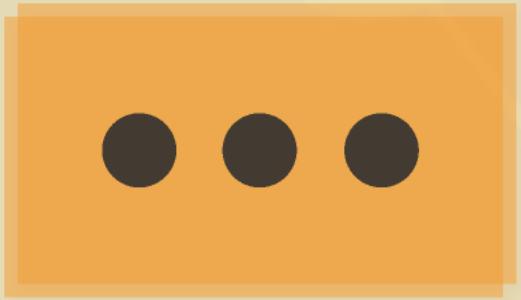
PairProphet

Humood Alanzi, Logan Roberts, Ryan Francis, Amin Mosallenejad, and Chau Vuong

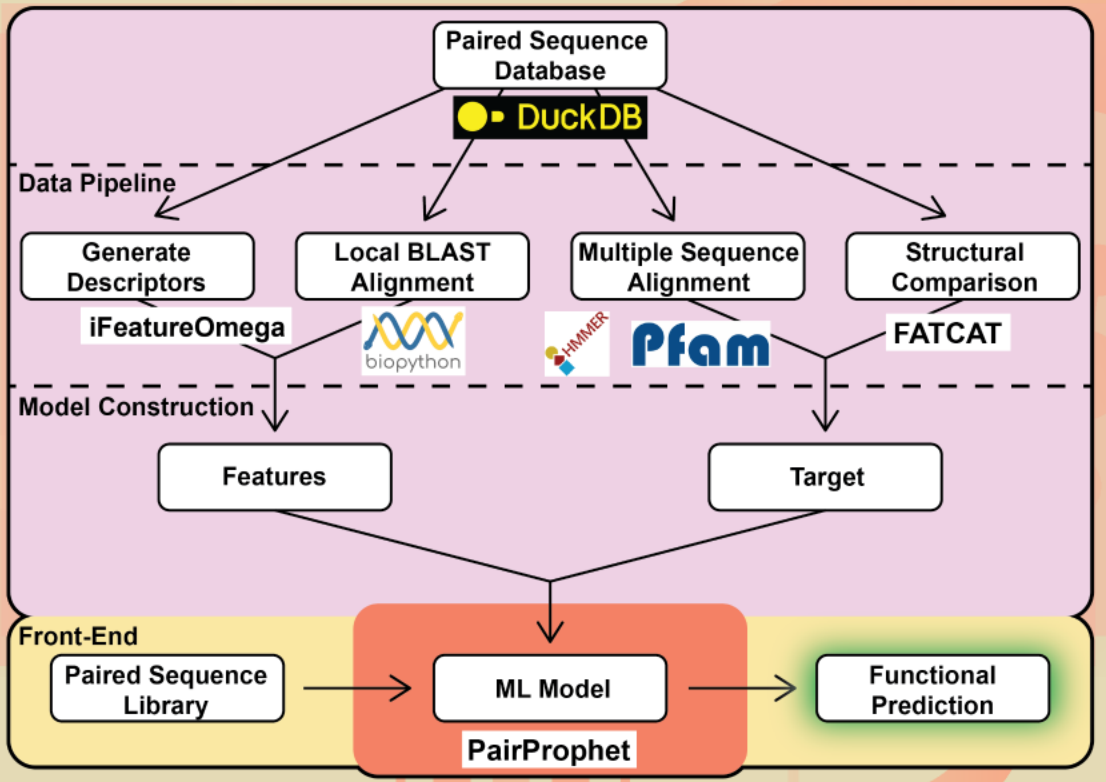
Background

Future Work



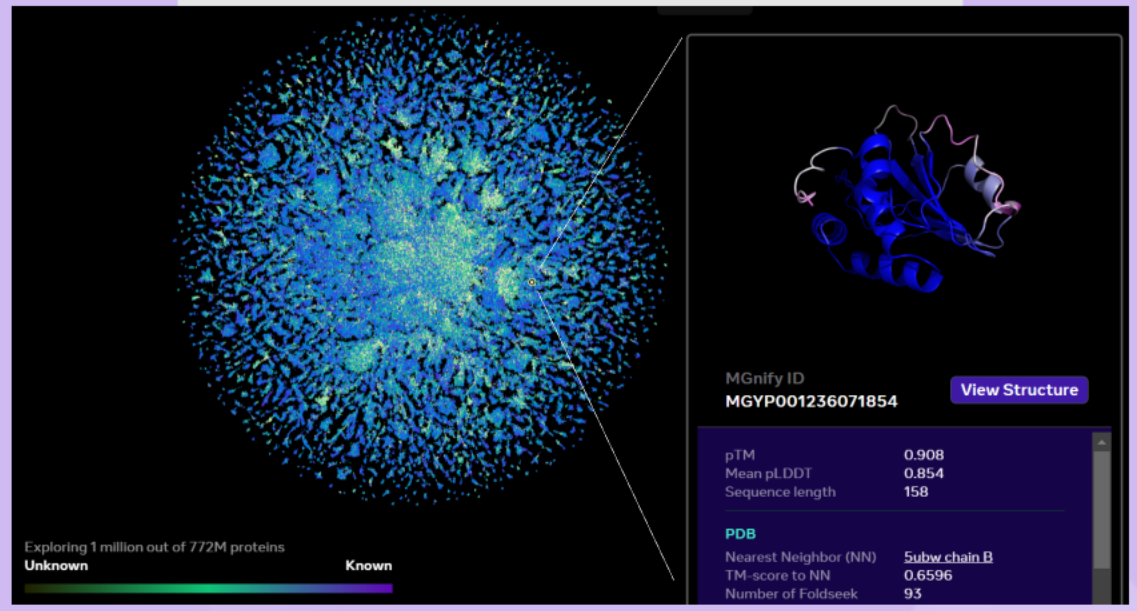
Acknowledgments

Pipeline

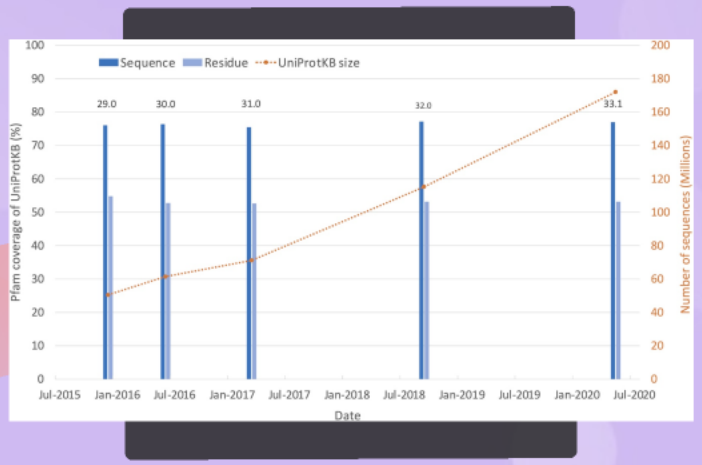


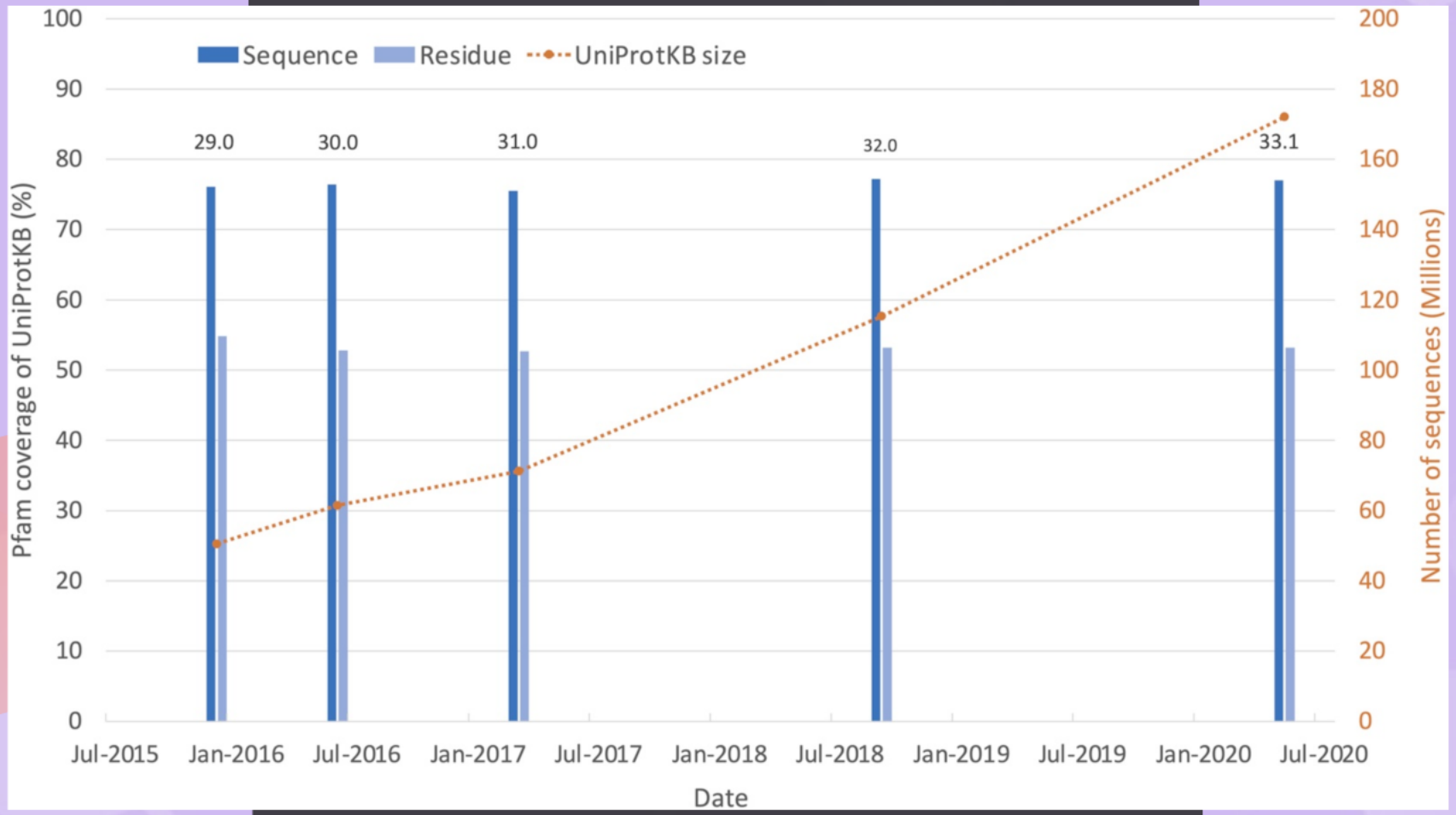
We are inundated with immense amounts of biological data

ESM Metagenomic Atlas with 772M proteins!



- The protein universe is massive
- Programs that filter through are various, inaccessible, or expensive to use.
- How can we reduce the search space for high-throughput screenings?



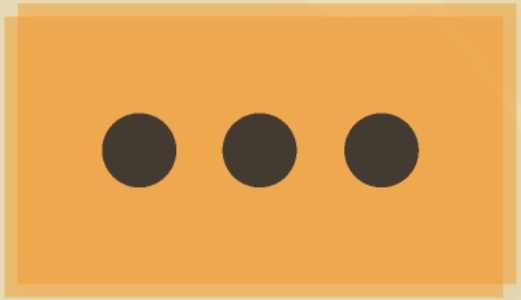

Sequence
Residue
UniProtKB size
29.0
30.0
31.0
32.0
33.1
Pfam coverage of UniProtKB (%)
Number of sequences (Millions)
Date
Jul-2015
Jan-2016
Jul-2016
Jan-2017
Jul-2017
Jan-2018
Jul-2018
Jan-2019
Jul-2019
Jan-2020
Jul-2020

