Being more reactive
Expanded chemical reaction support in the RDKit

Nadine Schneider

NIBR Informatics

3rd RDKit UGM, Darmstadt

October 23, 2014

NOVARTIS

What makes us more reactive?

- Development and integration of new fingerprints for chemical reactions in the RDKit
  - Difference fingerprint for similarity search or model building
  - Structural fingerprint for database scans or substructure search
- Including agents in our chemistry set
- Chemical reactions in the RDKit PgSQL cartridge
  - Supports GiST-index substucture/superstructure scans
  - Supports similarity search using reaction fingerprints

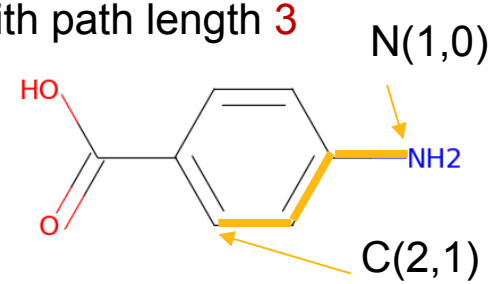
NOVARTIS

Development of new reaction fingerprints

How to encode a chemical reaction?

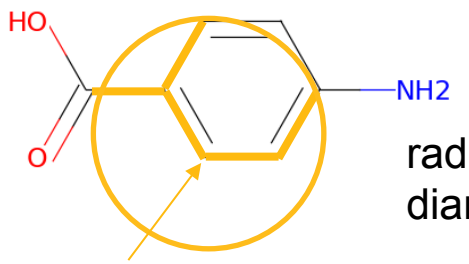
- Various molecular fingerprints are available to describe a molecule
  - Topological or path-based FP (AtomPairs (AP), Topological torsions (TT), RDKit FP)
  - Circular FP (ECFP, FCFP, Morgan)
  - Dictionary-based (MACCS keys)

AP with path length 3



Resulting bit:
C(2,1)-3-N(1,0)

ECFP4 or Morgan2



radius: 2
diameter: 4

Atom type:C_2_1_0_0_1

- Can we use these molecular FPs for reactions?
- How to capture the “important” information of a chemical reaction?

NOVARTIS

Development of new reaction fingerprints

How to encode a chemical reaction?

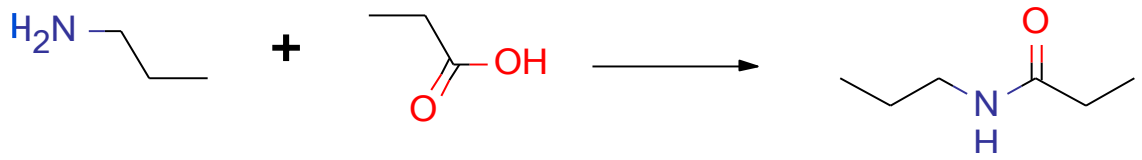
- In reactions the mechanism or the transformation is more interesting than the molecules their self
- Some approaches exist to capture a reaction in a FP:
  - Initial idea: build difference vectors of descriptors of the reactant and the products (Broughton et al. 2003, US patent application)
  - Reaction FPs applied to find similar metabolic reactions (Ridder & Wagener, ChemMedChem, 2008)
  - Reaction vectors used for *de novo* design of synthetically feasible molecules (Patel et al., JCIM, 2009)

→Encode the reaction as the descriptors that are gained in the product(s) and those lost from the reactant(s)

Construction of difference fingerprints

Combine molecular FPs to reaction FP

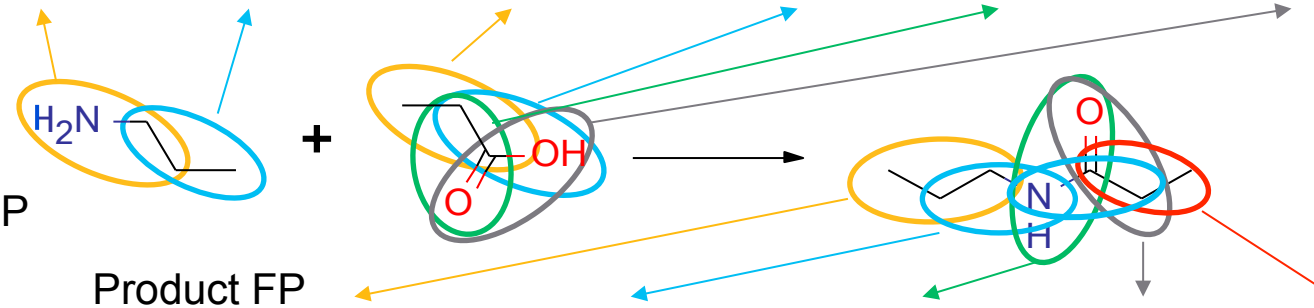
Carboxylic acid + amine reaction



Reactant FP

AP2	N(1,0)-2-C(2,0)	C(2,0)-2-C(1,0)	C(1,0)-2-C(3,1)	C(2,0)-2-O(1,1)	C(2,0)-2-O(1,0)	O(1,1)-2-O(1,0)
#	1	1	1	1	1	1

Build reaction FP



subtract

Product FP

AP2	C(1,0)-2-C(2,0)	C(2,0)-2-N(2,0)	N(2,0)-2-O(1,1)	O(1,1)-2-C(2,0)	C(3,1)-2-C(1,0)
#	1	2	1	1	1

Reaction FP

AP2	**N(1,0)-2-C(2,0)**	**C(2,0)-2-C(1,0)**	**C(1,0)-2-C(3,1)**	**C(2,0)-2-O(1,1)**	**C(2,0)-2-O(1,0)**	**O(1,1)-2-O(1,0)**	**C(2,0)-2-N(2,0)**	**N(2,0)-2-O(1,1)**
#	**-1**	**0**	**0**	**0**	**-1**	**-1**	**+2**	**+1**

