A Brief History of ChemistryFeaturization

(...and some lessons learned along the way)

Rachel Kurchin, PhD

JuliaMolSim minisymposium

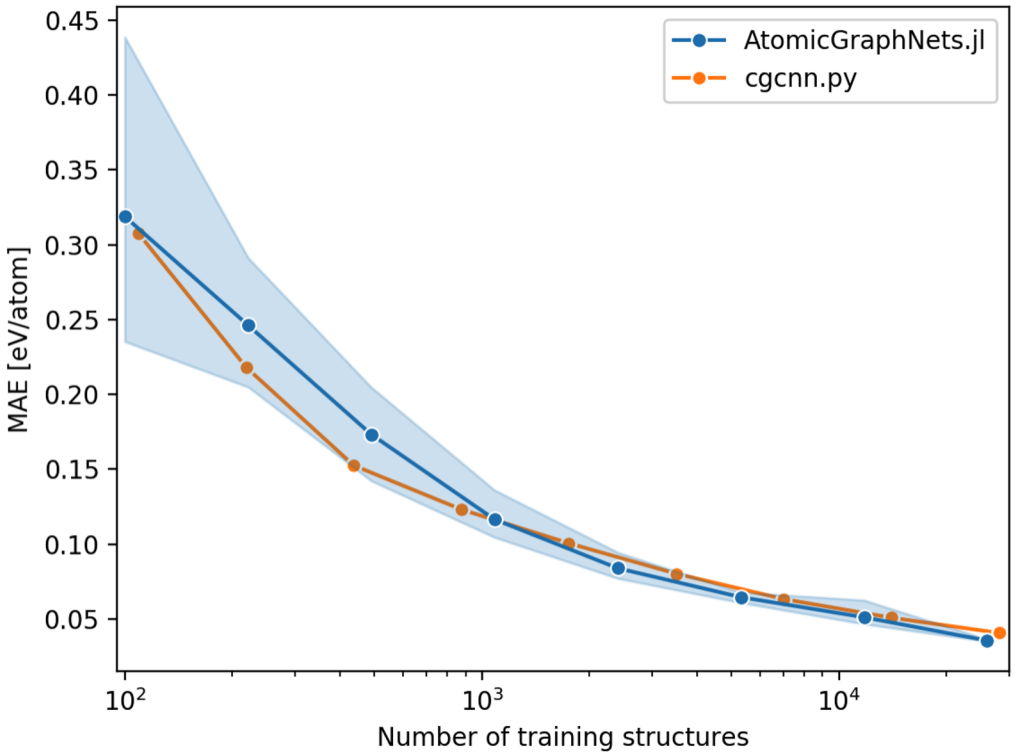
JuliaCon 2022

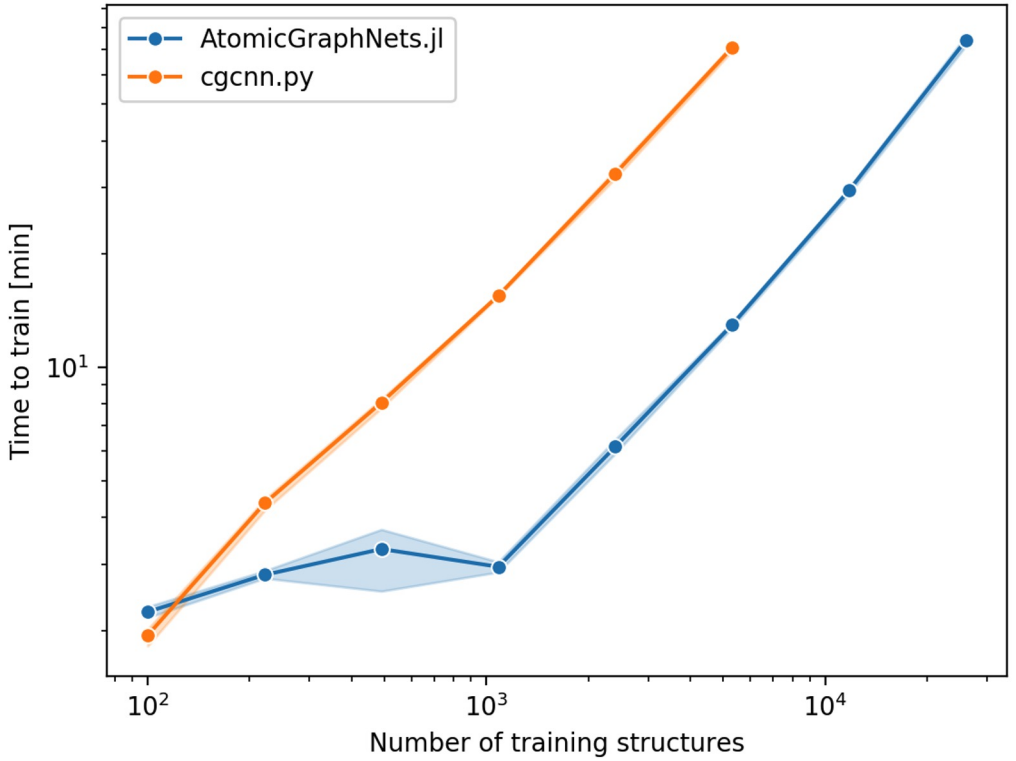

: rkurchin.github.io


: @rachel_kurchin

AtomicGraphNets.jl

- Built on Flux.jl, defines basic layers for convolution and pooling
- Competitive/superior performance relative to cgcnn.py:





Julia model = 21001 trainable params

Python model = 80833 trainable params

https://github.com/Chemellia/AtomicGraphNets.jl

today!

ChemistryFeaturization.jl is born

Jul ‘20

…out of frustration with standard approaches’ opacity and rigidity.

