A Scalable Framework for Synonymy Prediction: Inserting New Biomedical Terms into the UMLS

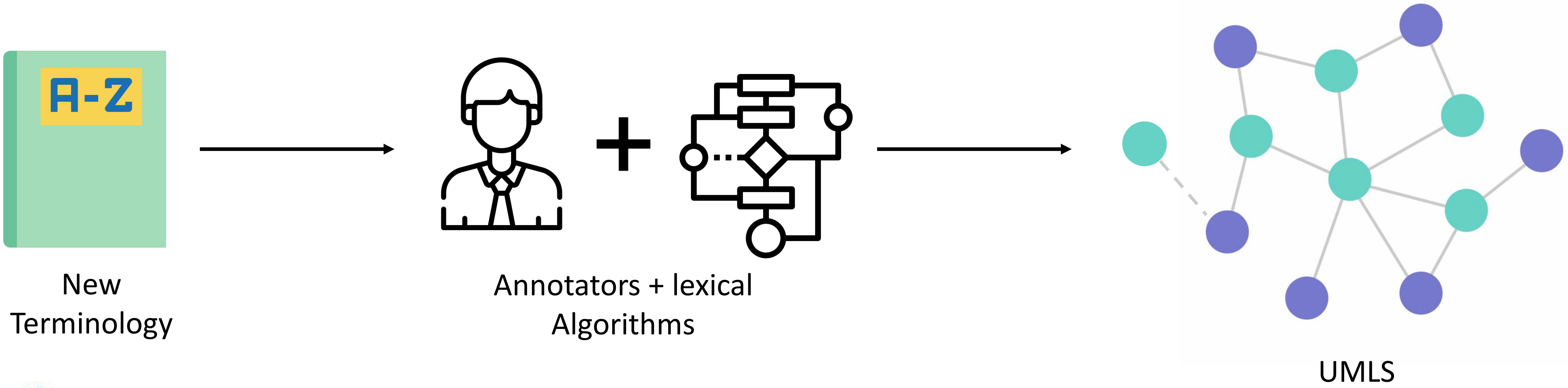
Vishesh Javangula

Dr. Olivier Bodenreider
Dr. Vinh Nguyen

National Library of Medicine
Lister Hill National Center for Biomedical Communications

Motivation

- UMLS is a biomedical terminology integration system that integrates over 200 biomedical vocabularies
- Construction Process through human annotators and lexical algorithms
  - **Problem:** Time-consuming and Error Prone
  - **Challenge:** Reduce running time and errors



