Fraggle – A new similarity searching algorithm

Jameed Hussain
Gavin Harper

GlaxoSmithKline

Introduction

- Brief history of the technique
- Why we created (yet) another similarity method
- How it works
- Performance

Brief history of Fraggle

- Was first written in 2008 using the Daylight toolkit
  - Currently 5 years old..
- One of several similarity methods which is in regular use in GSK
  - Method of choice for “boosting” SAR
- Has provided leads for several drug discovery programs

- Re-implemented using RDKit this year

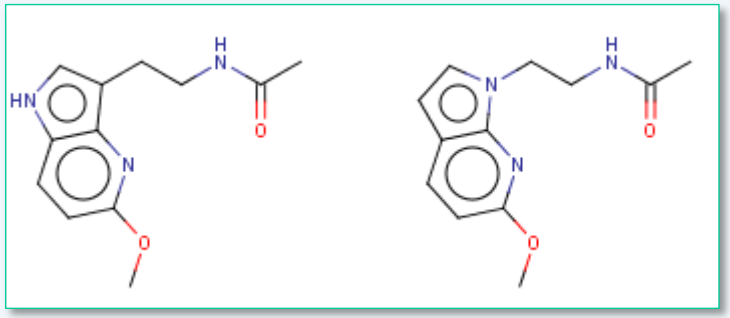
Chemical Similarity Methods

- There is no shortage of chemical similarity methods..
  - Path based fps
  - Morgan fps
  - Topological Torsion / Atom Pairs
  - 2D pharmacophore methods
    - RGs / ErGs.
  - 3D fps
- Why does the world need another ?
  - ...

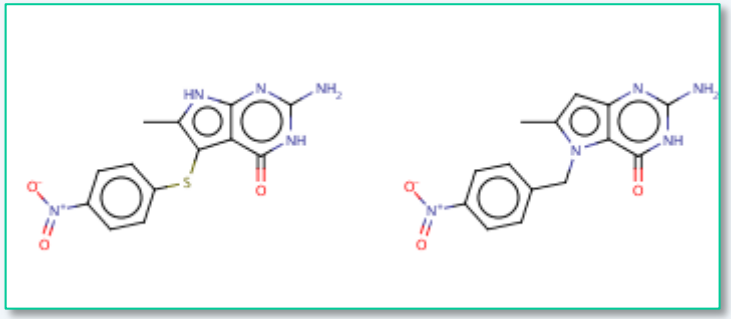
Chemical Similarity Methods

- Why did *we* create another similarity method ?

- Specifically built to fix a particular issue that affects path based fps
  - Small changes in the middle of a molecule
    - Affects other similarity methods too



ChEMBL_11085_A_27 & ChEMBL_11085_A_78
RDK5: 0.42
ECFP4: 0.65
TT: 0.47



ChEMBL_28_A_27 & ChEMBL_28_A_45
RDK5: 0.45
ECFP4: 0.66
TT: 0.48

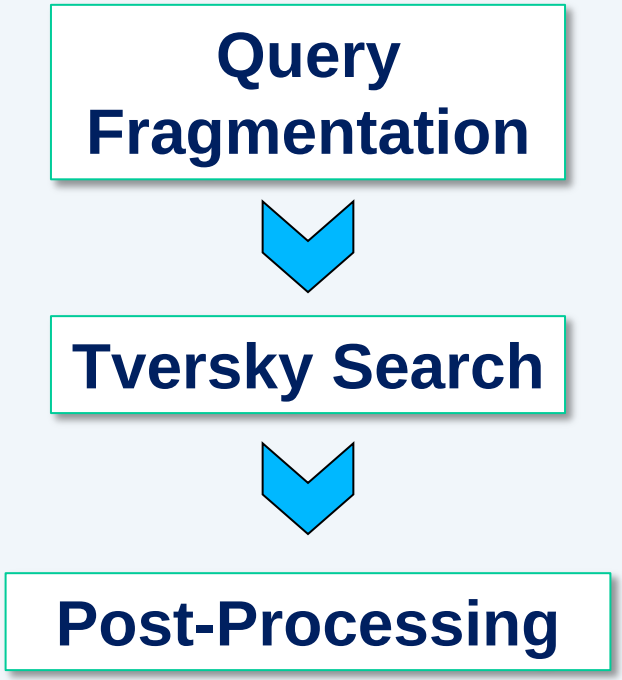
Riniker, S., & Landrum, G. A. (2013). Open-source platform to benchmark fingerprints for ligand-based virtual screening. *Journal of cheminformatics*, 5(1), 26.

Substructure searching

- Similarity and Substructure searching are complementary
- Substructure searching has a requirement of knowing which part of molecule is important
  - Fixed as the substructure, rest of compound can be anything
- Similarity searching has no requirement of a fixed substructure
  - “Most” of the compound needs to be the same
- How can we capture some of the benefits of a substructure search
  - “**Large changes in a small part of a molecule**”

Fraggle – how does it work?

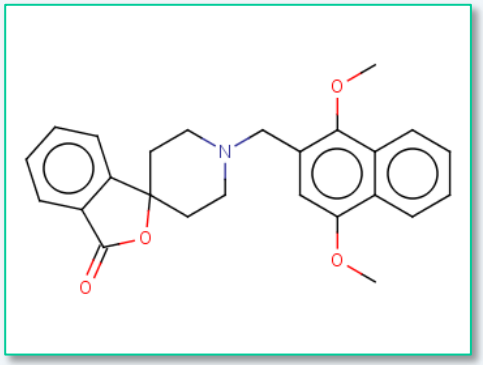
- Fraggle works in three steps:



Query fragmentation

- "Make the method behave like a substructure search"
- If you don't know which part of the molecule is important how do you know which substructure to search with ?
  - Use "all the interesting" substructures

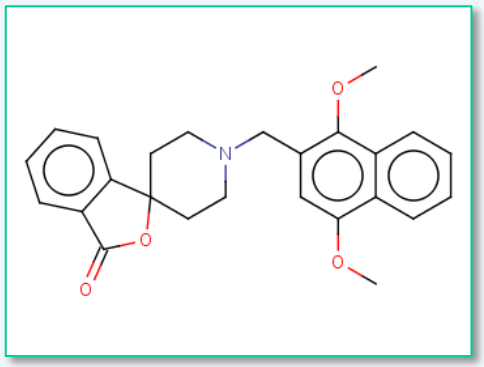
- Algorithm used to fragment query molecule and select the "interesting" substructures
  - Employs simple rules
  - Tries to capture all the constituent rings in a query molecule