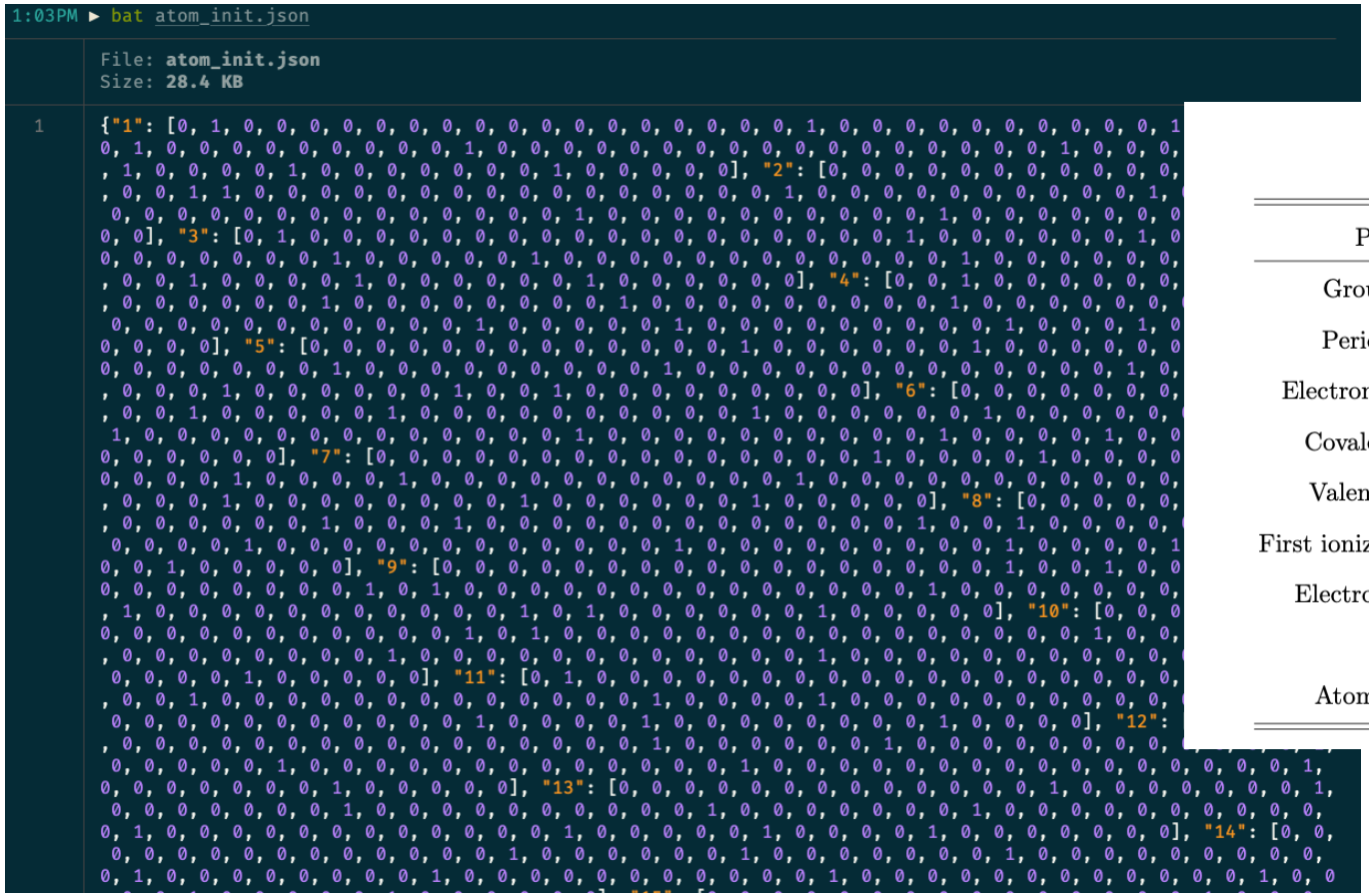


TABLE S2: Properties used in atom feature vector $\boldsymbol{v}_i$

Property	Unit	Range	# of categories
Group number	–	1,2, ..., 18	18
Period number	–	1,2, ..., 9[a]	9
Electronegativity[7, 8]	–	0.5–4.0	10
Covalent radius[3]	pm	25–250	10
Valence electrons	–	1, 2, ..., 12	12
First ionization energy[9][b]	eV	1.3–3.3	10
Electron affinity[10]	eV	-3–3.7	10
Block	–	s, p, d, f	4
Atomic volume[b]	cm^3/mol	1.5–4.3	10

today!

