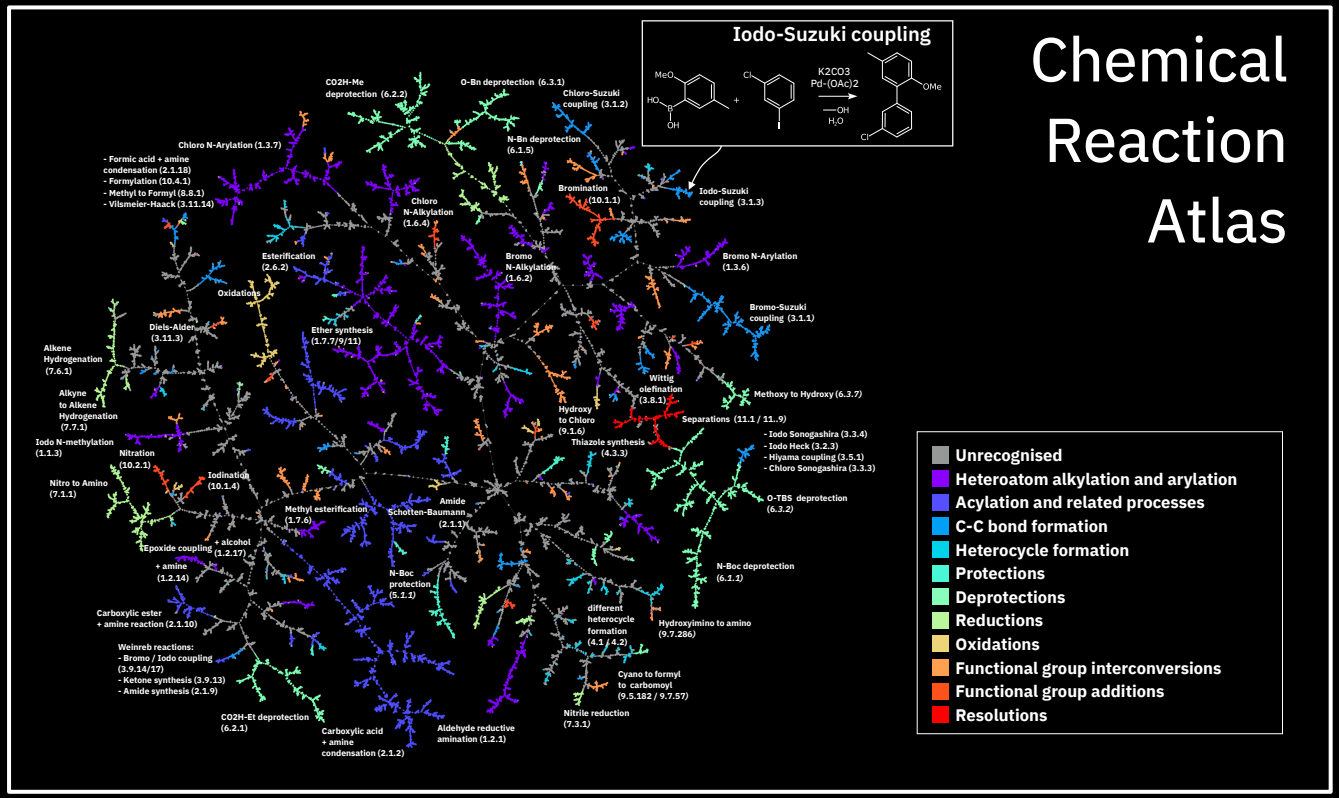
Discovery: Atom-mapping



Schwaller, P.; Hoover, B.; Reymond, J.-L.; Strobelt, H. & Laino, T., *Sci. Adv.*, **2021**, *7*, eabe4166.

Classifying and mapping reactions

Example 5/5



Schwaller, P.; Probst, D.; Vaucher, A. C.; Nair, V. H.; Kreutter, D.; Laino, T. & Reymond, J.-L., *Nat. Mach. Intell.*, **2021**, *3*, 144-152.

Outline

1. AI models for chemistry
2. Building a user interface and API
3. Use cases and special considerations
4. Challenges and outlook

Why provide interfaces to users?