NOVARTIS

Difference fingerprints in the RDKit

Implementation of customizable difference reaction FP

- RDKit reaction FP, represented as SparseIntvect:

$$reactionFP = w_{nonAgent}\left(\sum_{products\ \ i} productFP_i - \sum_{reactants\ \ i} rectantFP_i\right) + w_{agent}\sum_{agents\ \ i} agentFP_i$$

- Customizable interface allows to choose fingerprint type, bit size, and handling of agents

```
SparseIntVect<boost::uint32_t> *
DifferenceFingerprintChemReaction(const ChemicalReaction &rxn,
const ReactionFingerprintParams &params = DefaultDifferenceFPParams);
```

- Available in C++ as well as Python interface

Validation of the RDKit reaction FPs

How useful are these in model building and similarity search?

- First step: find some proper reaction data

→Problem: No public databases available

- Text-mining effort: extracted reactions from more than one million US granted patents (between 1976 and 2013) [1]
- 1109897 chemical reactions including information about agents, solvents, and yields if available
- RSC's RXNO ontology-based classification using the tool NameRxn [2]

→ Almost 600000 reactions could be classified in more than 300 reaction types

[1] Lowe DM: "Extraction of chemical structures and reactions from the literature." PhD thesis. University of Cambridge: Cambridge, UK; 2012.
[2] Reaction classification from Roger Sayle and Daniel Lowe (NextMove Software)

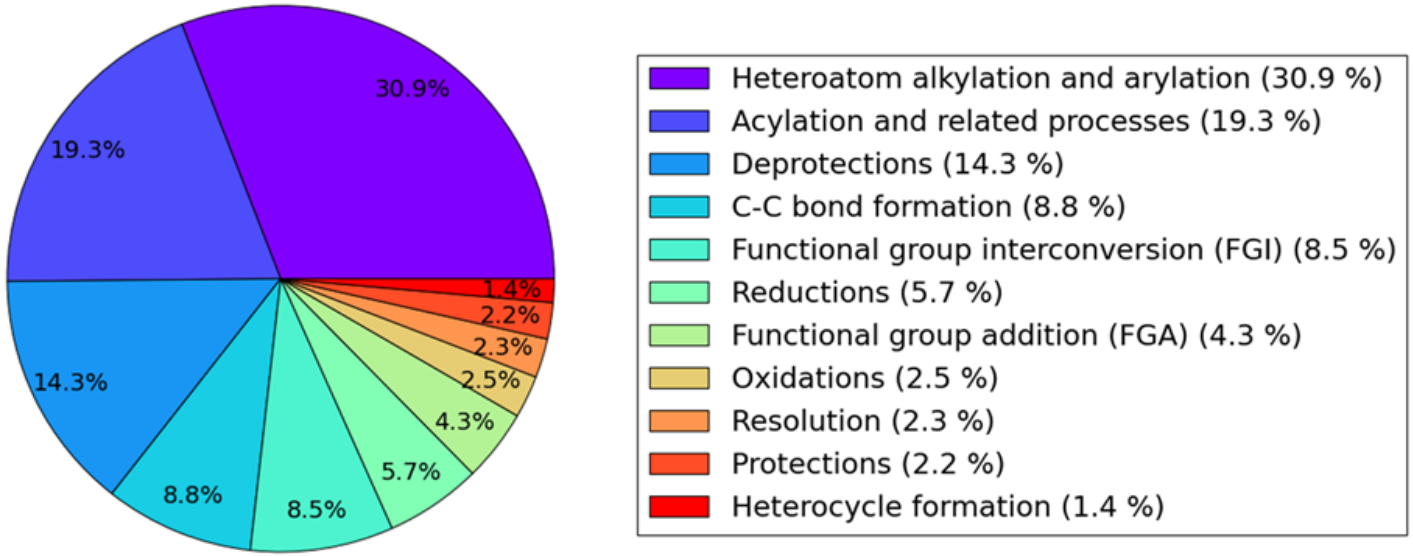
NOVARTIS

Reaction classification based on RXNO

Class distribution in the patent dataset

54 % of the data could be classified:

- 11 super-classes (e.g. ‘3’ C-C bond formation) [1]
- 80 classes/categories (e.g. ‘3.1’ Suzuki-Miyaura) [2]
- 318 named reactions/types (e.g. ‘3.1.1’ Bromo Suzuki coupling)