PairProphet

Humood Alanzi, Logan Roberts, Ryan Francis, Amin Mosallenejad, and Chau Vuong

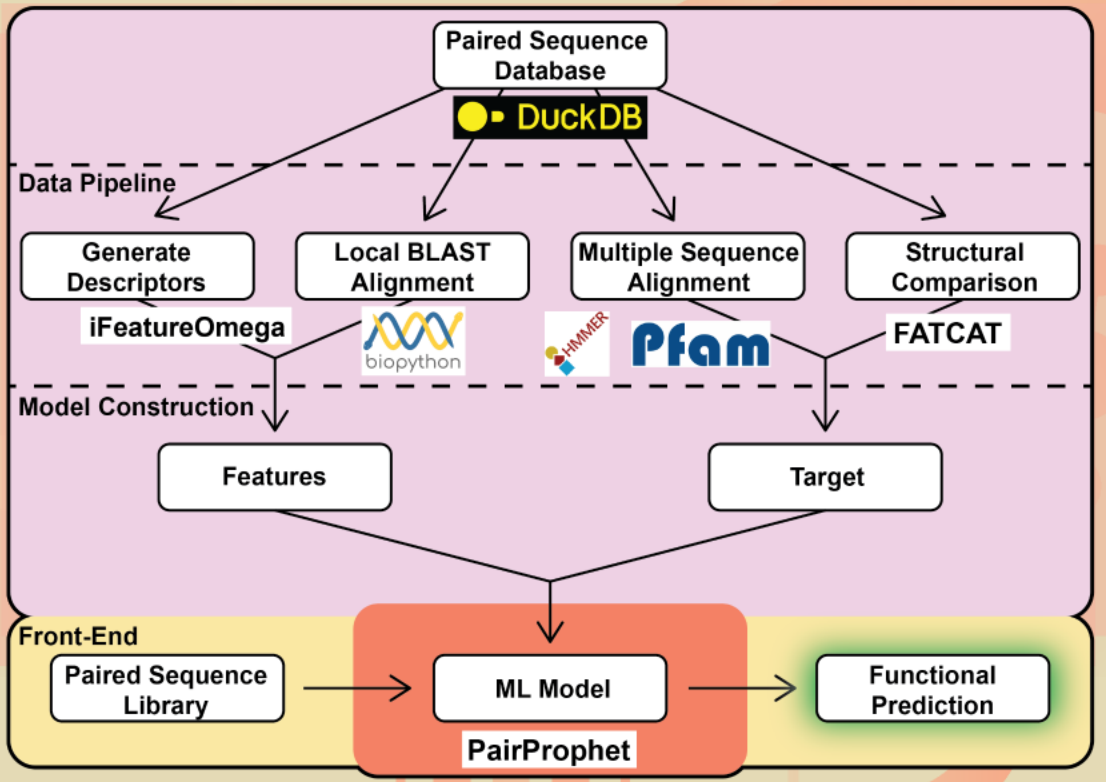
Background

Future Work



Acknowledgments

Pipeline

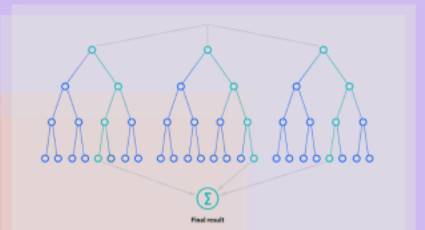


Pipeline

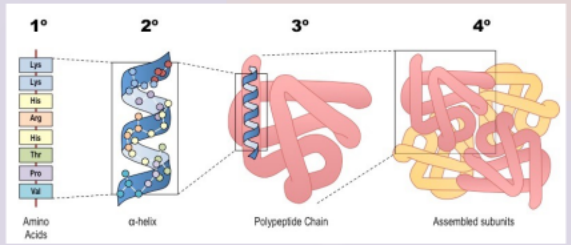
MSAs

HMMER

ML

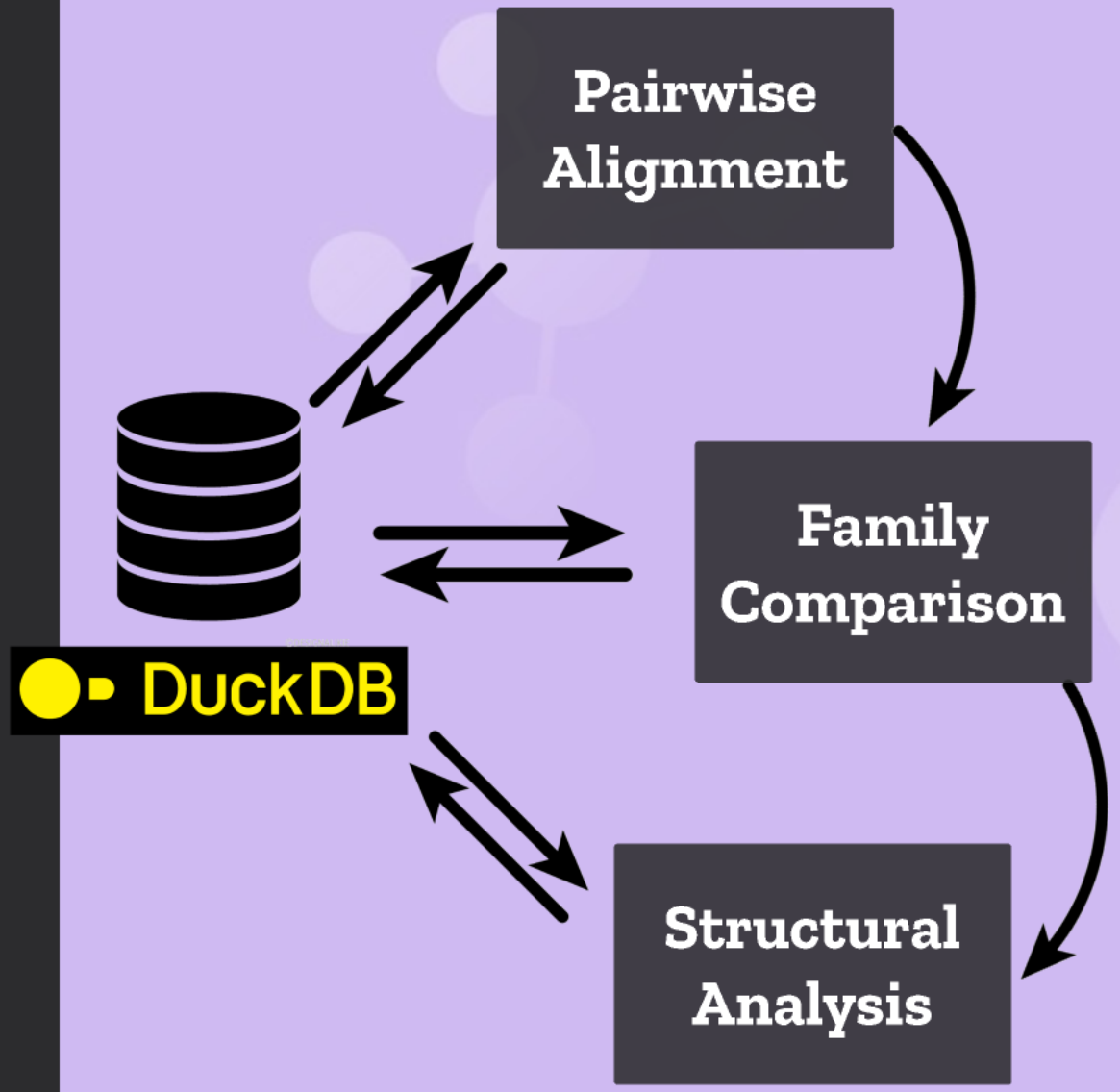


Structures



Data Management

DuckDB



Data Management

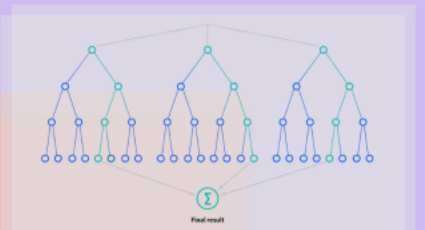
- Pairwise data is stored and updated in DuckDB database
- Modules like local BLAST call data, then add new columns to the growing table
- Finish with complete feature/target space