- New applications of AI to chemistry every week
- Accessibility and reproducibility often limited

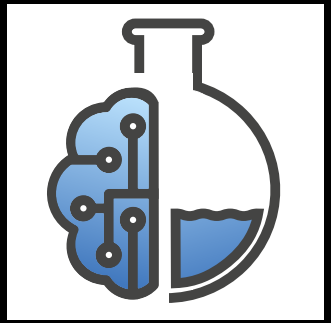
Model availability:

- Not available or limited availability (often)
- Open-source (often)
  - Easy to use?
  - Reproducible results?
- Web service (rare)

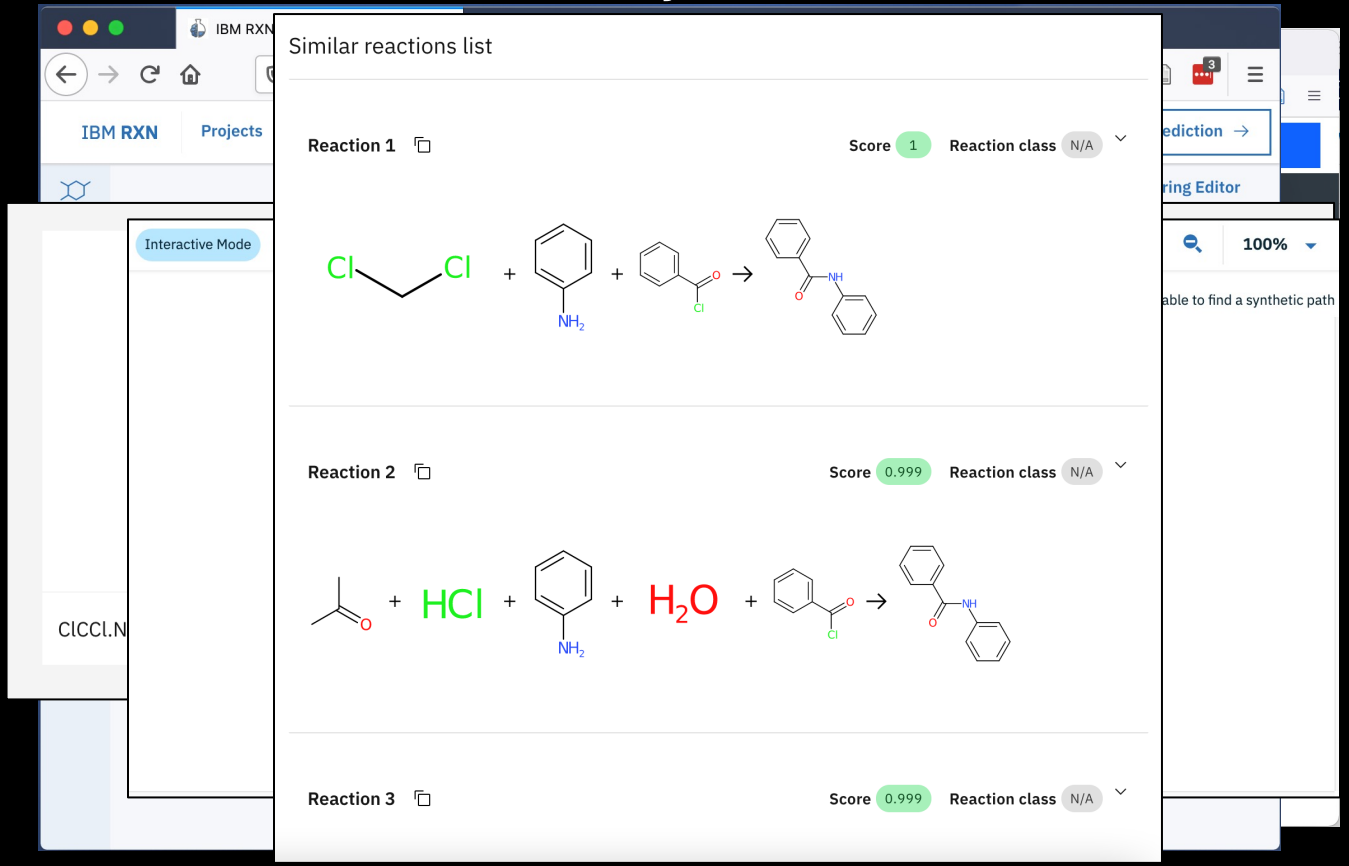
Consequence: limited use by the target audience (bench chemists)