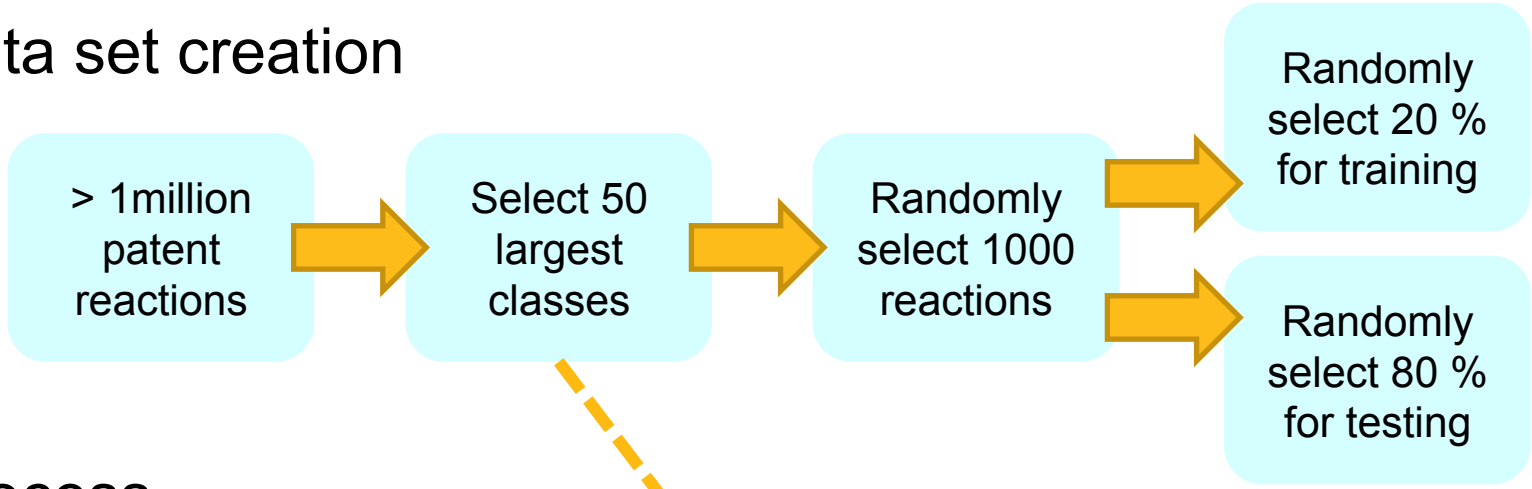
[1] Carey et al., OrgBiomolChem, 2006
[2] Roughley et al., JMedChem, 2011

NOVARTIS

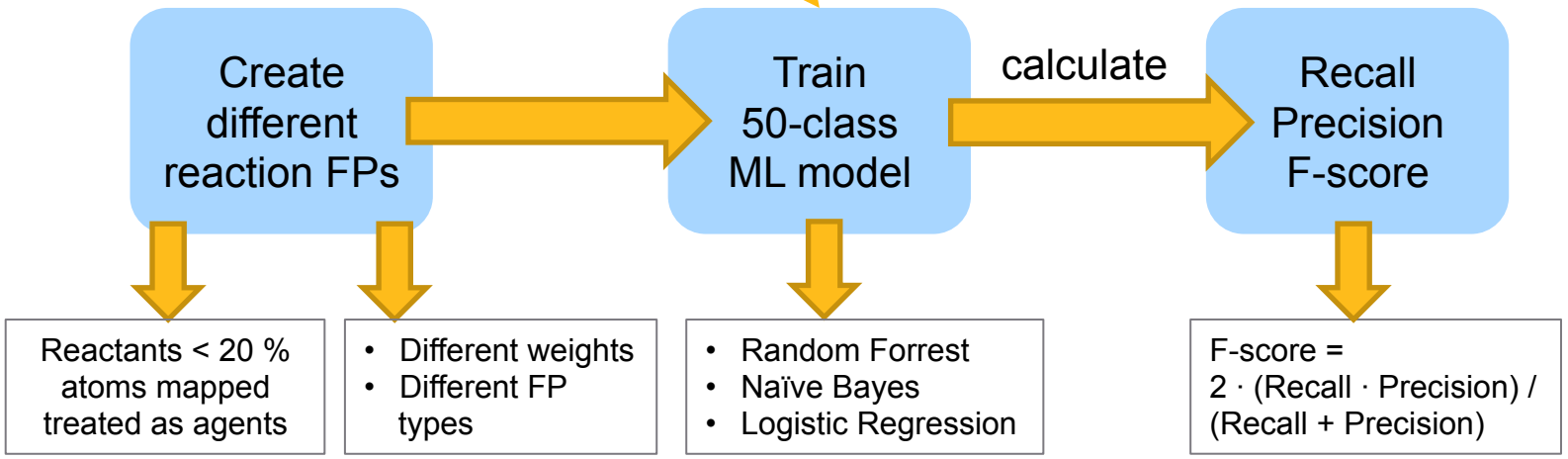
Reaction classification using reaction fingerprints

Experimental setup

- Data set creation



- Process



NOVARTIS

Learning reaction types

Influence of agents in the 50-class model

Results:

Test set

Fingerprint		RF (max depth 15)			RF (max depth 25)		
Type	Parameters	Recall	Prec	F-score	Recall	Prec	F-score
reactionFP 2048 bit (AP)	10_1_wA	0.79	0.85	0.82	0.89	0.89	0.89
	10_-1_wA	0.78	0.82	0.8	0.86	0.88	0.87
	1_1_wA	0.79	0.85	0.82	0.89	0.89	0.89
	w/oA	0.92	0.92	0.92	0.94	0.94	0.94
reactionFP 4096 bit (AP)	10_1_wA	0.8	0.85	0.82	0.89	0.9	0.9
	10_-1_wA	0.79	0.83	0.81	0.87	0.88	0.87
	1_1_wA	0.8	0.85	0.82	0.89	0.9	0.9
	w/oA	0.92	0.92	0.92	0.94	0.94	0.94

- Including agents in the reaction FP significantly reduces the performance
- 2 K reaction FP fingerprint size is sufficient
- Larger depth of the trees in the RF reduces the number of errors

Special fingerprints for agents

Do not mix reaction and agent FP!