ChemistryFeaturization Design Principles

Jul '20

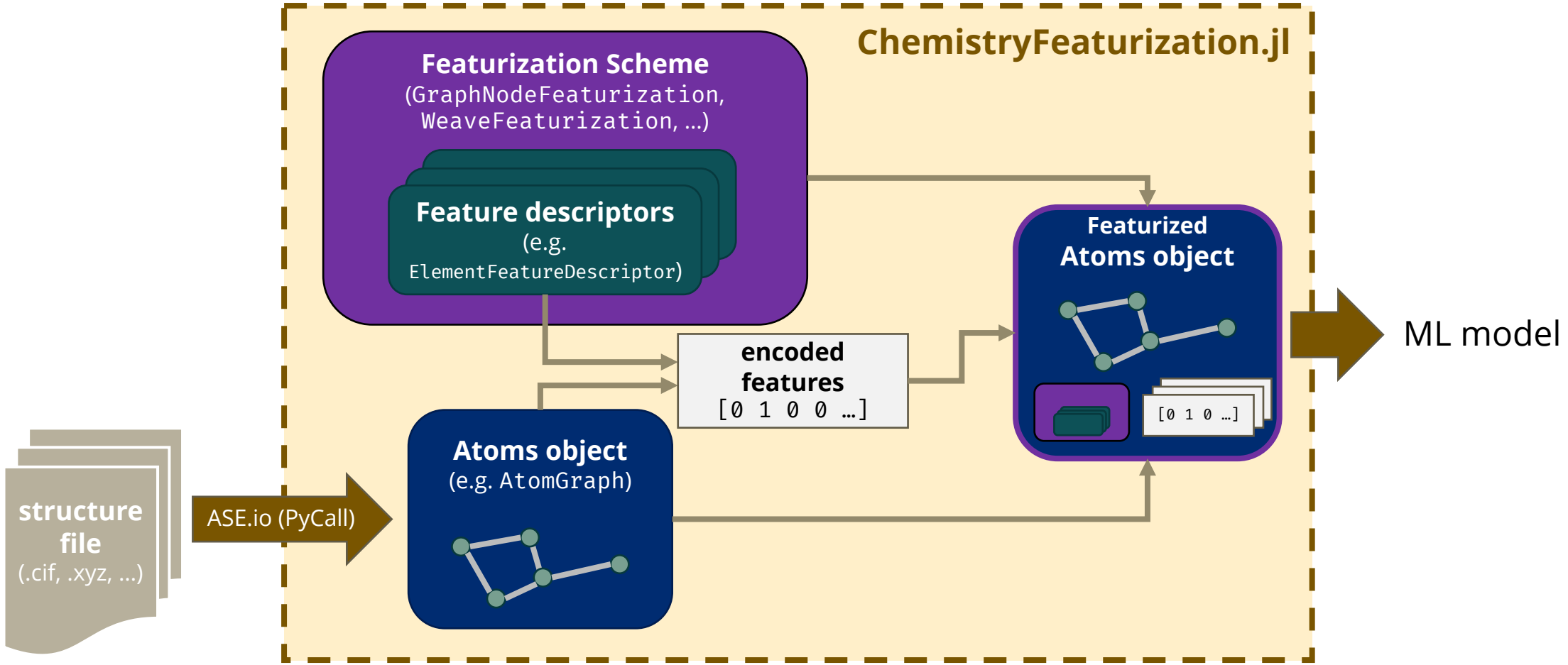
- Transparency: *what information am I encoding, and how?*
- Flexibility: *I want this feature (at this resolution) but not that one*
- Invertibility: *I want to be able to undo my feature encoding!*
- Versatility: *I'd like to use the same API for diverse atomic systems!*
- Extensibility: *…and it should be easy for me to make that happen!*