User interfaces

- Challenges to make models more accessible:
  - Requires time
  - Requires expertise
- Starting in 2018: IBM RXN for chemistry



IBM RXN for chemistry



Freely available web platform

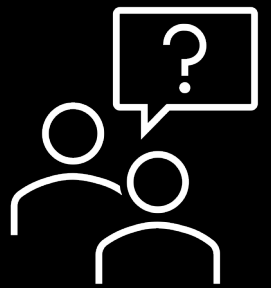
rxn.res.ibm.com

IBM RXN for chemistry

- Positive reception by the community

- At the beginning, limited to:
  - Use via web browser
  - Reaction outcome prediction model

- Quite soon, a user: “Can I access the API?”



Going beyond a user interface

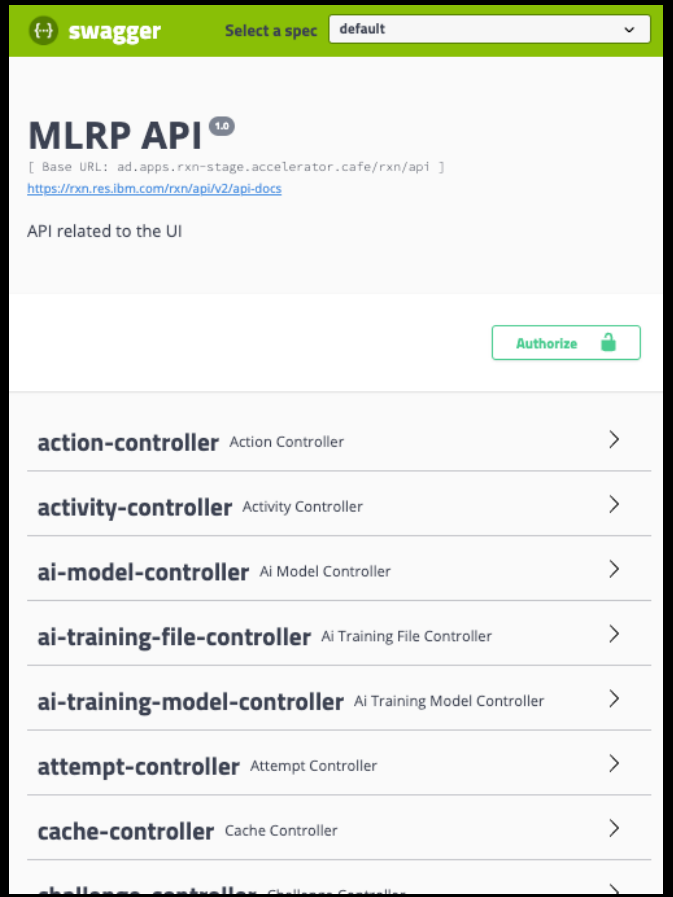
- User interface not always enough:
  - Automated predictions
  - Composability & modularity
  - Integration in other workflows
  - …

→ application programming interface (API)

IBM RXN API

- Same endpoints as for UI
- Direct use: not straightforward (authentication, etc.)

→ Python package for API wrapper



API wrapper on GitHub:
github.com/rxn4chemistry/rxn4chemistry

Launcher | rxn4chemistry_tour.ipynb

Markdown | Python 3 (ipykernel)

Running a reaction prediction is as simple as:

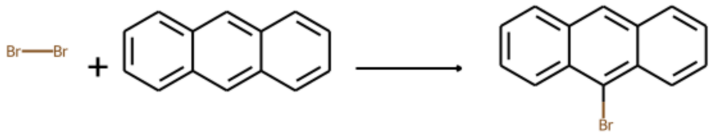
[5]:
```
predict_reaction_response = rxn4chemistry_wrapper.predict_reaction(
    'BrBr.c1ccc2cc3ccccc3cc2c1'
)
```

NOTE: we have set limitations on the number of calls per second and per minute in the public version of RXN for Chemistry. These limits can be tweaked or removed in on-premise deployments. Those limitations are currently set to 5 calls per minute, in most cases this is not a problematic limitation.