- Feature FP:
  - Very general description of the physico-chemical properties of agents:

    MW, NumAtoms, NumRings, LogP, NumRadicalElectrons, TPSA, NumHeteroAtoms, NumHAcceptors, NumHDonors
- Morgan2 FP:
  - Substructure description of the agents
- Dictionary-based FP:
  - Representing the most common agents found in the patent reactions
- Concatenate reaction and agent FP

Learning reaction types

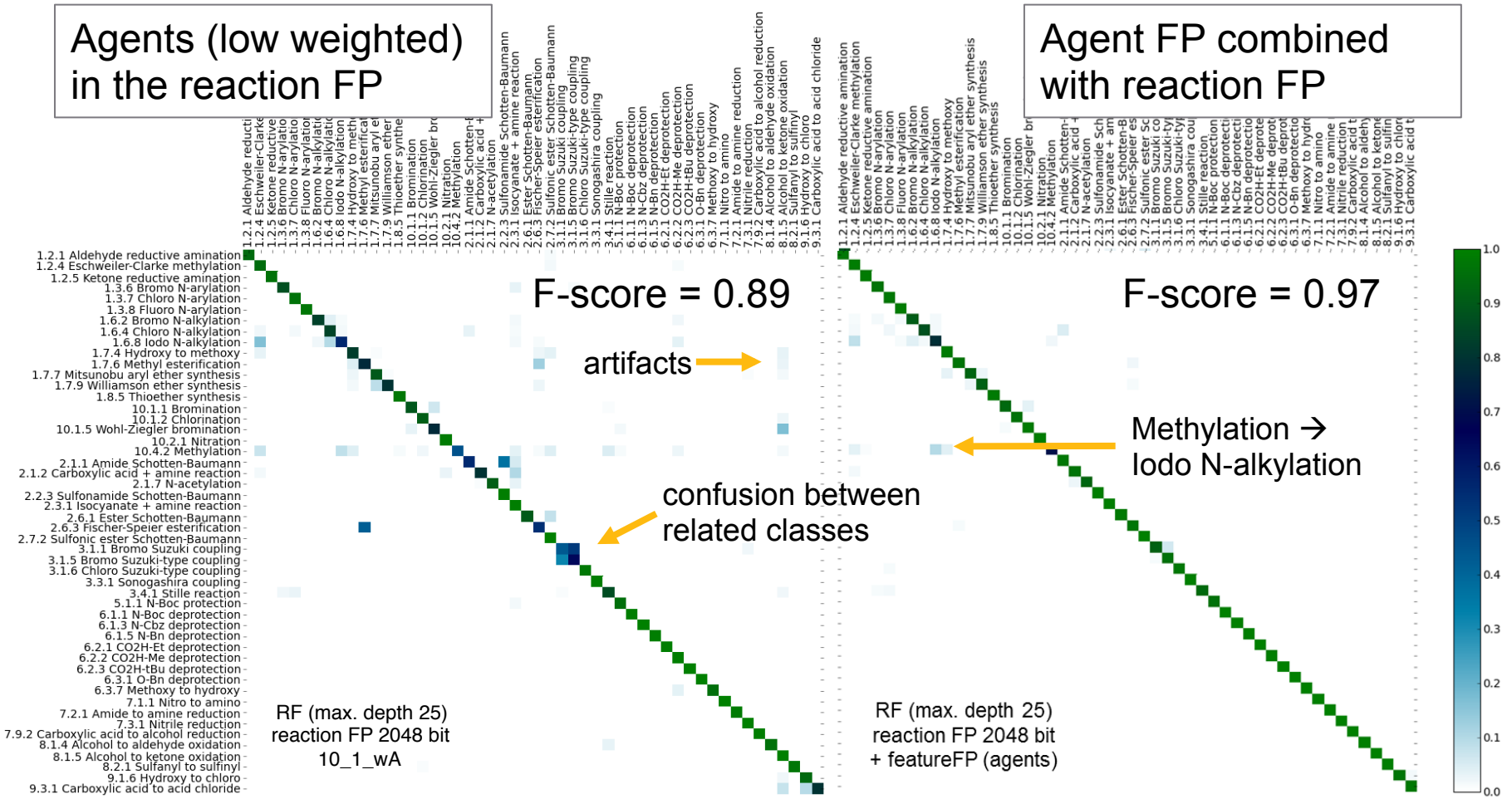
Include special fingerprints for agents

- Agents can be helpful to distinguish related reaction classes if they are incorporated properly

Fingerprint		RF (max depth 15)			RF (max depth 25)		
Type	**Parameters**	**Recall**	**Prec**	**F-score**	**Recall**	**Prec**	**F-score**
reactionFP 2048 bit (AP)	10_1_wA	0.79	0.85	0.82	0.89	0.89	0.89
	w/oA	0.92	0.92	0.92	0.94	0.94	0.94
reactionFP 2048 bit + featureFP (agents)		0.95	0.95	0.95	0.97	0.97	0.97
reactionFP 2048 bit + dictionary-based FP (agents)		0.95	0.95	0.95	0.96	0.96	0.96
reactionFP 2048 bit + Morgan2 FP (agents)		0.95	0.95	0.95	0.97	0.97	0.97