Pipeline

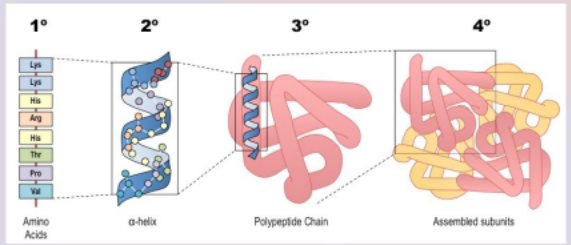
MSAs

HMMER

ML



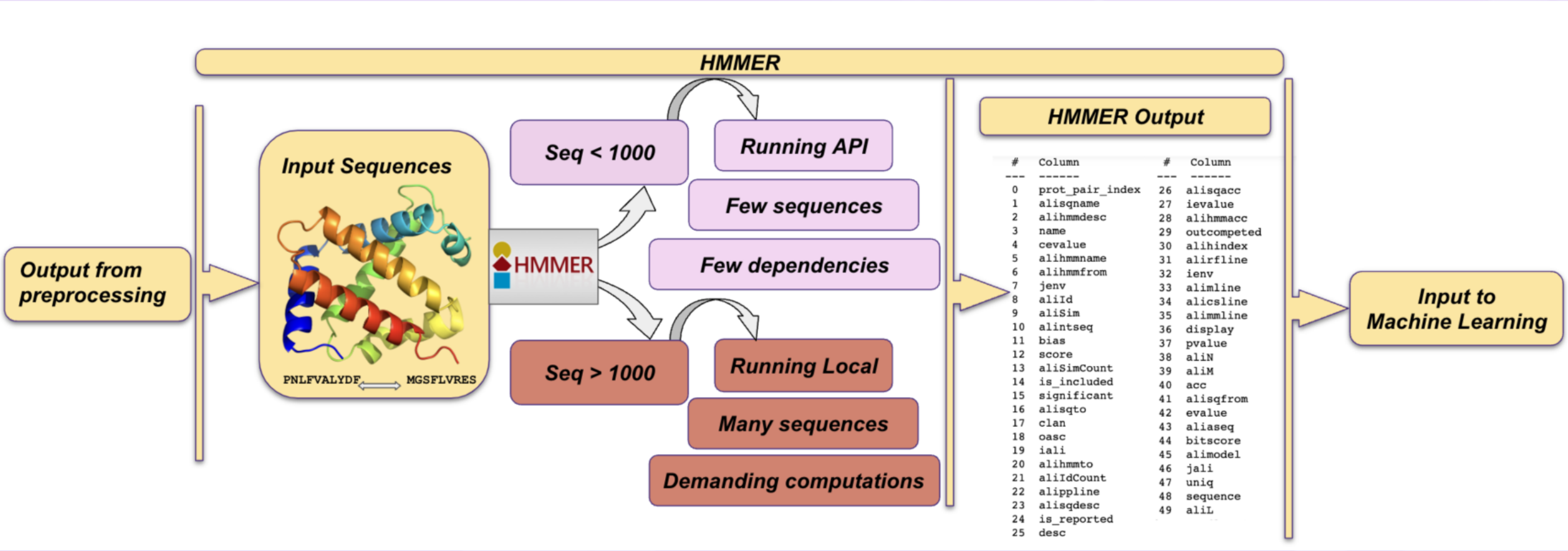
Structures



Data Management

DuckDB

Utilizing Multiple Sequence Alignments to infer protein family

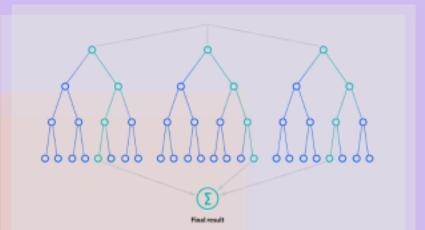


Pipeline

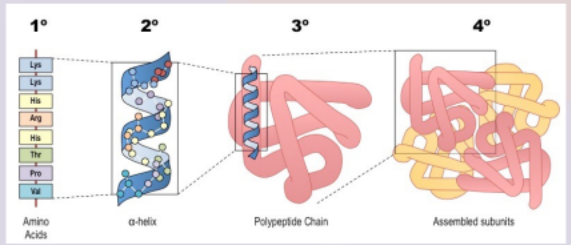
MSAs

HMMER

ML



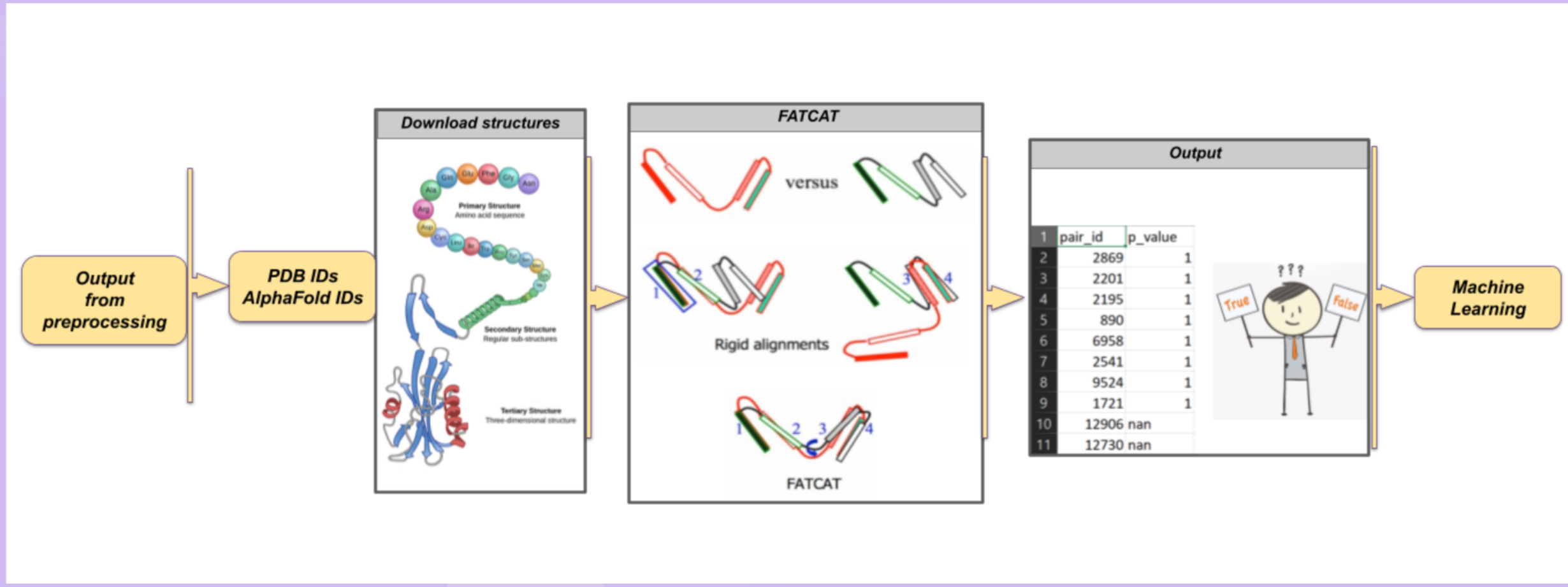
Structures



Data Management

DuckDB

Compare structures with FATCAT

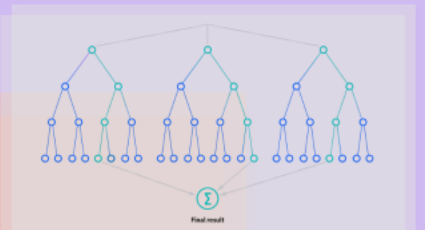


Pipeline

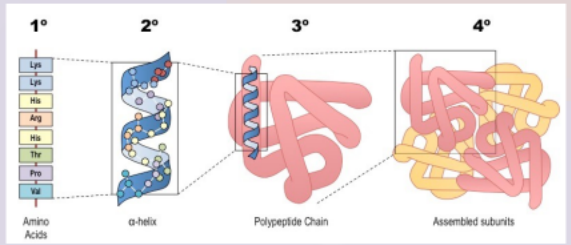
MSAs

HMMER

ML



Structures



Data Management

DuckDB

Protein Pair Functionality Prediction

Combine Software

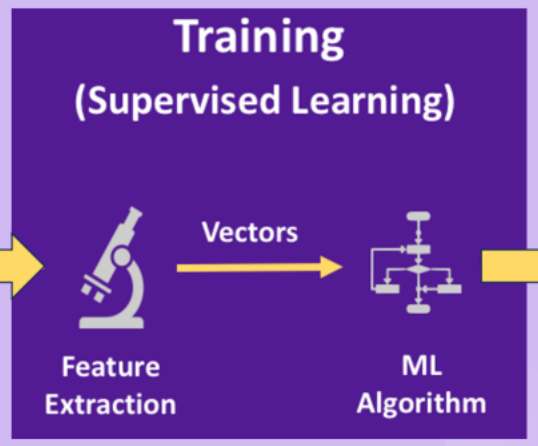
PyHMMER

FATCAT

biopython

Generate Features + Train Model

iFeatureOmega-CLI



Generate Performance Output

```
Evaluation Scale:
0.0% <=Accuracy<= 100.0%
0.0 <=AUC<= 1.0
0.0 <=auPR<= 1.0
0.0 <=F1_Score<= 1.0
-1.0 <=MCC<= 1.0

Accuracy: 81.7980%
Mean Precision: 0.8977
Mean F1_Score: 0.8895
Mean Recall: 0.8814
```

seq1	seq2	hmmer_match	structure_match	both_match
MAESG…	MPSQI…	TRUE	TRUE	TRUE
MTSGL…	MAMSR…	TRUE	FALSE	FALSE
MRAIG…	MATTT…	FALSE	TRUE	FALSE
MITVQ…	MSELK…	FALSE	FALSE	TRUE
MHELS…	MGLCE…	FALSE	TRUE	FALSE

Validation Steps

- Validate with different classifiers
- Select features from iFeatureOmega
- Optimize hyperparams with Optuna