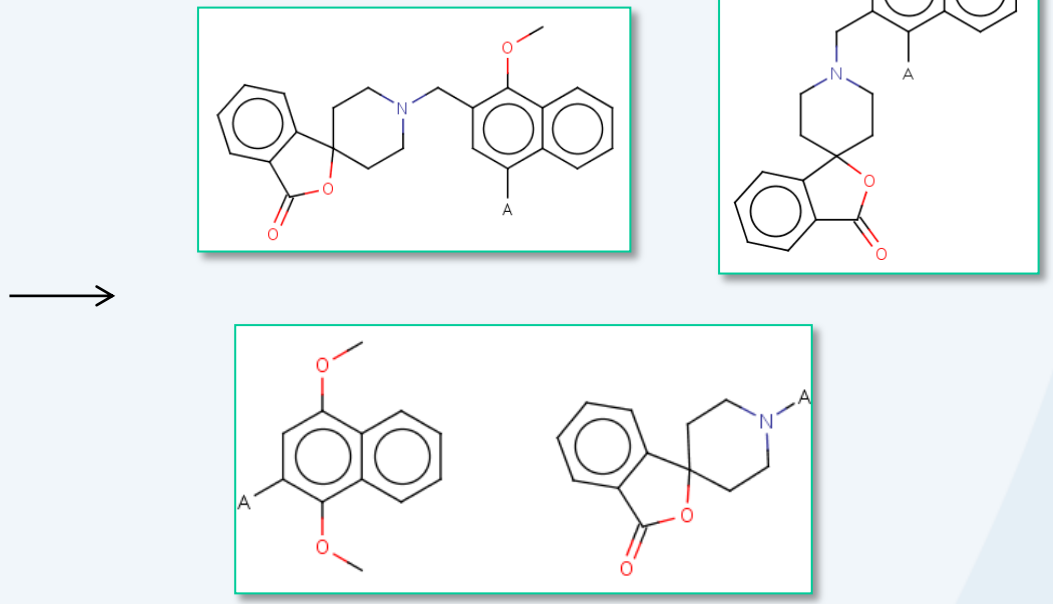
ChEMBL_11265_A_41

Fragmentation Algorithm – Acyclic cuts

- Enumerate all the single acyclic bond cuts
  - Discard fragmentations where you only chop a single atom off
  - Keep fragment if >60% of query molecule
- Enumerate all the double acyclic bond cuts
  - Discard fragmentations where you only chop a single atom off
  - Keep the two fragments with one attachment point
    - Needs to be >60% of query molecule

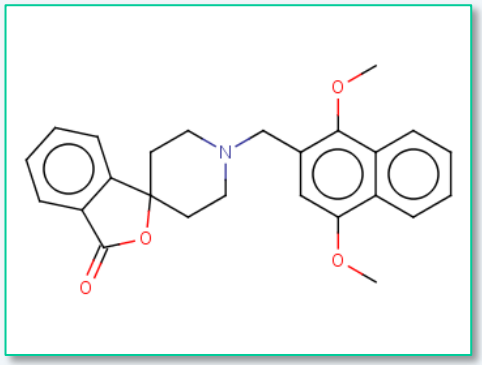


ChEMBL_11265_A_41

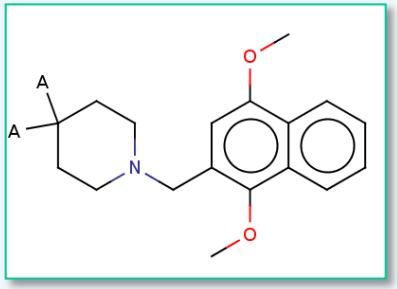


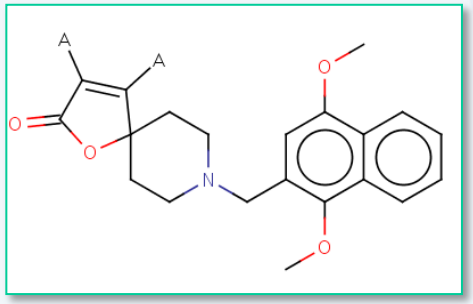
Fragmentation Algorithm – Ring cuts

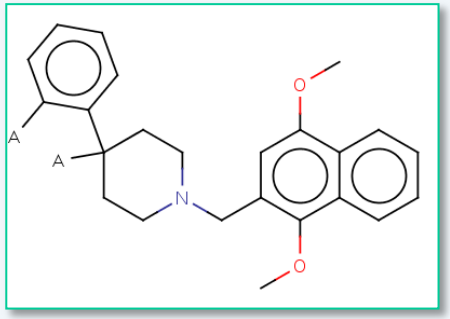
- For compounds with fused / spiro ring systems
- Enumerate all single “ring cuts” - cut at the 2 exocyclic bonds
  - Need to be >40% of query molecule
- Enumerate all single “ring cuts” with an acyclic bond cut
  - Needs to be >60% of query molecule