Learning reaction types

Excellent performance combining reaction and agent FP



NOVARTIS

Learning reaction types

Impact of different fingerprint types and ML methods

- Using more local fingerprints (AP3, Morgan2, TT) to better capture the reaction transformation

Fingerprint		RF (max depth 25)			K-Means (k=3)		
Type	**Parameters**	**Recall**	**Prec**	**F-score**	**Recall**	**Prec**	**F-score**
transformationFP 2048 bit	AP3	0.95	0.95	0.95	0.87	0.88	0.88
	Morgan2	0.94	0.94	0.94	0.88	0.89	0.88
	TT	0.82	0.85	0.84	0.76	0.78	0.77
transformationFP 4096 bit w/o negative counts	AP3	0.95	0.95	0.95	-	-	-
	Morgan2	0.94	0.94	0.94	-	-	-
	TT	0.84	0.85	0.84	-	-	-
		multinomial NB (α=0.0001)			**LR**		
transformationFP 2048 bit	AP3	-	-	-	0.95	0.95	0.95
	Morgan2	-	-	-	0.94	0.94	0.94
	TT	-	-	-	0.91	0.91	0.91
transformationFP 4096 bit w/o negative counts	AP3	0.93	0.93	0.93	-	-	-
	Morgan2	0.89	0.9	0.89	-	-	-
	TT	0.87	0.87	0.87	-	-	-

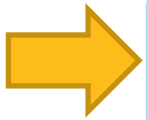
- Logistic regression (LR) best performance across all FPs

Selection of the final reaction fingerprint

How minimal it could be?

Final model:

- Transformation FP (256 bit)
- Concatenate agent feature FP (9 bit)
- ML: non-parameter dependent Logistic Regression



Keep it as simple as possible

Fingerprint		LR		
Type	**Parameters**	**Recall**	**Prec**	**F-score**
transformationFP AP3 (folded)	4096 bit	0.95	0.95	0.95
	2048 bit	0.95	0.95	0.95
	1024 bit	0.95	0.95	0.95
	512 bit	0.95	0.95	0.95
	256 bit	0.95	0.95	0.95
	128 bit	0.93	0.93	0.93
transformationFP AP3 (unfolded)	(1232 bits)	0.95	0.95	0.95
transformationFP AP3 (folded) + featureFP (agents)	256 bit + 9 bit	0.97	0.97	0.97
transformationFP AP3 (folded) + Morgan2 FP (agents)	256 bit + 256 bit	0.97	0.97	0.97
transformationFP AP3 (unfolded) + featureFP (agents)	(1241 bits)	0.98	0.98	0.98
transformationFP AP3 (unfolded) + Morgan2 FP (agents)	(13828 bits)	0.98	0.98	0.98

NOVARTIS

Some failures of our new reaction fingerprint

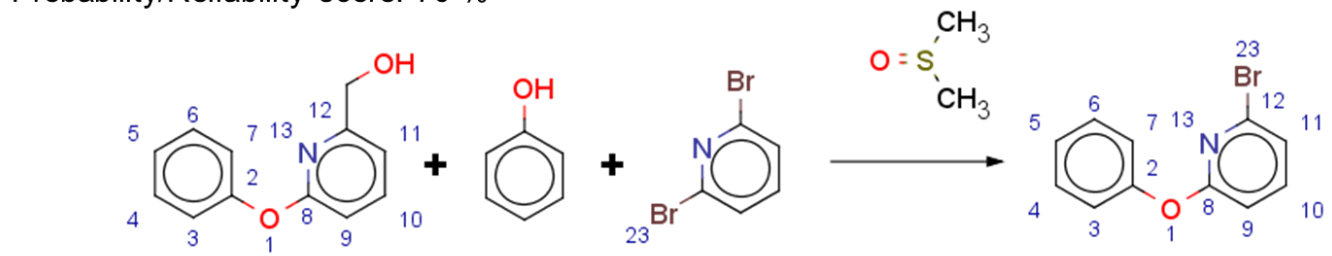
Are they explainable?

- Problem: atom-mapping incorrect → true reactants removed to agents

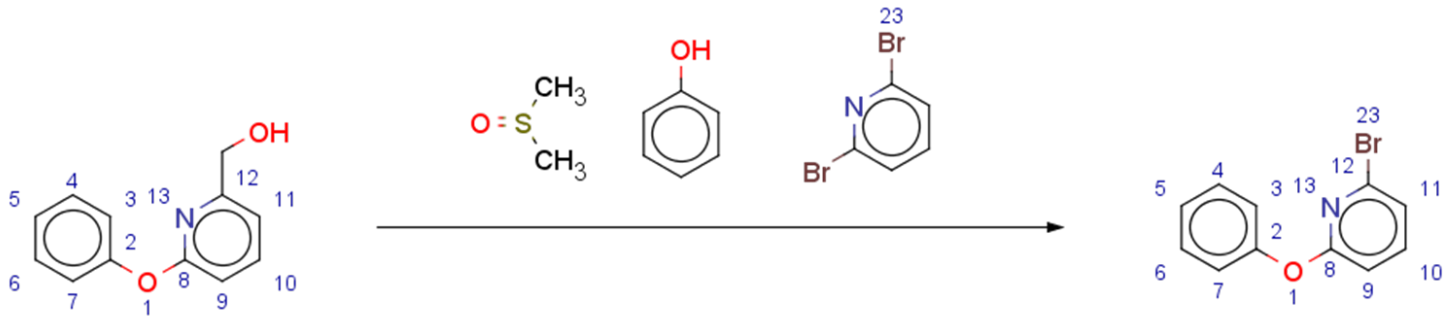
Reaction type: **Williamson ether synthesis (1.7.9)**