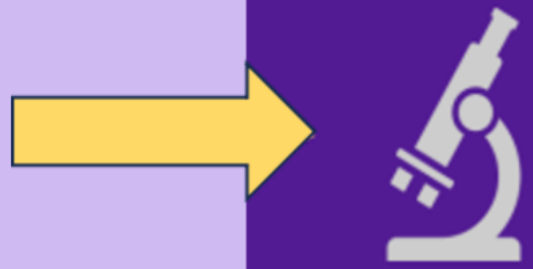
iFeatureOmega-CLI

PyHMMER

FATCAT

Training

(Supervised Learning)



Vectors

Feature Extraction

ML Algorithm

Protein Pair Functionality Prediction

Combine Software

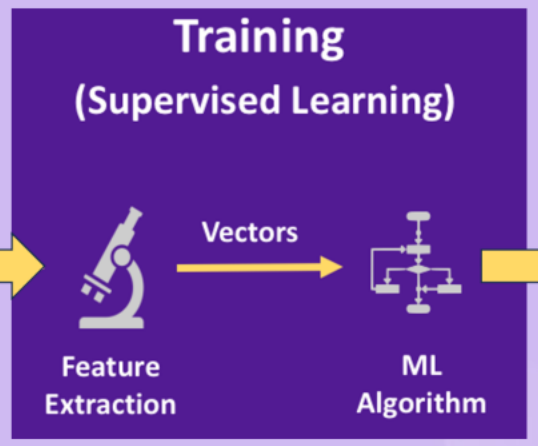
PyHMMER

FATCAT

biopython

Generate Features + Train Model

iFeatureOmega-CLI



Generate Performance Output

```
Evaluation Scale:
0.0% <=Accuracy<= 100.0%
0.0 <=AUC<= 1.0
0.0 <=auPR<= 1.0
0.0 <=F1_Score<= 1.0
-1.0 <=MCC<= 1.0

Accuracy: 81.7980%
Mean Precision: 0.8977
Mean F1_Score: 0.8895
Mean Recall: 0.8814
```

seq1	seq2	hmmer_match	structure_match	both_match
MAESG…	MPSQI…	TRUE	TRUE	TRUE
MTSGL…	MAMSR…	TRUE	FALSE	FALSE
MRAIG…	MATTT…	FALSE	TRUE	FALSE
MITVQ…	MSELK…	FALSE	FALSE	TRUE
MHELS…	MGLCE…	FALSE	TRUE	FALSE

Validation Steps

- Validate with different classifiers
- Select features from iFeatureOmega
- Optimize hyperparams with Optuna

ures +

Generate Performance Output

Validation Step

ega-CLI

g
earning)

ML
Algorithm

```
Evaluation Scale:
0.0% <=Accuracy<= 100.0%
0.0 <=AUC<= 1.0
0.0 <=auPR<= 1.0
0.0 <=F1_Score<= 1.0
-1.0 <=MCC<= 1.0

Accuracy: 81.7980%
Mean Precision: 0.8977
Mean F1_Score: 0.8895
Mean Recall: 0.8814
```

seq1	seq2	hmmer_match	structure_match	both_match
MAESG…	MPSQI…	TRUE	TRUE	TRUE
MTSGL…	MAMSR…	TRUE	FALSE	FALSE
MRAIG…	MATTT…	FALSE	TRUE	FALSE
MITVQ…	MSELK…	FALSE	FALSE	TRUE
MHELS…	MGLCE…	FALSE	TRUE	FALSE