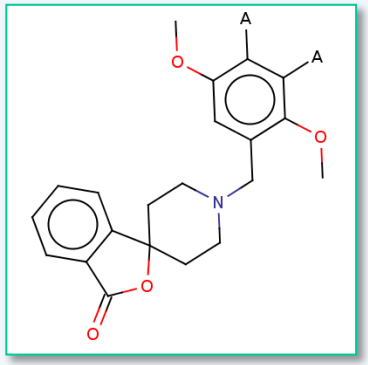


ChEMBL_11265_A_41



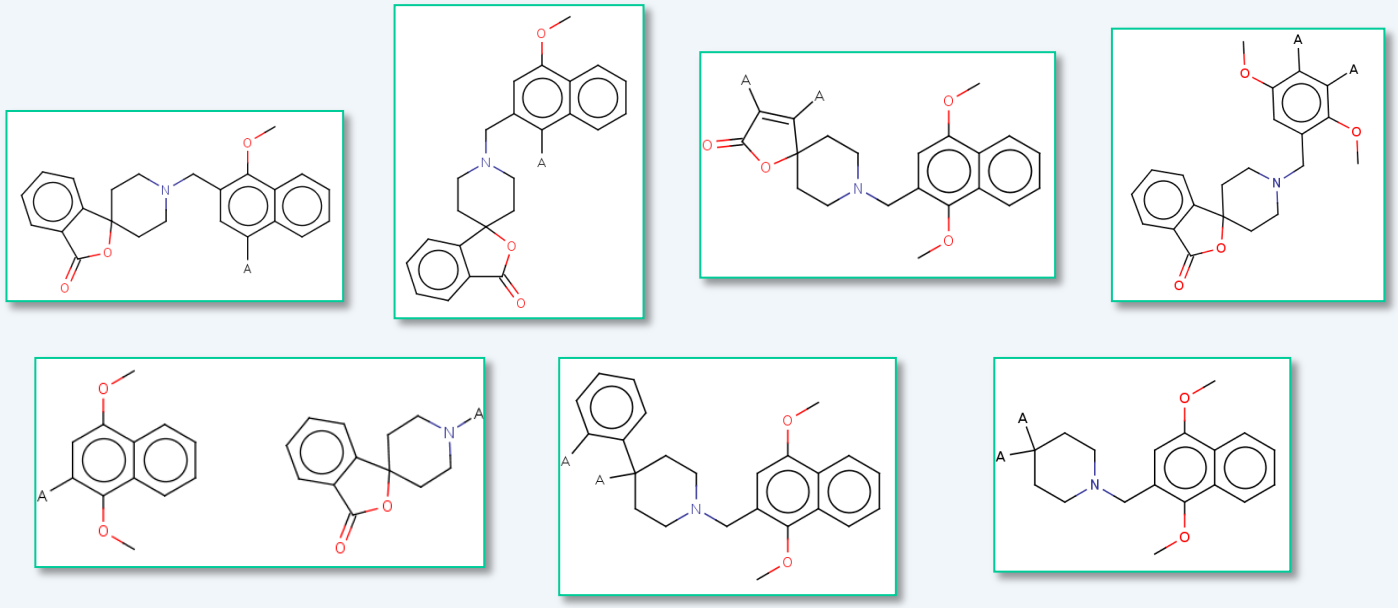






Tversky Search

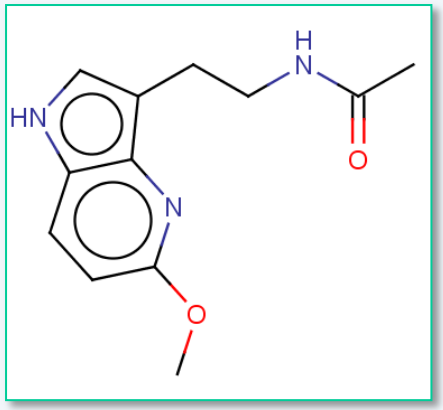
- For each fragmentation carry out a Tversky search against the database
  - ChemAxon FP
    - Alpha=0.95, Beta=0.05 (“substructure similarity”)
    - Tversky similarity cut-off=0.9
- Tversky search gives superior results compared to substructure searching (more “fuzziness”)



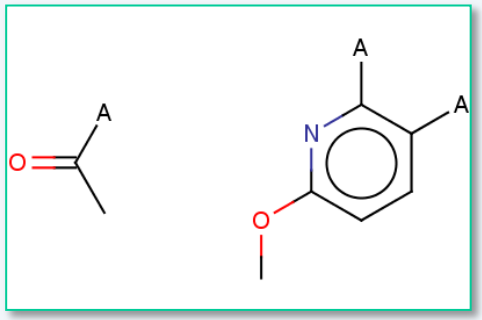
GlaxoSmithKline

Post Processing

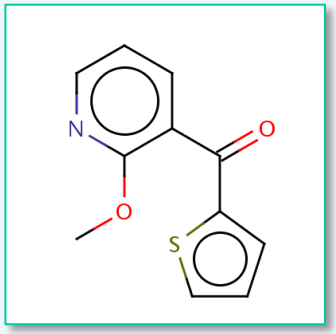
- Tversky search can retrieve results which are uninteresting with respect to the original query molecule