[6]:
```
predict_reaction_results = rxn4chemistry_wrapper.get_predict_reaction_results(
    predict_reaction_response['prediction_id']
)
```

[7]:
```
get_reaction_from_smiles(predict_reaction_results['response']['payload']['attempts'][0]['smiles'])
```

[7]:



It is possible to run reaction prediction in batches (not storing the information in any project) to use the service in a highthroughput fashion:

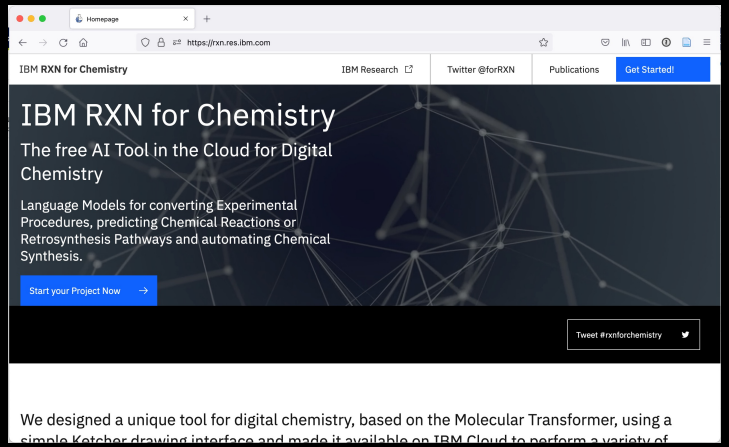
[8]:
```
predict_rection_batch_response = rxn4chemistry_wrapper.predict_reaction_batch(
```

Outline

1. AI models for chemistry
2. Building a user interface and API
3. Use cases and special considerations
4. Challenges and outlook

Current status

rxn.res.ibm.com



github.com/rxn4chemistry/rxn4chemistry

Status early 2023:
~35k registered users
~9M predictions

Use cases of IBM RXN

Chemical synthesis

Chemical reactivity screening

Education

... and others !

Use cases of IBM RXN

Chemical synthesis

Chemical reactivity screening

Education

... and others !

Use cases of IBM RXN

Chemical synthesis

Chemical reactivity screening

Education

... and others !

Use cases of IBM RXN

Chemical synthesis

Chemical reactivity screening

Education

... and others !

Use cases of IBM RXN

Chemical synthesis

Chemical reactivity screening

Education

... and others !