Predicted reaction type: **Bromination (10.1.1)**

Probability/Reliability score: 70 %



Removing unmapped reactants to agents



NOVARTIS

Some failures of our new reaction fingerprint

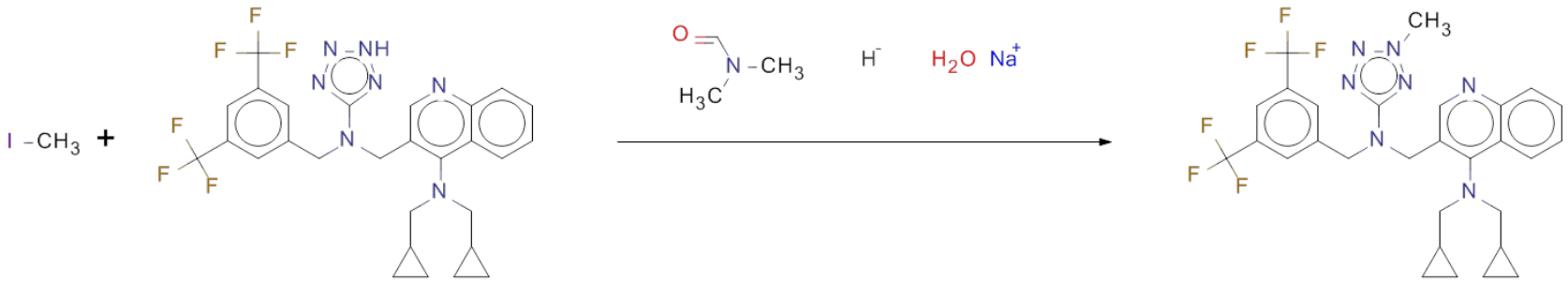
Are they explainable? (cont.)

- Problem: ambiguous classification of a reaction type

Reaction type: **Methylation (10.4.2)**

Predicted reaction type: **Iodo N-alkylation (1.6.8)**

Probability/Reliability score: 69 %



NOVARTIS

What's next?

Conclusion, issues, and perspectives

- Learn more about the new reaction FP

→Find the details in the upcoming paper ☺ (almost submitted)

- Some of the new functionality is already contained in the RDKit github version (agents, reaction FP, etc.)

→Provide the new transformation and agent FP in the upcoming RDKit release

- Performance of the fingerprint strongly depends on the atom-mapping

→Become more independent of it concerning the determination of agent

Reactions in the PostgreSQL database cartidge

Last but not least

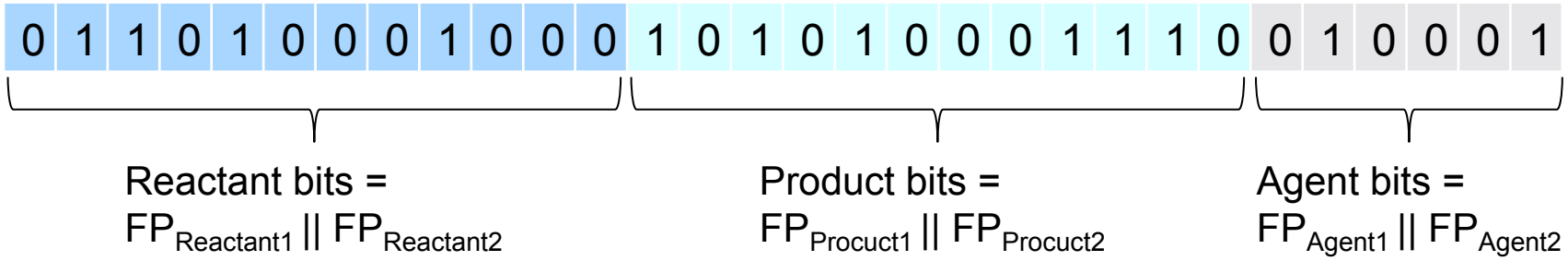
- RDKit PgSQl cartridge supports new reaction type now
- Input reaction as SMILES, SMARTS or CTAB
- Supports reaction fingerprints
- Supports GiST-index substructure/superstructure scans
  - Using a 4096bit reaction pattern-FP as search key
- Agents can be included or not in FP scan or for calculations

Generation of a new reaction substructure FP

New reaction substructure FP in the RDKit

Also with a customizable interface

- Represented as ExplicitBitVect:



- Customizable interface allows to choose fingerprint type, bit size, and handling of agents (bit ratio in the FP)

```
ExplicitBitVect *
StructuralFingerprintChemReaction(const ChemicalReaction &rxn,
const ReactionFingerprintParams &params = DefaultStructuralFPParams));
```