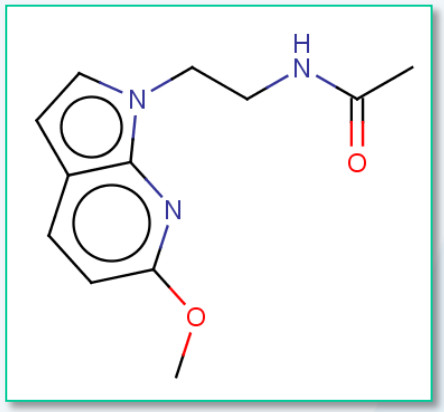
ChEMBL_11085_A_27



Query Fragmentation

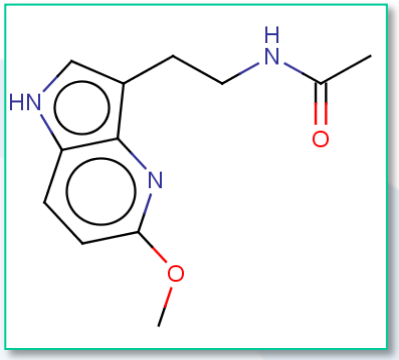


ChEMBL_zinc_D_3054
Tversky: 0.90

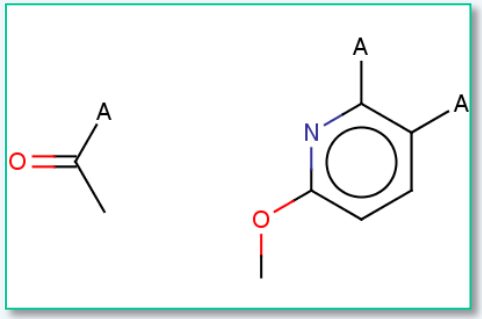


ChEMBL_11085_A_78
Tversky: 0.90

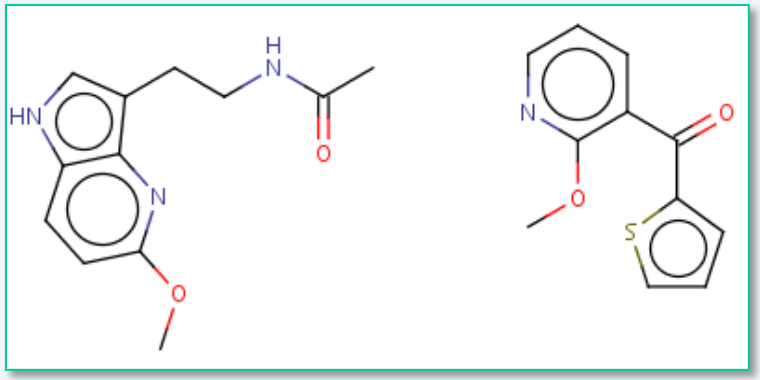
Post Processing



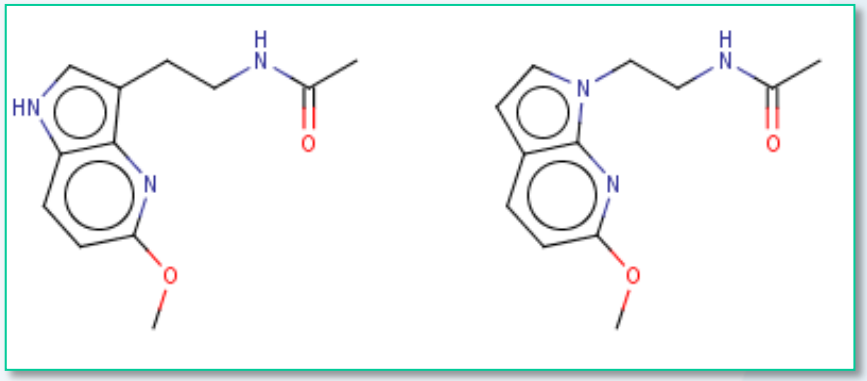
ChEMBL_11085_A_27



Query Fragmentation



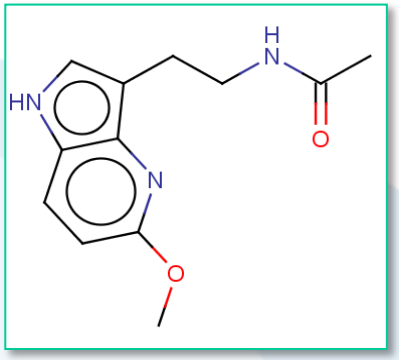
RDK5 Similarity: 0.36



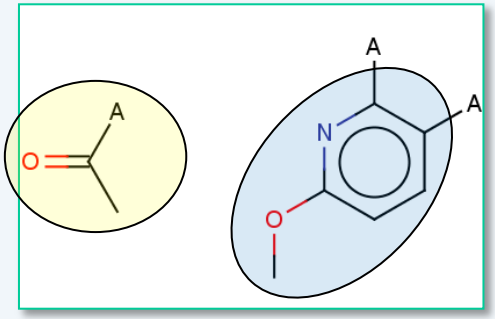
RDK5 Similarity: 0.42

GlaxoSmithKline

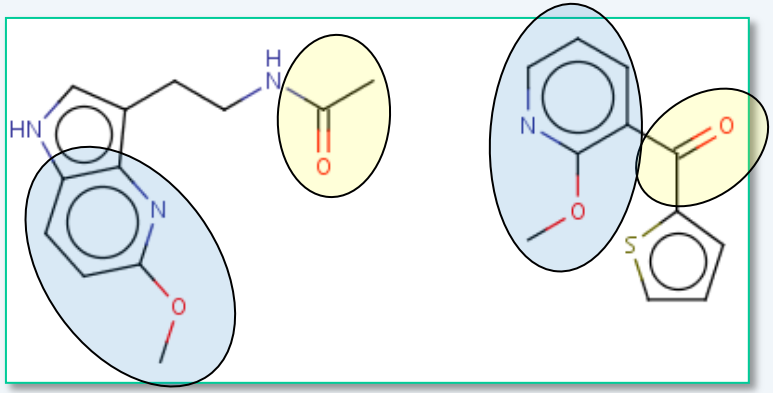
Post Processing

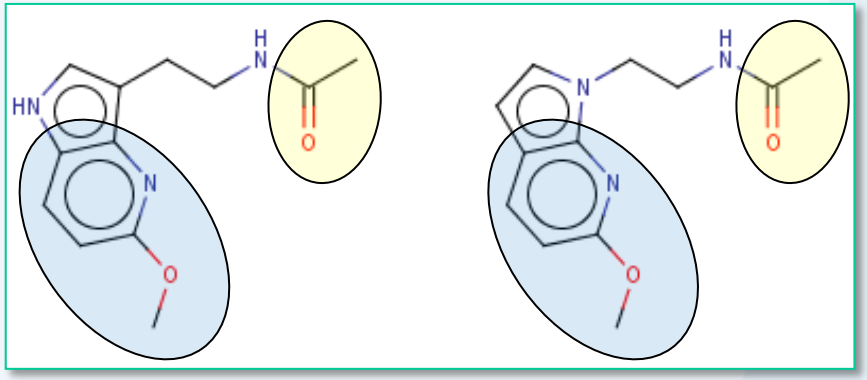


ChEMBL_11085_A_27



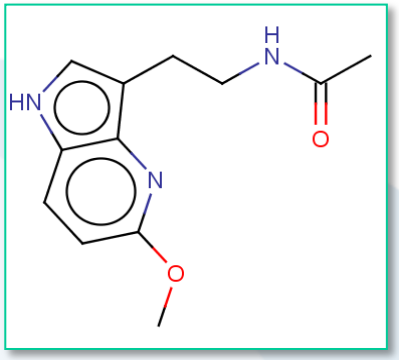
Query Fragmentation



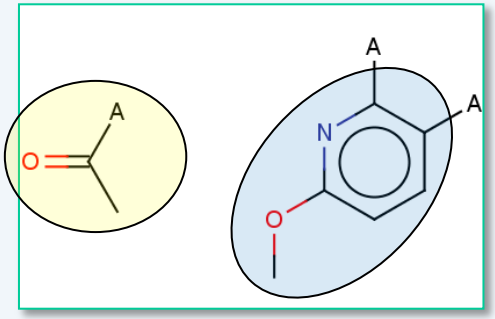


GlaxoSmithKline

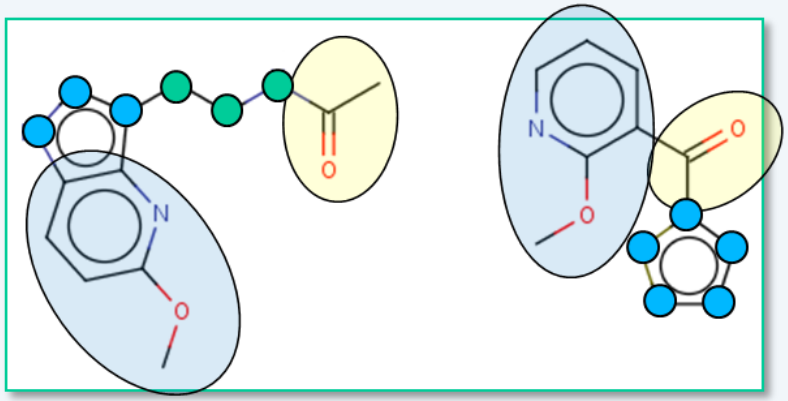
Post Processing



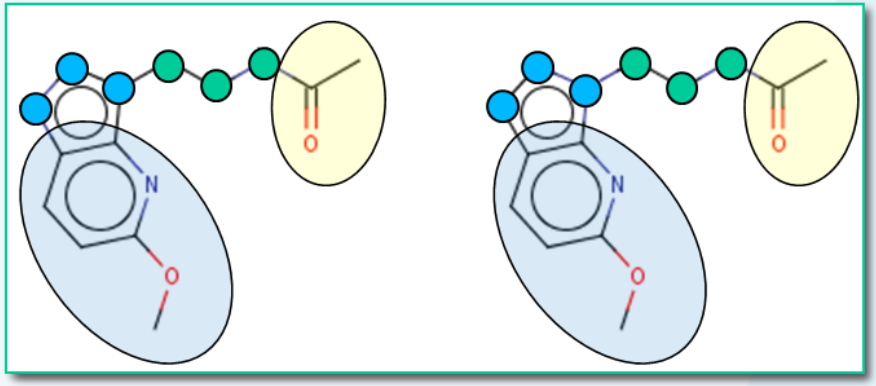
ChEMBL_11085_A_27



Query Fragmentation



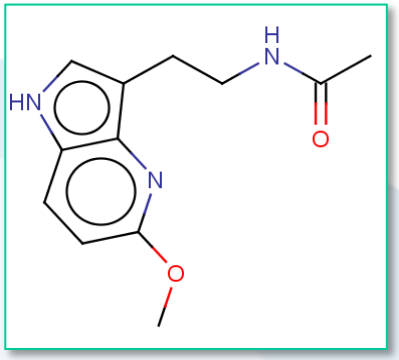
False Positive
RDK5 Similarity: 0.25



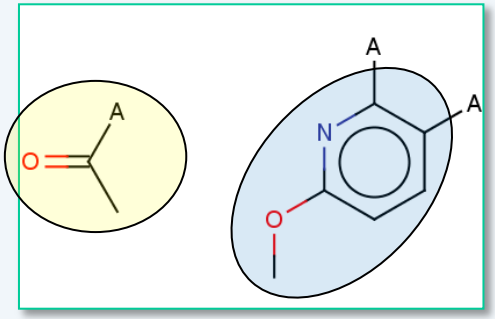
High Scoring Match
RDK5 Similarity: 1.0

GlaxoSmithKline

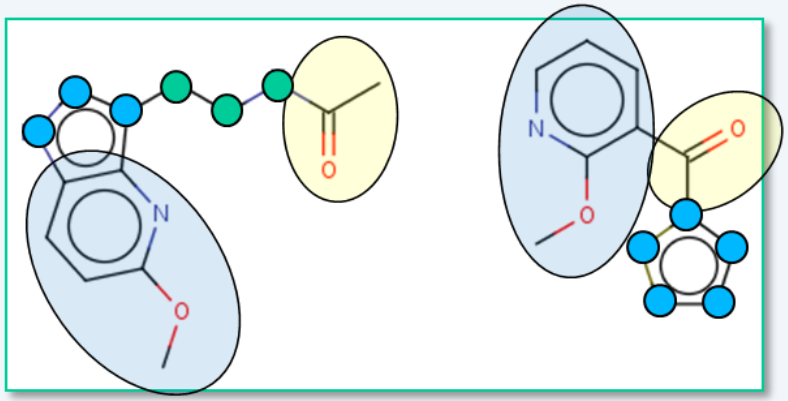
Post Processing



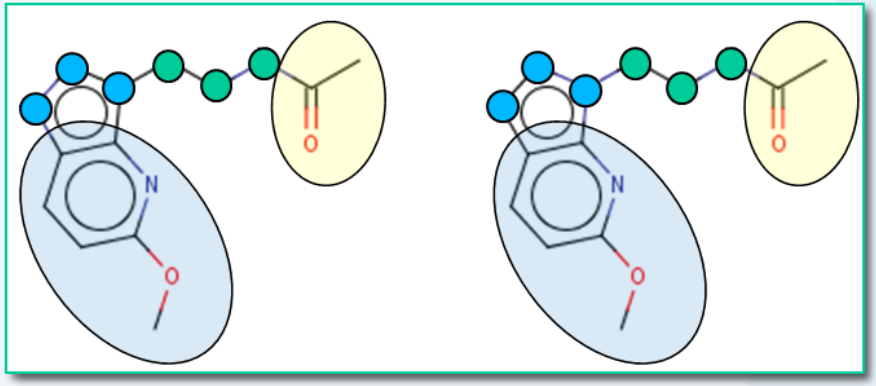
ChEMBL_11085_A_27



Query Fragmentation



False Positive
Fraggle Similarity: 0.36



High Scoring Match