Concept: "one-stop-shop" for atomic representations, feature specification, encoding/decoding, providing standardized output types for ML models we build

today!

ChemistryFeaturization as of July 2020

Jul '20



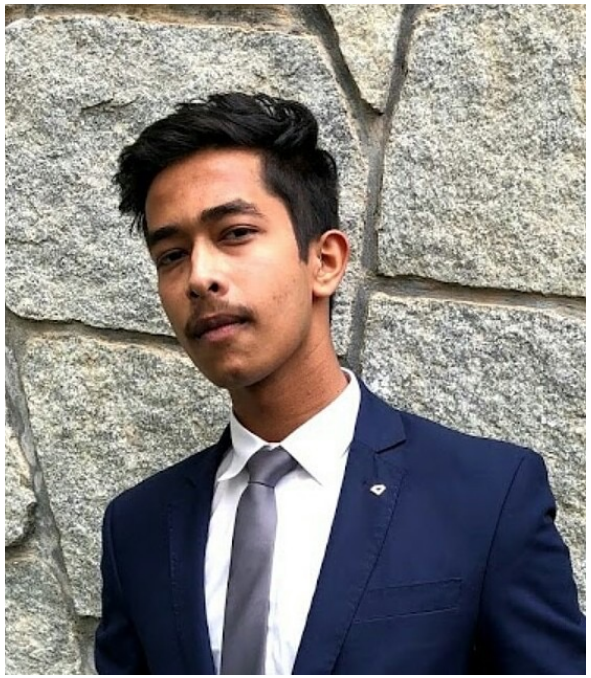
today!

More support!

THE MOLECULAR SCIENCES SOFTWARE INSTITUTE

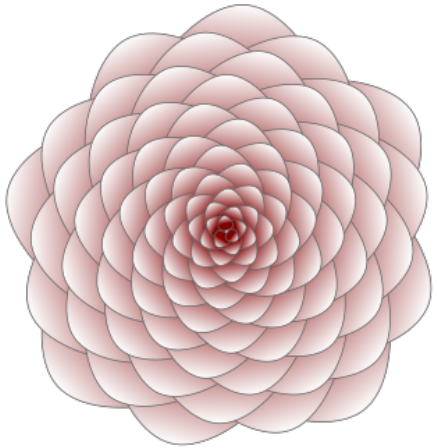
Nov ‘20

Jan ‘21



today!