- Available in C++ as well as Python interface

NOVARTIS

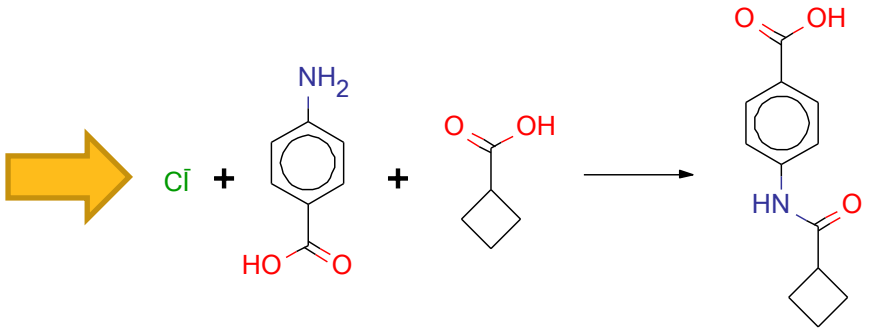
Some benchmarking of the substructure FP

Use it (4K, PatternFP) to scan the Patent database

- Experiment 1: scan for a special reaction

Input:

[Cl–].[NH2:1][c:2]1[cH:3][cH:4][c:5]([C:6](=[O:7])[OH:8])[cH:9][cH:10]1.[O:17]=[C:16](O)[CH:12]1[CH2:13][CH2:14][CH2:15]1>>[O:7]=[C:6]([OH:8])[c:5]1[cH:4][cH:3][c:2]([NH:1][C:16](=[O:17])[CH:12]2[CH2:13][CH2:14][CH2:15]2)[cH:10][cH:9]1



- Using a sequential scan of the database (substructure matching):
  - Execution time: 120948.864 ms, Number of matches: 1
- Using the new GiST-index and the substructure FP:
  - Goal: reduce the number of substructure matchings
  - Execution time: 93.986 ms (0.08 % of the time)
  - Number of reactions after the scan: 13 (0.0012 % of the whole database)

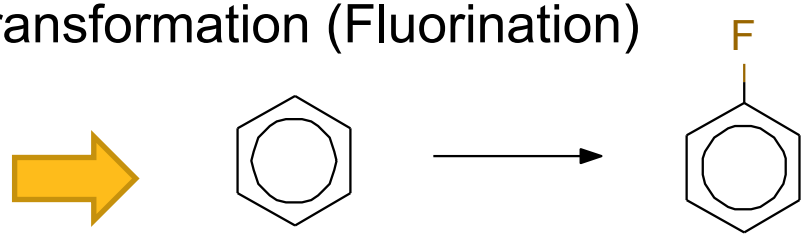
NOVARTIS

Some benchmarking of the substructure FP

Use it (4K, PatternFP) to scan the Patent database

- Experiment 2: scan for a generic transformation (Fluorination)

Input: c1ccccc1>>c1cccc(F)c1



- Using a sequential scan of the database (substructure matching):
  - Execution time: 154569.851 ms
  - Number of matches: 144107
- Using the new GiST-index and the substructure FP:
  - Execution time: 21299.675 ms (14 % of the time)
  - Number of reactions after the scan: 146217 (13 % of the whole database)

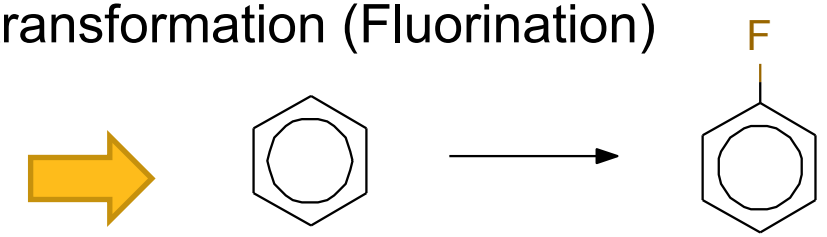
→ only 2110 were additionally removed by the substructure match

Some benchmarking of the substructure FP

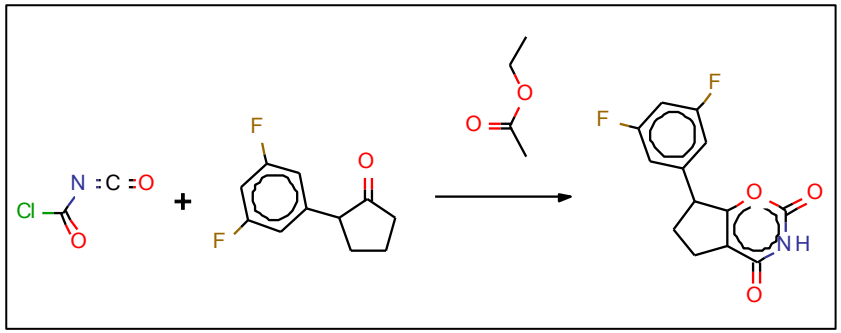
Ok, we are fast now, but do we really get back the transformation?