RDK5 Similarity: 1.0

GlaxoSmithKline

Post Processing – gory details...

- Post Matching algorithm:
  - For the query fragmentation and the db molecule pair
    - Map the fragmentation on the molecule
    - Modify the non-matching atoms of molecule
      - Aromatic atoms become *
      - Aliphatic atoms become Sc
  - Carry out a RDK5 fp Tanimoto similarity using these “modified” query and db molecule
    - Done for every “fragmentation” and the highest similarity is selected
  - Compare the highest similarity with the RDK5 fp Tanimoto on the unmodified query and db molecule
    - Pick the highest to give the Fraggle similarity

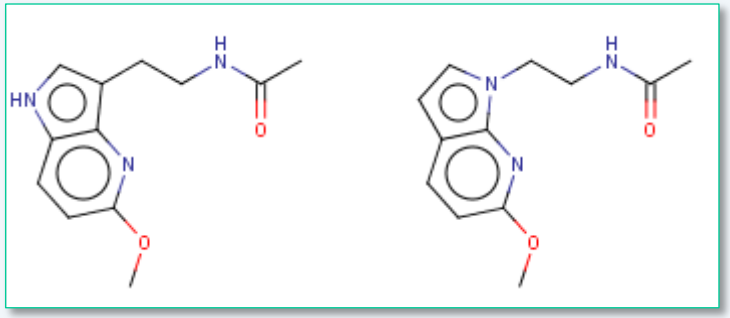
Fragment Mapping

- Matching of the fragments on retrieved and query molecules carried using partial fingerprints and Tversky similarity
  - A partial fingerprint (pFP) of an atom (in a compound) are the bits it sets in the compound fingerprint
    - Compare the pFP of every atom of a molecule against the FP of the fragments
      - Tversky >0.8 is considered a match
- Partial fingerprints with Tversky allows for very computationally cheap alignments
  - Crude but fast
- Perfectly adequate for this application
  - “Fuzziness” is good

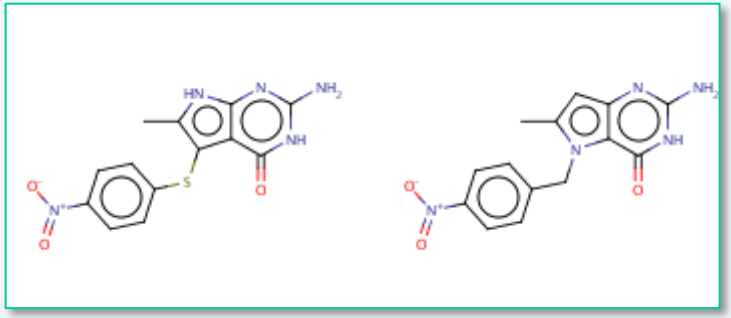
What types of compounds does Fraggle find?