- Validate with di
classifiers

- Select features f
iFeatureOmega

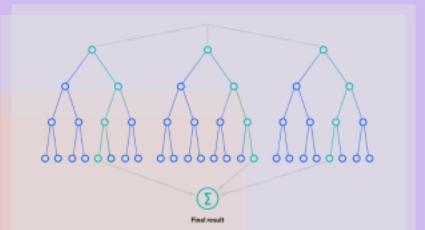
- Optimize hyper
with Optuna

Pipeline

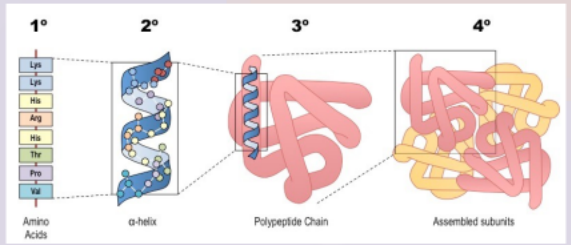
MSAs

HMMER

ML



Structures



Data Management

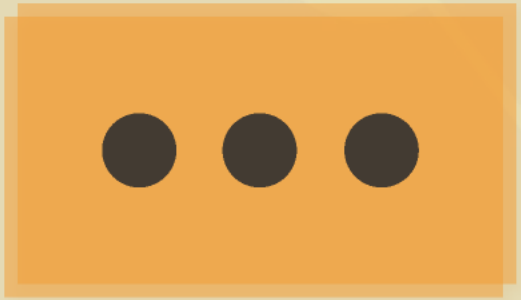
DuckDB

PairProphet

Humood Alanzi, Logan Roberts, Ryan Francis, Amin Mosallenejad, and Chau Vuong

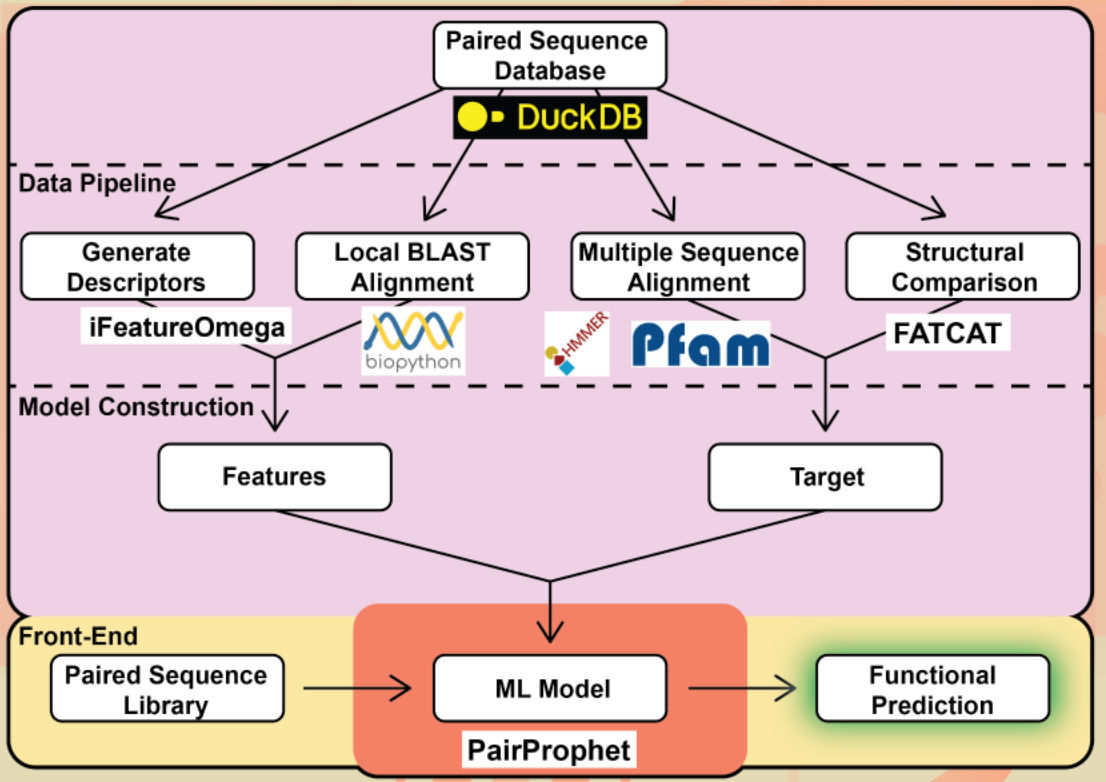
Background

Future Work



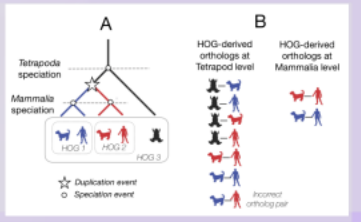
Acknowledgments

Pipeline



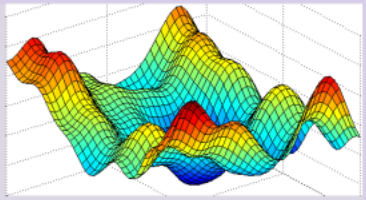
Modularity

New Database



Version Contol

ML Architecture



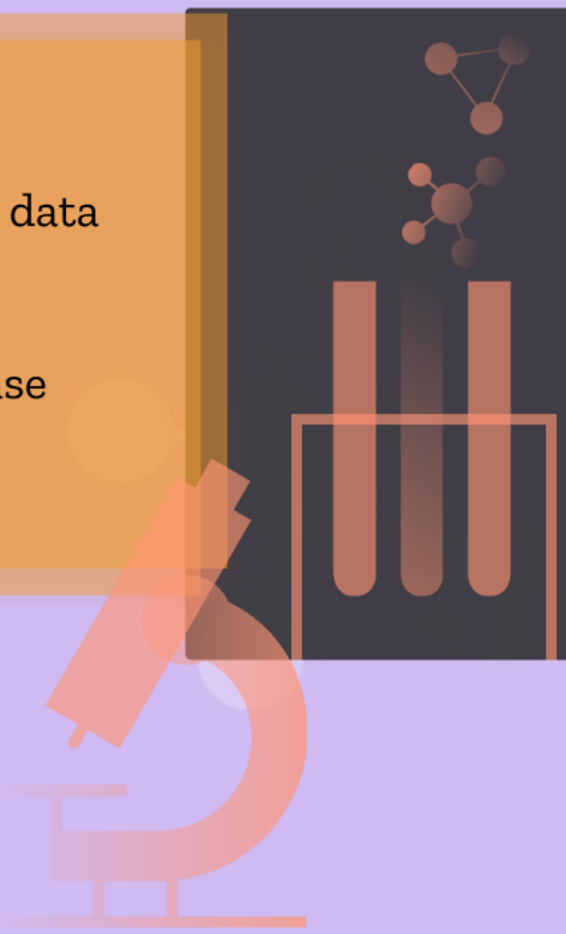
Future Work

1. Improve modularity of pipeline

2. Use version control to improve data reproducibility

3. Train model with better database

4. Expand model architecture



Pipeline Modularity

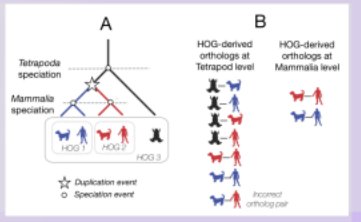
Adopt an object-oriented approach to the code to allow the user more configuration options.

Create a python wrapper for FATCAT to allow the user to run the structural alignment from the command line to be more accessible.

Create a REST API to allow the user to run the model from a web interface.

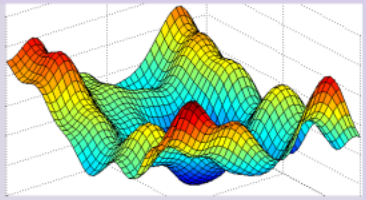
Modularity

New Database



Version Contol

ML Architecture



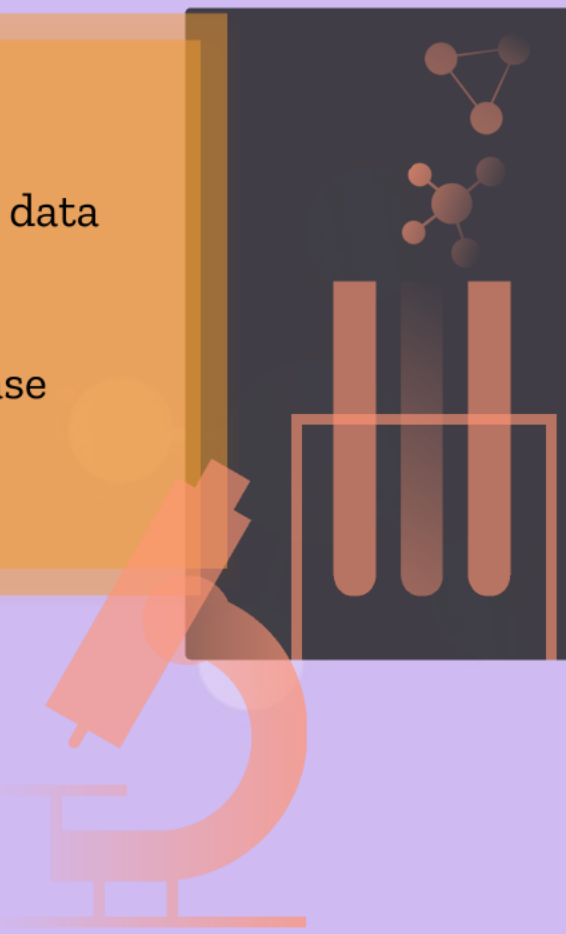
Future Work

1. Improve modularity of pipeline

2. Use version control to improve data reproducibility

3. Train model with better database

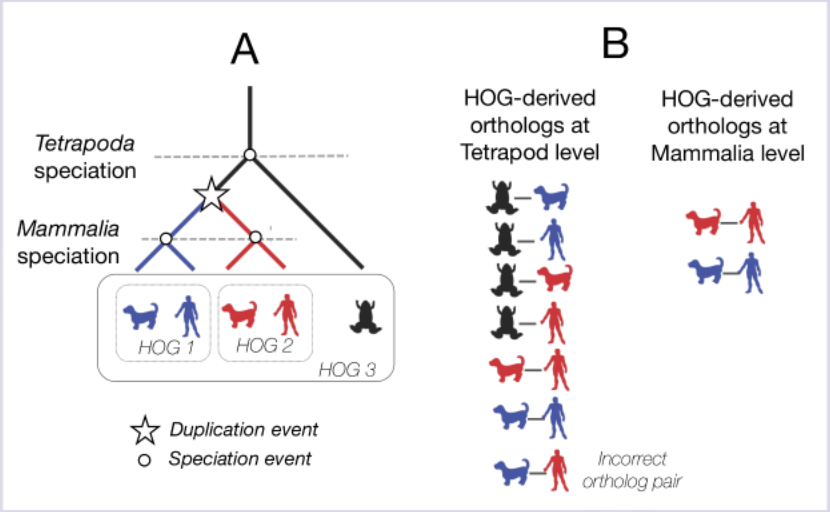
4. Expand model architecture



Upgrade Database

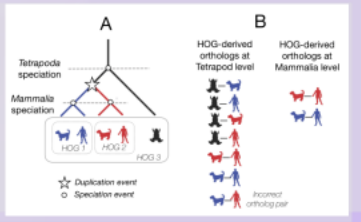
-Use the OMA table of orthologs as the main input for the model instead of learn2thermDB.