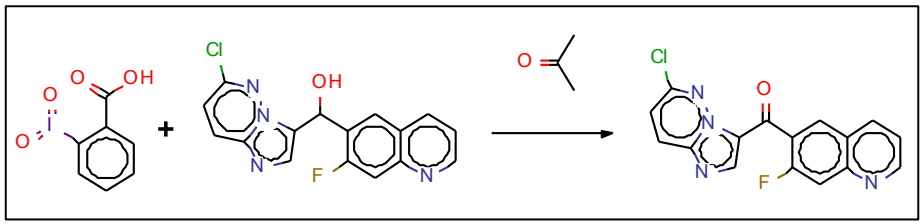
- Experiment 2: scan for a generic transformation (Fluorination)

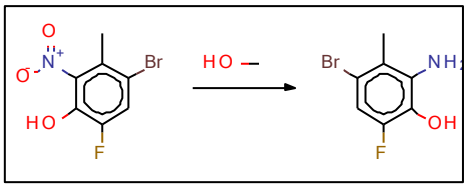
Input: c1ccccc1>>c1cccc(F)c1



- Number of matches: 144107 → have a look at the reactions







→ This is not really what we wanted to have!

NOVARTIS

What else you could find in the cartridge?

- Calculate reaction FP and search for similar reactions
- Using the cartridge in the IPython notebook, you can directly visualize your results

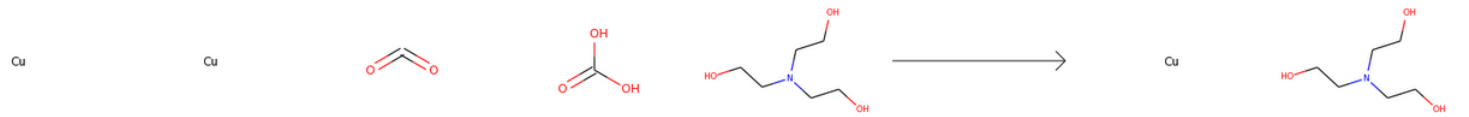
```
In [30]: data = %sql select smiles from rxn_smiles limit 3;
         data
```

3 rows affected.

Out[30]:

smiles
[Cu].[Cu+2:5].O=C=O.O=C([O-])[O-].[OH:15][CH2:14][CH2:13][N:12]([CH2:16][CH2:17][OH:18])[CH2:11][CH2:10][OH:9]>O.CC(=O)O>[Cu:5].[OH:9][CH2:10][CH2:11][N:12]([CH2:13][CH2:14][OH:15])[CH2:16][CH2:17][OH:18]
[Br].[CH3:14][CH:12]1[CH2:11][C:10]([CH3:15])([CH3:16])[CH2:9][C:8]([C:17]#[N:18])([NH:7][NH:6][C:2]([CH3:3])([CH3:4])[CH3:5])[CH2:13]1>>[CH3:14][CH:12]1[CH2:11][C:10]([CH3:15])([CH3:16])[CH2:9][C:8]([C:17]#[N:18])([N:7]=[N:6][C:2]([CH3:5])([CH3:3])[CH3:4])[CH2:13]1
[Cl-].[NH2:1][c:2]1[cH:3][cH:4][c:5]([C:6](=[O:7])[OH:8])[cH:9][cH:10]1.[O:17]=[C:16](O)[CH:12]1[CH2:13][CH2:14][CH2:15]1>>[O:7]=[C:6]([OH:8])[c:5]1[cH:4][cH:3][c:2]([NH:1][C:16](=[O:17])[CH:12]2[CH2:13][CH2:14][CH2:15]2)[cH:10][cH:9]1

```
In [42]: smis = [removeMappingNumbersFromSmiles(str(x)) for x in data.DataFrame()['smiles']]
         data2 = pd.DataFrame(smis, columns=['smi'])
         AddReactionColumnToFrame(data2, smilesCol='smi', reactCol = 'Reaction')
         data2 = data2.drop('smi',1)
         from IPython.core.display import HTML,display
         display(HTML(data2.to_html()))
```

	Reaction
0	
1	

NOVARTIS

Open issues in the reaction functionality

What is missing, what should be improved?

- Improve the substructure matching for reactions
- Become more independent of the atom-mapping concerning the determination of agent
- Provide the new transformation and agent FP in the upcoming RDKit release
- Some better depiction of reactions, similarity maps for reactions
- Suggestions? Wishes?

Acknowledgements

- NIBR
  - Greg Landrum
  - Nikolas Fechner
  - Mike Tarselli
  - Novartis & NIBR Education office for funding

- NextMove Software:
  - Roger Sayle
  - Daniel Lowe

NOVARTIS

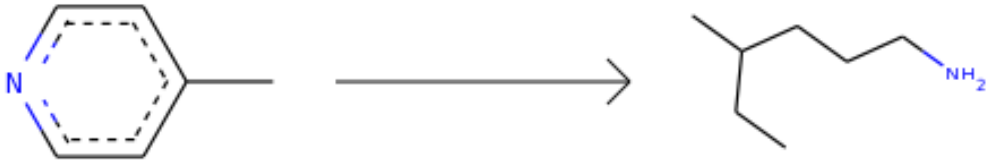
Inclusion of agents in reactions

Adaptions to parsers and writers

- MDL parser/writer:
  - Supports ChemAxon's new extension to MDL RXN files format (third count field for agents)
- Daylight parser/writer:
  - Agents in reaction smiles/smarts are now included
  - Option to move unmapped reactant or product molecules to agents after parsing
- New function to convert molecules in reactions
  - RXN role of the molecule must be defined, e.g., in the eighth property column of the RXN file

Some new things change established behavior

- Intra-molecular bond breaks are now supported in products




```
Cc1ccncc1>>(CCCC.NCC)
```

- Support of redundant atom mapping numbers in products
  - → What about reactants?
- Agents are included when writing SMILES, SMARTS and RXN files

NOVARTIS