Special considerations

Data confidentiality

User feedback

Concept of users and projects

Symmetry UI-API

Special considerations

Data confidentiality

User feedback

Concept of users and projects

Symmetry UI-API

Special considerations

Data confidentiality

→ Portability is important

back

and projects

UI-API

Special considerations

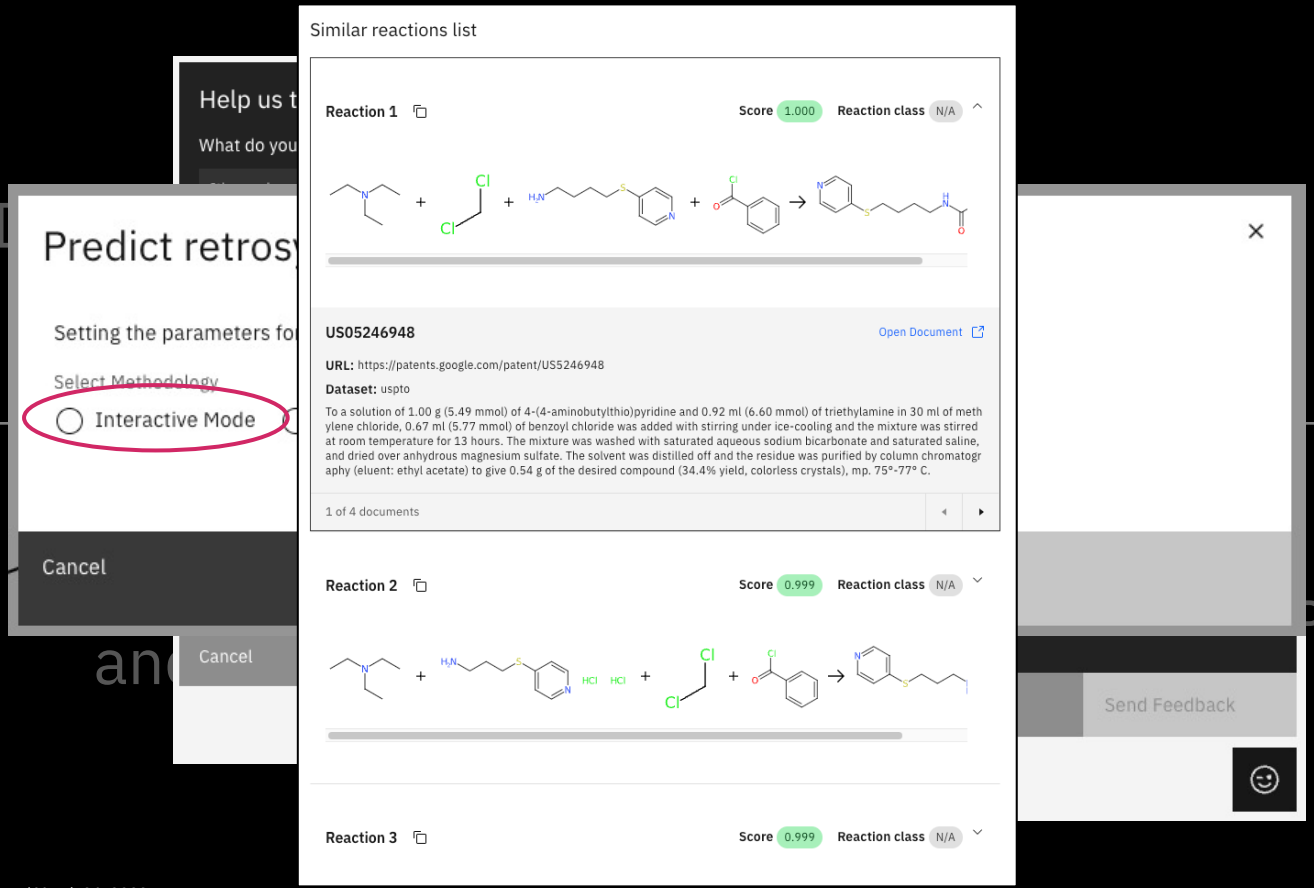
Data confidentiality

User feedback

Concept of users and projects

Symmetry UI-API

Special considerations



Special considerations

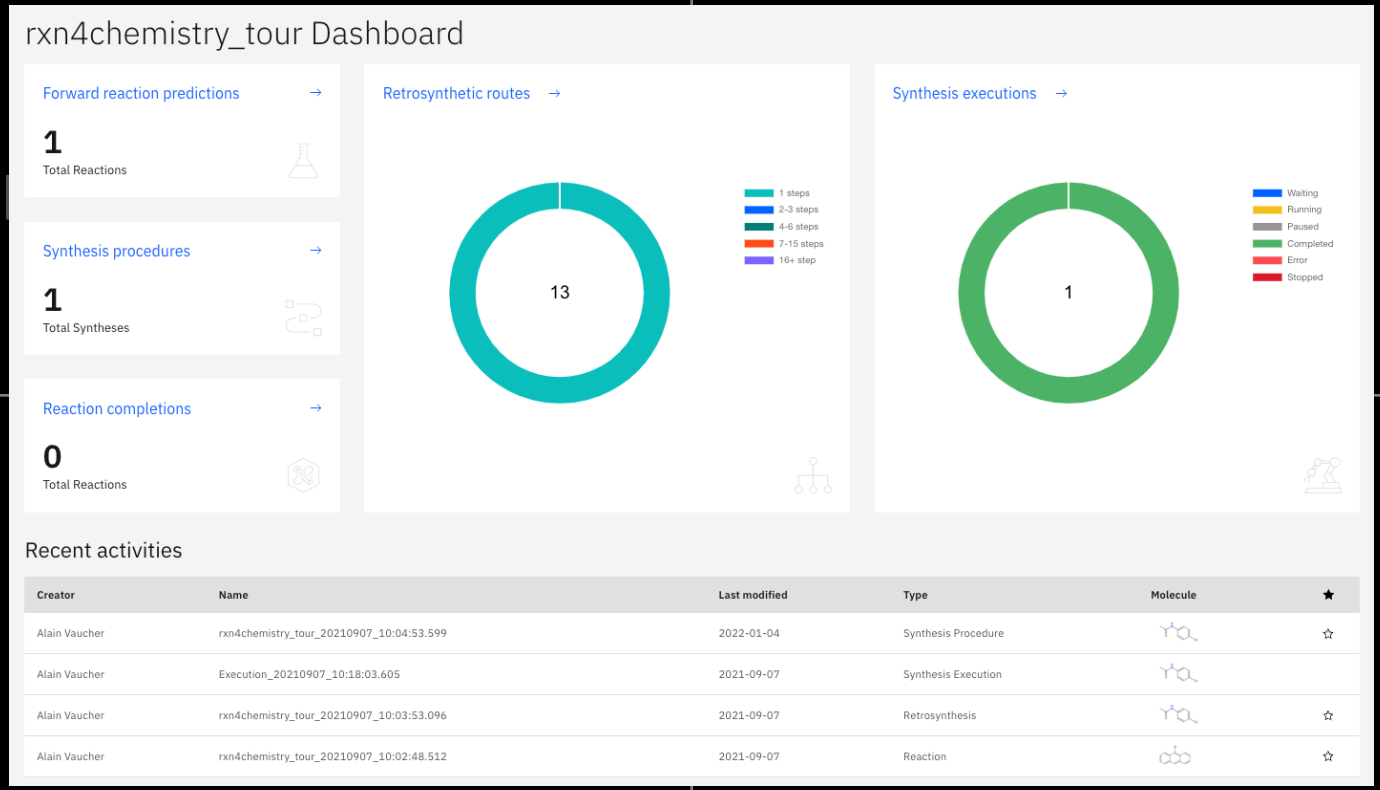
Data confidentiality

User feedback

Concept of users and projects

Symmetry UI-API

Special considerations



Special considerations

Data confidentiality

User feedback

Concept of users and projects

Symmetry UI-API

Special considerations

Project list 14

Search by name

Name	Owner	Creation date	Reactions	Retrosynthetic routes	Synthesis	Reaction completion
reaction completion	Alain Vaucher	2023-02-13	0	0	0	1
[redacted]	Alain Vaucher	2023-02-13	0	0	0	5
[redacted]	Alain Vaucher	2023-02-10	0	0	0	1
[redacted]	Alain Vaucher	2022-02-24	0	3	0	0