-Explore other large protein pair databases to train model



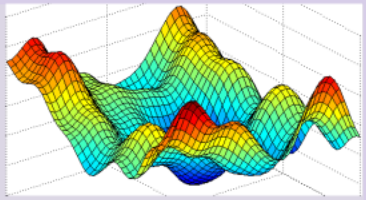
Modularity

New Database



Version Contol

ML Architecture



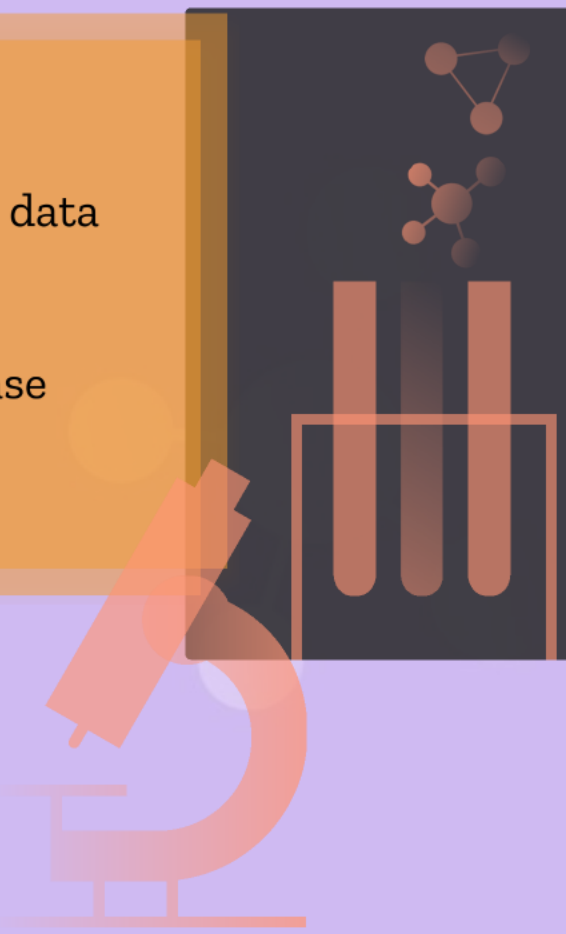
Future Work

1. Improve modularity of pipeline

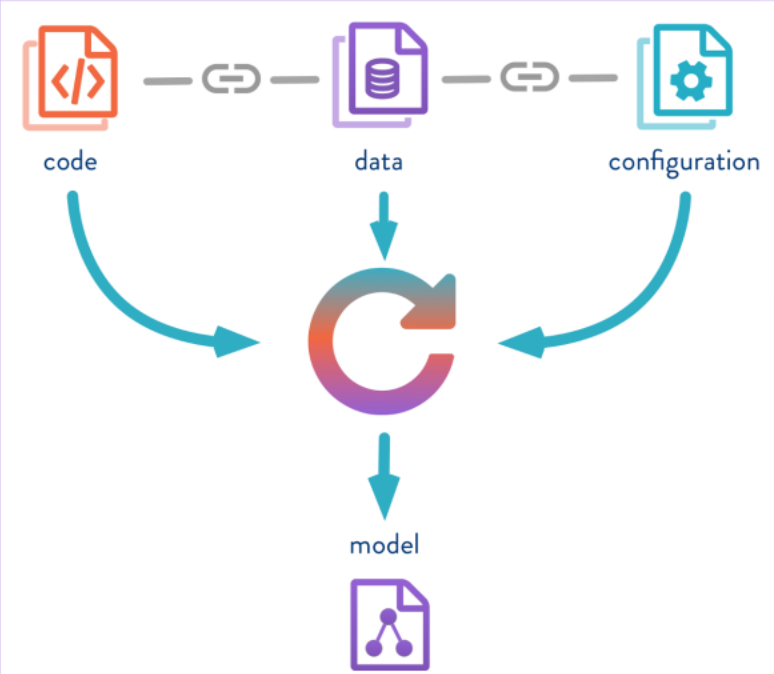
2. Use version control to improve data reproducibility

3. Train model with better database

4. Expand model architecture



Version-Controlled Datasets



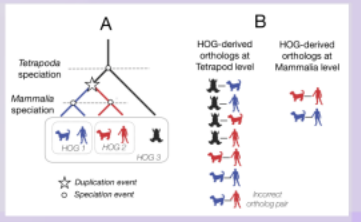
- Imlpemenet data version control to allow the user to track the changes in the data and improve model reproducibility.

- Integrate runclassifiers script with new datasets to optimize model performance

- Explore new ways to utilize generated features

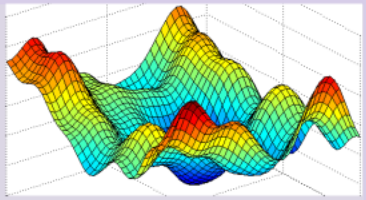
Modularity

New Database



Version Contol

ML Architecture



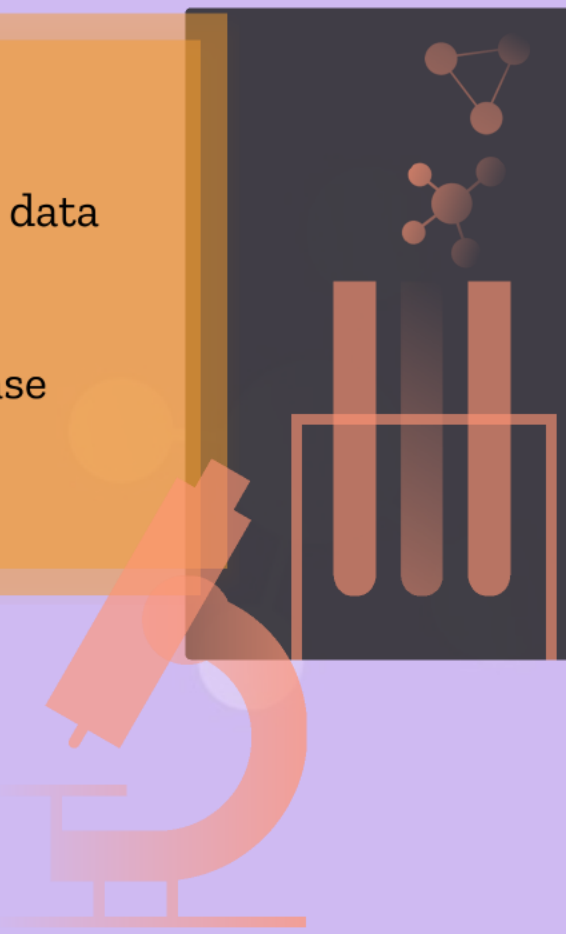
Future Work

1. Improve modularity of pipeline

2. Use version control to improve data reproducibility

3. Train model with better database

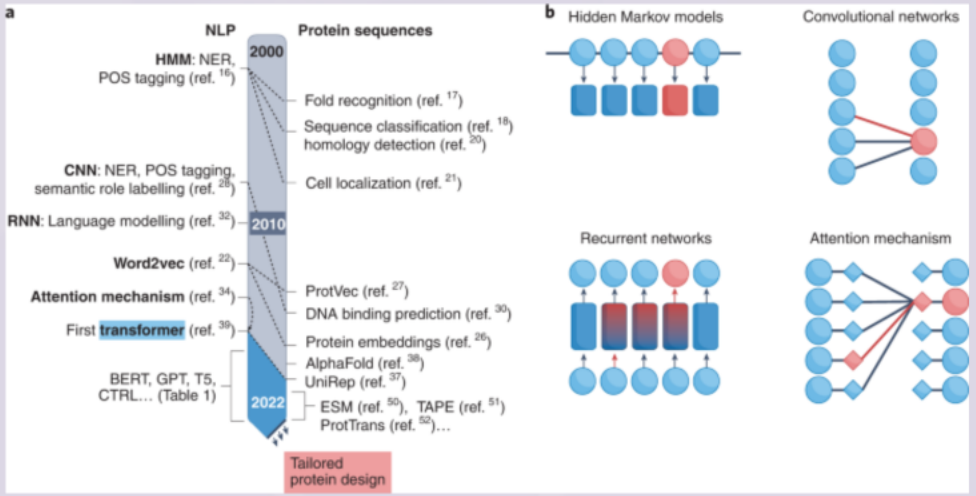
4. Expand model architecture



Explore new ML Architecture

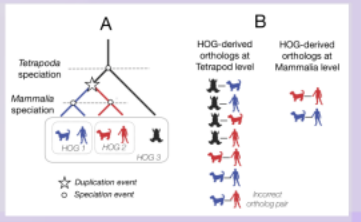
- Incorporate Gaussian processes so user can understand uncertainty