Introducing…



Chemellia

May ‘21

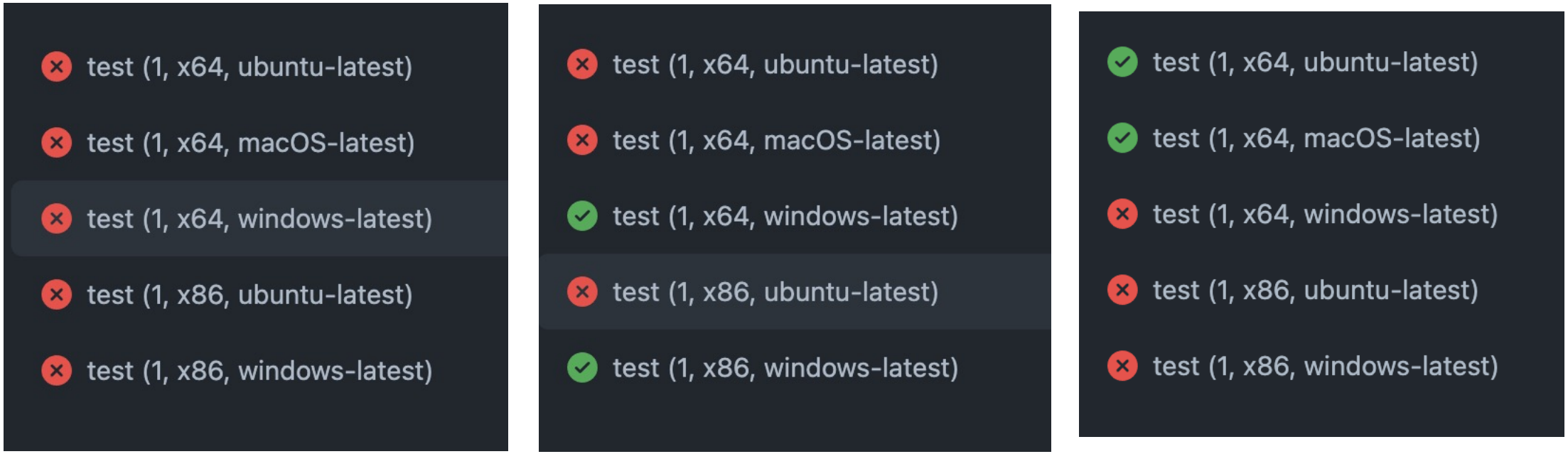
Ecosystem for machine learning with atoms encompassing ChemistryFeaturization, AtomicGraphNets, and more!

https://chemellia.org, https://github.com/Chemellia

today!

ChemistryFeaturization gets bloated...

A representative set of CI results as a PR progresses...



Nov '21

...mainly due to too many deps on other heavy/complicated packages like PyCall/Conda, etc.

today!

...and then slims down!

File: **Project.toml**

```
name = "ChemistryFeaturization"
uuid = "6c925690-434a-421d-aea7-51398c5b007a"
authors = ["Rachel Kurchin <rkurchin@cmu.edu>", "Anant Thazhema
version = "0.6.1"

[deps]
CSV = "336ed68f-0bac-5ca0-87d4-7b16caf5d00b"
Colors = "5ae59095-9a9b-59fe-a467-6f913c188581"
Compose = "a81c6b42-2e10-5240-aca2-a61377ecd94b
Conda = "8f4d0f93-b110-5947-807f-2305c1781a2d"
DataFrames = "a93c6f00-e57d-5684-b7b6-d8193f3e4
DataStructures = "864edb3b-99cc-5e75-8d2d-829cb
Flux = "587475ba-b771-5e3f-ad9e-33799f191a9c"
GraphPlot = "a2cc645c-3eea-5389-862e-a155d00522
Graphs = "86223c79-3864-5bf0-83f7-82e725a168b6"
JSON3 = "0f8b85d8-7281-11e9-16c2-39a750bddbf1"
LinearAlgebra = "37e2e46d-f89d-539d-b4ee-838fcc
MolecularGraph = "6c89ec66-9cd8-5372-9f91-fabc5
NearestNeighbors = "b8a86587-4115-5ab1-83bc-aa9
PyCall = "438e738f-606a-5dbb-bf0a-cddfbfd45ab0"
Serialization = "9e88b42a-f829-5b0c-bbe9-9e9231
SimpleWeightedGraphs = "47aef6b3-ad0c-573a-a1e2
SparseArrays = "2f01184e-e22b-5df5-ae63-d93ebab
Xtals = "ede5f01d-793e-4c47-9885-c447d1f18d6d"
```