- Not as sensitive to changes in the middle of a molecule

- Fraggle similarity for the pairs of cmpds is below is 1:



ChEMBL_11085_A_27 & ChEMBL_11085_A_78
Fraggle: 1.0
RDK5: 0.42
ECFP4: 0.65
TT: 0.47

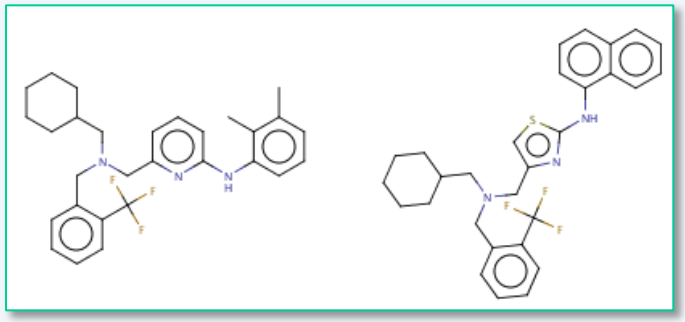


ChEMBL_28_A_27 & ChEMBL_28_A_45
Fraggle: 1.0
RDK5: 0.45
ECFP4: 0.66
TT: 0.48

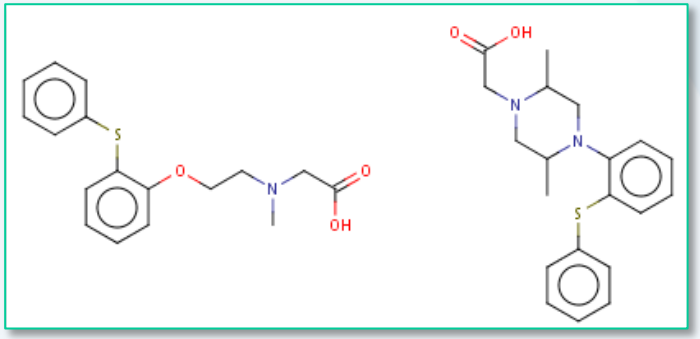
GlaxoSmithKline

What types of compounds does Fraggle find?

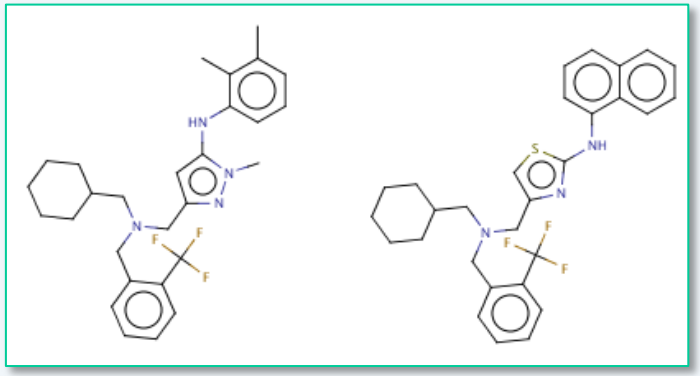
- **“Large changes in a small part of a molecule”**



ChEMBL_10579_A_78 & ChEMBL_10579_A_39
Fraggle: 0.89
RDK5: 0.62
ECFP4: 0.8
TT: 0.78



ChEMBL_11682_A_2 & ChEMBL_11682_A_52
Fraggle: 0.86
RDK5: 0.38
ECFP4: 0.64
TT: 0.57

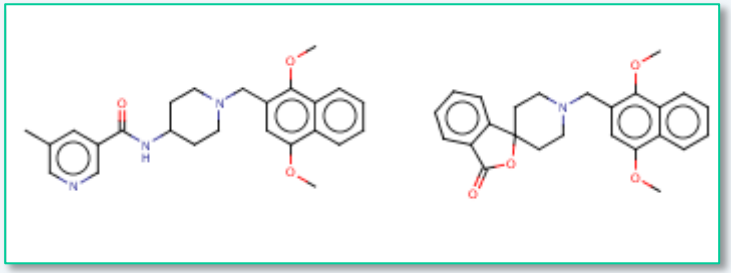


ChEMBL_10579_A_16 & ChEMBL_10579_A_39
Fraggle: 0.89
RDK5: 0.52
ECFP4: 0.75
TT: 0.68

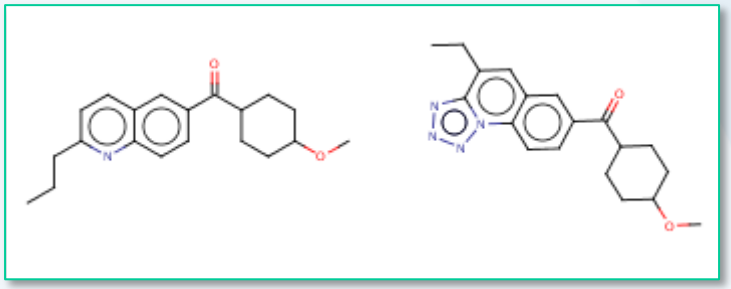
GlaxoSmithKline

What types of compounds does Fraggle find?