- Explore use of Neural Networks/Embedded Language Models for expanding feature space



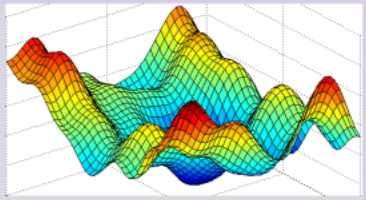
Modularity

New Database



Version Contol

ML Architecture



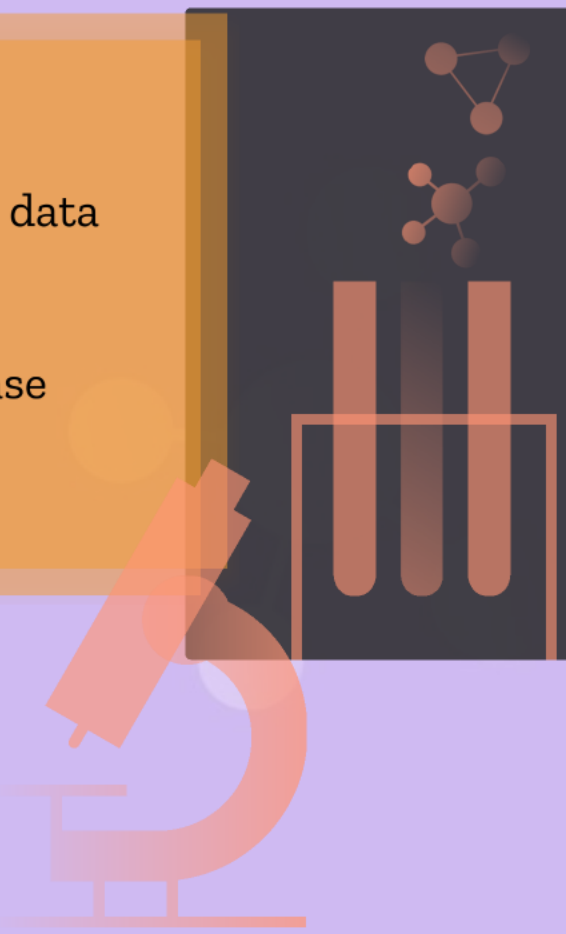
Future Work

1. Improve modularity of pipeline

2. Use version control to improve data reproducibility

3. Train model with better database

4. Expand model architecture

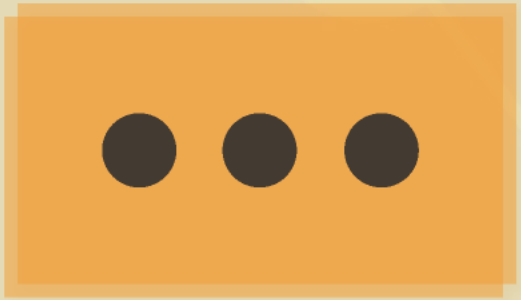


PairProphet

Humood Alanzi, Logan Roberts, Ryan Francis, Amin Mosallenejad, and Chau Vuong

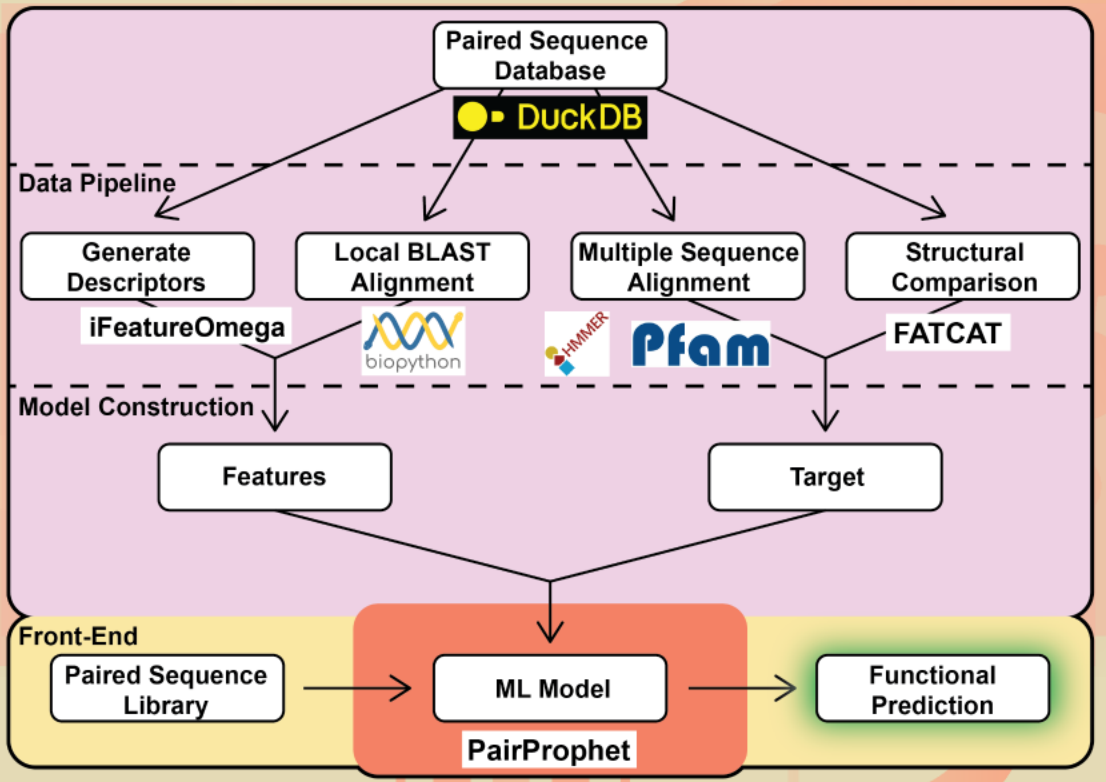
Background

Future Work



Acknowledgments

Pipeline



We would like to acknowledge **Prof. David Beck** and **Evan Komp** for sponsoring and mentoring our project.

References

Please visit our documentation website for a full list of references:

https://github.com/learn2therm/PairProphet

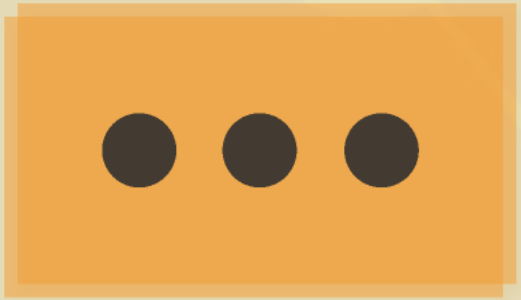
https://pairprophet.readthedocs.io

PairProphet

Humood Alanzi, Logan Roberts, Ryan Francis, Amin Mosallenejad, and Chau Vuong

Background

Future Work



Acknowledgments

Pipeline