[redacted]	Alain Vaucher	2022-02-17	0	35	18	0
[redacted]	Alain Vaucher	2021-11-30	0	1	0	0
[redacted]	Chiara Faralli	2021-10-19	13	137	19	0
[redacted]	Alain Vaucher	2021-10-13	1	14	1	0
[redacted]	Alain Vaucher	2021-09-08	1	0	0	0
[redacted]	Alain Vaucher	2021-09-07	1	13	1	0
[redacted]	Chiara Faralli	2020-10-21	0	16	15	0
[redacted]	Alain Vaucher	2020-09-28	0	7	3	0
[redacted]	Chiara Faralli	2020-06-15	6	4	1	0
[redacted]	Alain Vaucher	2020-06-11	42	171	42	13

Items per page: 20 | 1-14 of 14 items | 1 of 1 page

Projects

```
[26]: projects_response = rxn4chemistry_wrapper.list_all_projects()
      projects = projects_response["response"]["payload"]["content"]

[27]: print(len(projects))
```

```
14
```

```
[28]: from pprint import pprint
      pprint(projects[-1])
```

```
{'attempts': [],
 'computedFields': {},
 'createdBy': '6eec3679-b8de-40e2-b73e-0dcd1b9ccf1c',
 'createdOn': 1676273811483,
 'description': None,
 'embed': {},
 'id': '63e9e89350f97b001f83c766',
 'isFavorite': False,
 'metadata': {},
 'modifiedBy': '6eec3679-b8de-40e2-b73e-0dcd1b9ccf1c',
 'modifiedOn': 1676273811483,
 'name': 'reaction completion',
 'visibility': None}
```