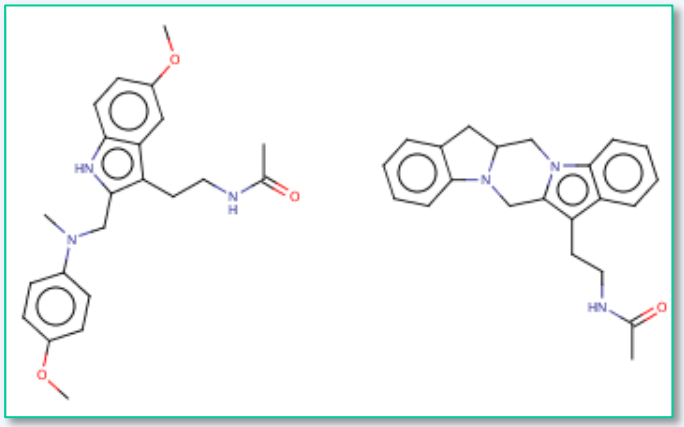
- Performs very well with fused and spiro queries



ChEMBL_11265_A_64 & ChEMBL_11265_A_41
Fraggle: 0.81
RDK5: 0.49
ECFP4: 0.66
TT: 0.59



ChEMBL_11279_A_53 & ChEMBL_11279_A_35
Fraggle: 0.92
RDK5: 0.63
ECFP4: 0.7
TT: 0.61



ChEMBL_11085_A_97 & ChEMBL_11085_A_74
Fraggle: 0.81
RDK5: 0.64
ECFP4: 0.44
TT: 0.31

GlaxoSmithKline

Performance - AUC

- Acknowledge Sereina Riniker and Greg Landrum work
  - Riniker, S., & Landrum, G. A. (2013). Open-source platform to benchmark fingerprints for ligand-based virtual screening. *Journal of cheminformatics*, *5*(1), 26.
- Compared Fraggle, RDK5, TT, ECFP4, MACCS, ECFP0
- Results from post-hoc Friedman tests of the average rank:

	RDK5	Fraggle	ECFP4	MACCS	ECFP0
TT	X	O	-	-	-
RDK5		X	O	-	-
Fraggle			X	-	-
ECFP4				O	-
MACCS					-
ECFP0					

X: No statistical significant difference
O: Difference around the confidence level
- : Statistically significant difference

GlaxoSmithKline

Performance – BEDROCK20

- Results from post-hoc Friedman test of the average rank:

	ECFP4	RDK5	Fraggle	MACCS	ECFP0
TT	X	X	O	-	-
ECFP4		X	X	-	-
RDK5			X	-	-
Fraggle				-	-
MACCS					-
ECFP0					

X: No statistical significant difference
O: Difference around the confidence level
- : Statistically significant difference

- Fraggle “in the mix” with the best performing methods
  - Benefits from RDK5 for AUC metric
  - Similar performance to ECFP4,RDK5 (and TT) for BEDROCK20

Correlation with other methods

- Take all actives from evaluation platform
  - For actives in each dataset generate similarity matrix
    - How does the similarity ranking correlate (Spearman) between methods?
- Fraggle worth running with other top performing methods

ChEMBL:

	AP	ECFP4	Fraggle	RDK5	RDK6	RDK7	TT
AP	---	0.84	0.68	0.68	0.63	0.52	0.77
ECFP4	0.84	---	0.64	0.65	0.56	0.43	0.84
Fraggle	0.68	0.64	---	0.87	0.77	0.59	0.60
RDK5	0.68	0.65	0.87	---	0.89	0.71	0.64
RDK6	0.63	0.56	0.77	0.89	---	0.93	0.53
RDK7	0.52	0.43	0.59	0.71	0.93	---	0.39
TT	0.77	0.84	0.60	0.64	0.53	0.39	---

MUV:

	AP	ECFP4	Fraggle	RDK5	RDK6	RDK7	TT
AP	---	0.79	0.45	0.48	0.39	0.30	0.69
ECFP4	0.79	---	0.41	0.47	0.35	0.22	0.78
Fraggle	0.45	0.41	---	0.67	0.52	0.37	0.35
RDK5	0.48	0.47	0.67	---	0.86	0.67	0.48
RDK6	0.39	0.35	0.52	0.86	---	0.93	0.35
RDK7	0.30	0.22	0.37	0.67	0.93	---	0.21
TT	0.69	0.78	0.35	0.48	0.35	0.21	---

1.00
0.40
0.20

Possible Enhancements

- The method has a number of “tuneable” parameters
  - Size of fragments selected for Tversky searching
  - FP and parameters to use for Tversky searching against db
    - Does RDK5 give better results than ChemAxon FP?
    - What is the optimum alpha, beta and cut-off parameters to use
  - Tversky parameters for pFP comparison
- The parameters chosen are based on very limited datasets and our judgement
  - Balance speed vs retrieval performance

- What happens if I drop the Tversky db searching step?
  - “Post process” every cmpd in db

- Evaluation platform provides a more rigorous way to determine the “best general” parameters