✨ Interface-ify! ✨

File: **Project.toml**

```
name = "ChemistryFeaturization"
uuid = "6c925690-434a-421d-aea7-51398c5
authors = ["Rachel Kurchin <rkurchin@cm
version = "0.7.0"

[deps]
AtomsBase = "a963bdd2-2df7-4f54-a1ee-49
CSV = "336ed68f-0bac-5ca0-87d4-7b16caf5
DataFrames = "a93c6f00-e57d-5684-b7b6-d
Flux = "587475ba-b771-5e3f-ad9e-33799f1
JSON3 = "0f8b85d8-7281-11e9-16c2-39a750
LinearAlgebra = "37e2e46d-f89d-539d-b4e
Serialization = "9e88b42a-f829-5b0c-bbe
SparseArrays = "2f01184e-e22b-5df5-ae63
```

Feb '22

today!

ChemistryFeaturization today!

- Mostly a lightweight interface with a few concrete functionalities exported
- Supports featurization of AtomsBase systems
- Separation of concerns between feature descriptors, codecs, featurization schemes, featurized structures

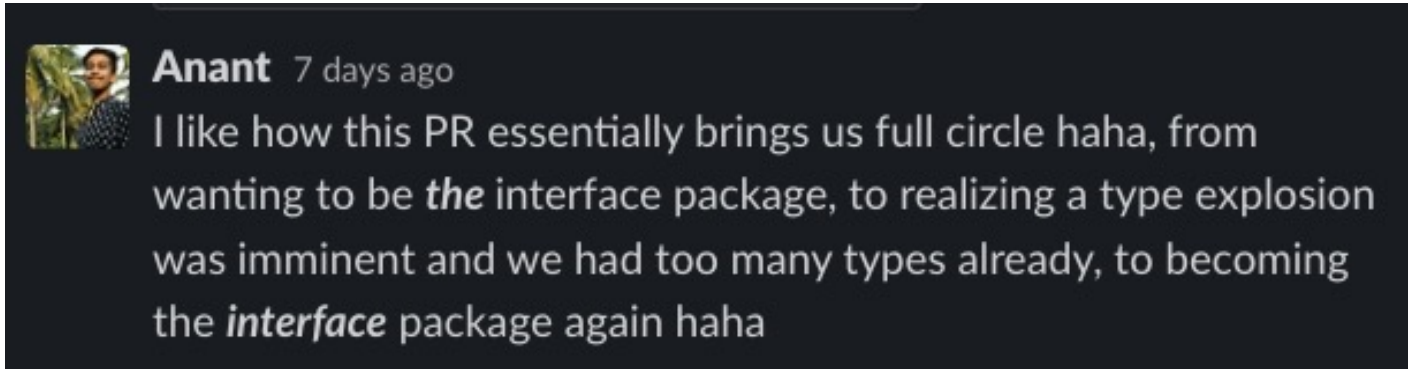
What's next?

Hopefully more active development again soon! 🤞
(We always welcome contributions!)

today!

Lessons Learned (and/or reinforced)!

- Julia interfaces are awesome



- Be willing to think big, but unashamed to start small
- Sometimes, writing up the description of your problem to send to the Julia Slack is enough for you to figure it out (#rubberduckcoding) but when it's not, they'll still answer your question within minutes

Acknowledgements!

THE MOLECULAR SCIENCES SOFTWARE INSTITUTE

Google Summer of Code

@cormullion for the beautiful logo!