Outline

1. AI models for chemistry
2. Building a user interface and API
3. Use cases and special considerations
4. Challenges and outlook

Challenges

Backward compatibility

Cost

API rate limits

Handling of API errors

Challenges

Backward compatibility

Cost

API rate limits

Handling of API errors

Challenges

Backward comp

Cost

API rate lim

ling of API errors

https://xkcd.com/1172/

Challenges

Backward compatibility

Cost

API rate limits

Handling of API errors

Challenges

Backward compatibility

Cost

API rate limits

Handling of API errors

Challenges

Backward compatibility

Cost

```
[7]: predict_reaction_response_1 = rxn4chemistry_wrapper.predict_reaction('BrBr.c1ccc2cc3ccccc3cc2c1')
     predict_reaction_response_2 = rxn4chemistry_wrapper.predict_reaction('BrBr.c1ccc2cc3ccccc3cc2c1CC')

ERROR:rxn4chemistry.response_handler:The service might be overloaded at the moment. Please try again.
ERROR:rxn4chemistry.response_handler:Full response: {"timestamp":1679481031108,"status":429,"error":"T
oo Many Requests","message":"Too many requests","path":"/rxn/api/api/v1/predictions/pr"}
```

API rate limits

Handling of API errors

Challenges

Backward compatibility

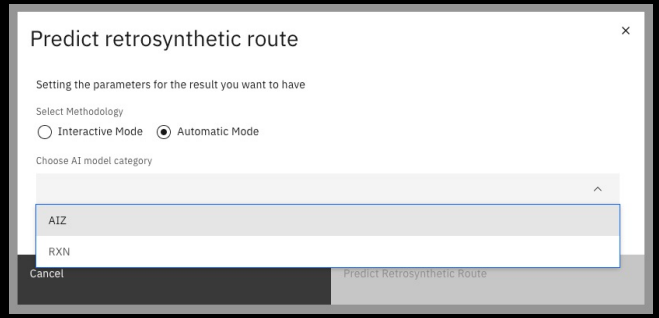
Cost

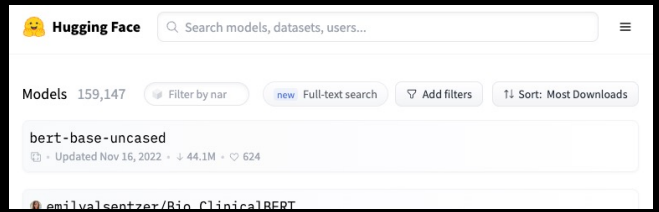
API rate limits

Handling of API errors

Outlook

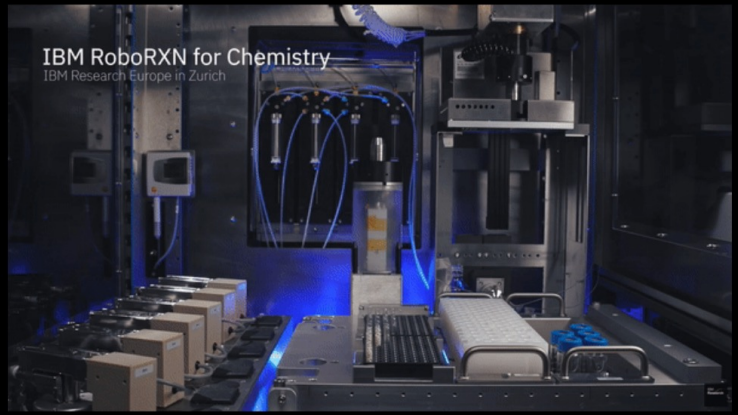
Integration of external models





Steering synthesis robot from the API

```
[5]: synthesis_execution = rxn4chemistry_wrapper.start_synthesis(synthesis_id)
```



Thank you for your attention!

Questions or comments

E-mail: ava@zurich.ibm.com

Twitter: @acvaucher

Talks by other team members:
ibm.biz/acs-spring-2023

Thank you for your attention!

Questions or comments

E-mail: ava@zurich.ibm.com

Twitter: @acvaucher