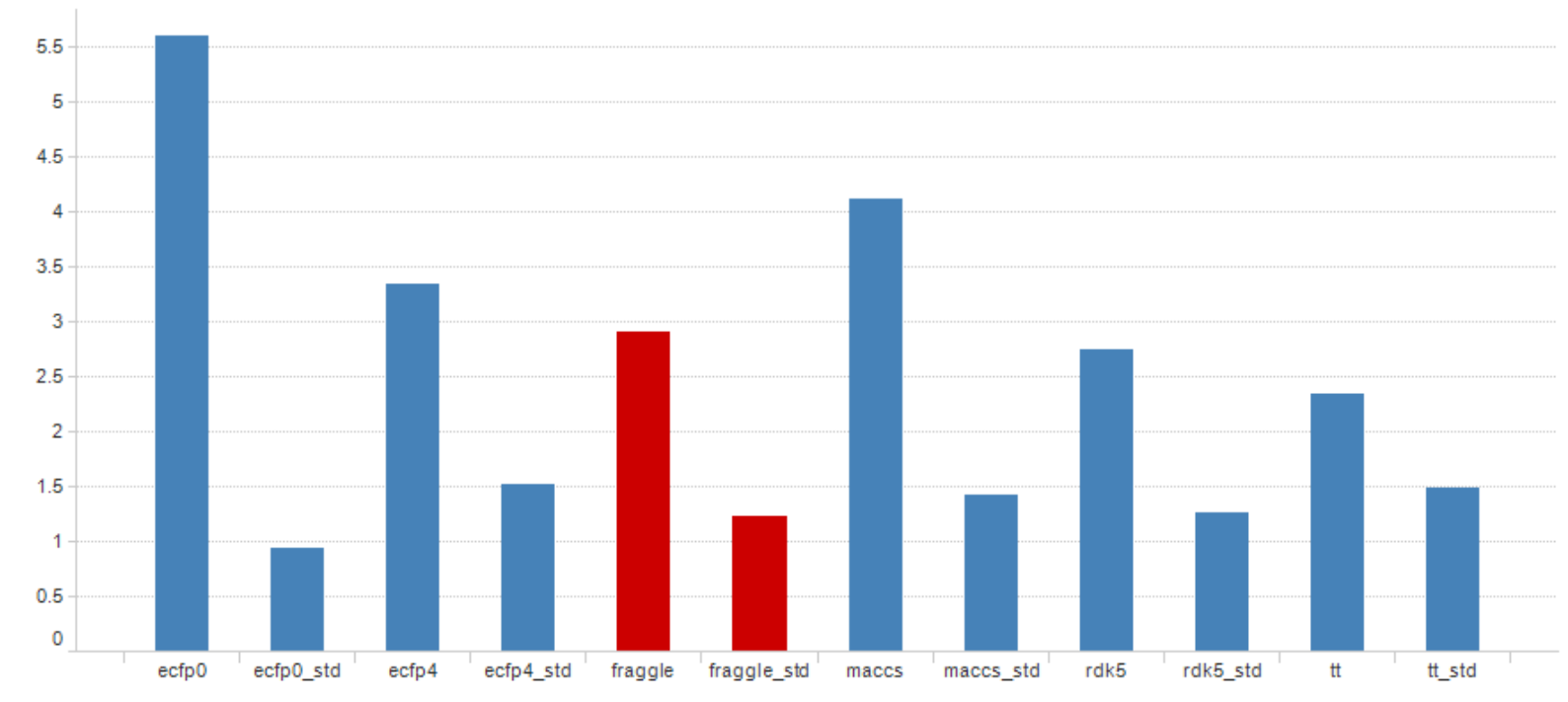
Summary

- Brief history of the technique
- Why we created (yet) another similarity method
- How it works
- Performance

Back-up Slides

Performance

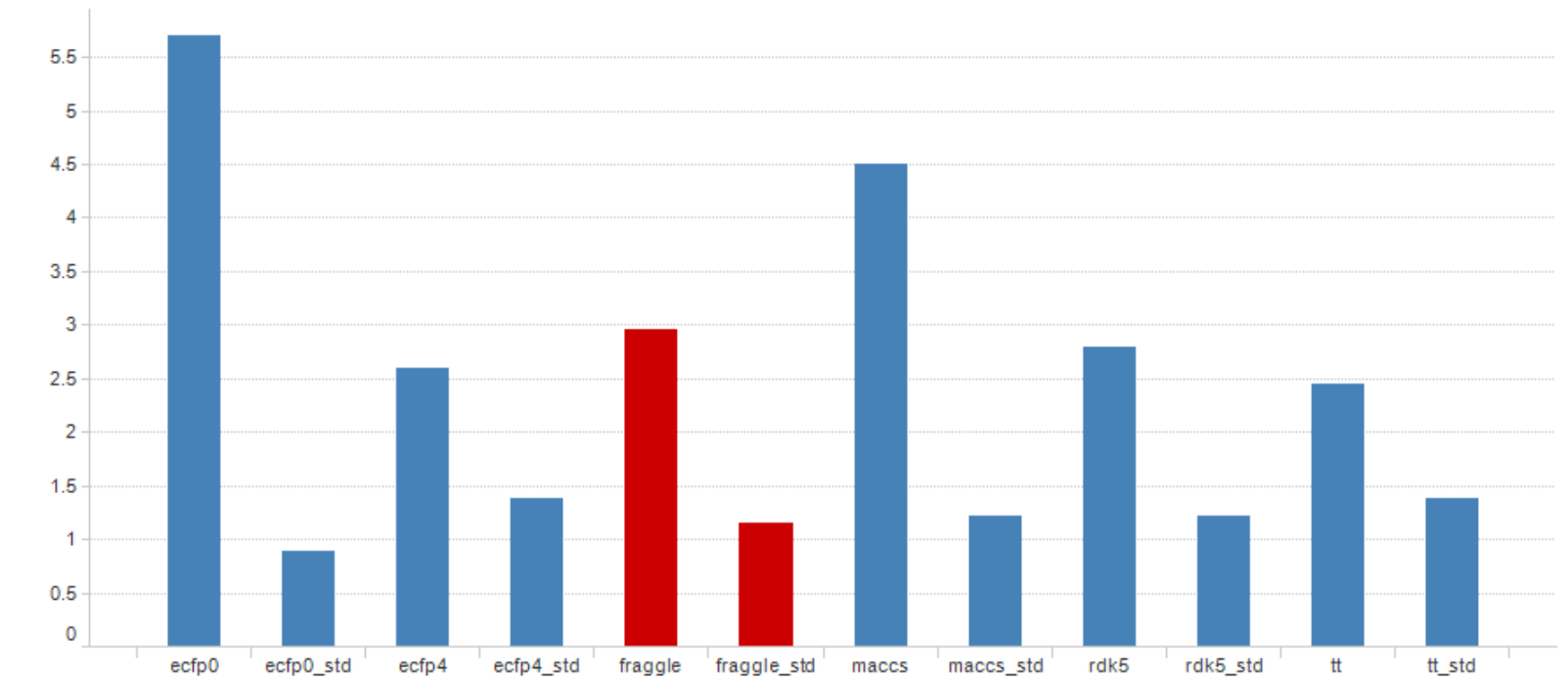
- AUC Rankings:



Smaller is better

Performance

- BEDROCK20 Rankings:



Smaller is better

Correlation with other methods

- Take all actives from evaluation platform
  - For actives in each dataset generate similarity matrix
    - How does the similarity ranking correlate (Spearman) between methods?

DUD:

	AP	ECFP4	Fraggle	RDK5	RDK6	RDK7	TT
AP	---	0.94	0.86	0.85	0.83	0.72	0.90
ECFP4	0.94	---	0.88	0.90	0.87	0.74	0.96
Fraggle	0.86	0.88	---	0.93	0.90	0.78	0.86
RDK5	0.85	0.90	0.93	---	0.97	0.85	0.90
RDK6	0.83	0.87	0.90	0.97	---	0.93	0.88
RDK7	0.72	0.74	0.78	0.85	0.93	---	0.75
TT	0.90	0.96	0.86	0.90	0.88	0.75	---

Tversky Metric

- When comparing molecule A and molecule B:

$$\frac{c}{\alpha a + \beta b + c}$$

a *is the count of bits on in mol A but not in mol B.*
b *is the count of bits on in mol B but not in mol A.*
c *is the count of the bits on in both mol A and mol B.*

- α=1 β=0: similarity of molecule B as a superstructure of molecule A
- α=0 β=1: similarity of molecule B as a substructure of molecule A
- α=0.5 β=0.5: Tanimoto similarity