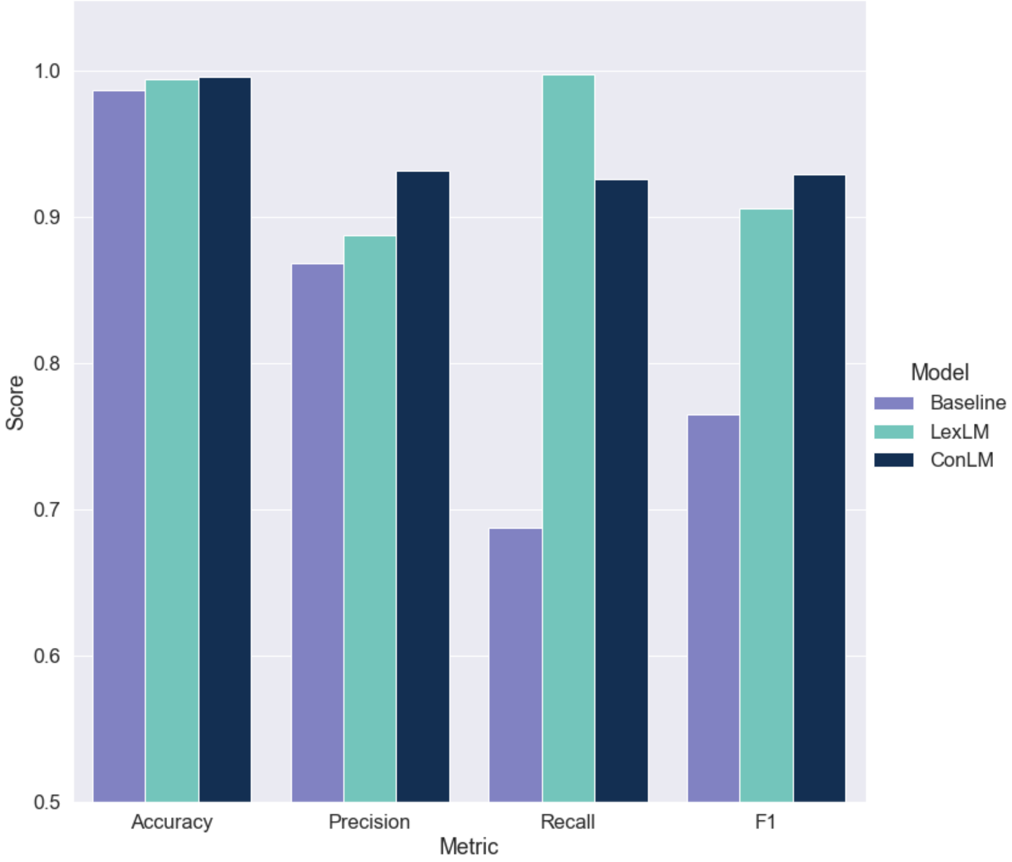
New Terminology

Annotators + lexical Algorithms

UMLS

Prior Work

- Dataset
  - UMLS Version 2020AA
  - 1.73×10^8 pairs
  - Negative pairs had varying degrees of lexical similarity
  - Maximized coverage of AUIs in training + test dataset
- Limitations:
  - Models have not been applied to real use case



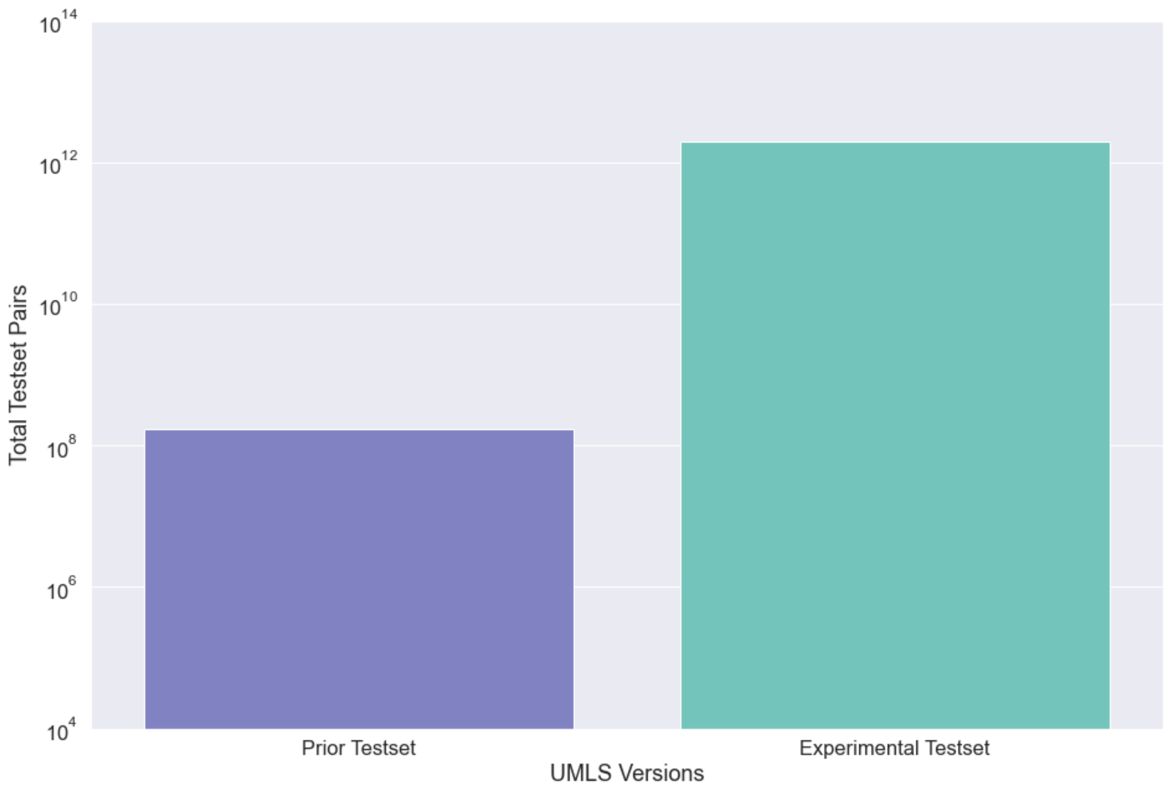
Model metrics across baseline, LexLM (model trained using lexical information), and ConLM (model trained with lexical and contextual information)

Goals

- Create a framework to apply prior models on a real use case: inserting new biomedical terms into an existing version of the UMLS Metathesaurus.
- Minimize running time when **predicting synonymy** to improve feasibility of model adoption into the UMLS construction process
    - Months and years → days and weeks

Challenges and Contribution

- Challenge:
  - Scalability
    - Experiment size: 2×10^{12}
    - Prior Dataset size: 1.73×10^{8}
- Contribution:
  - Approach to minimize running time required for synonym prediction



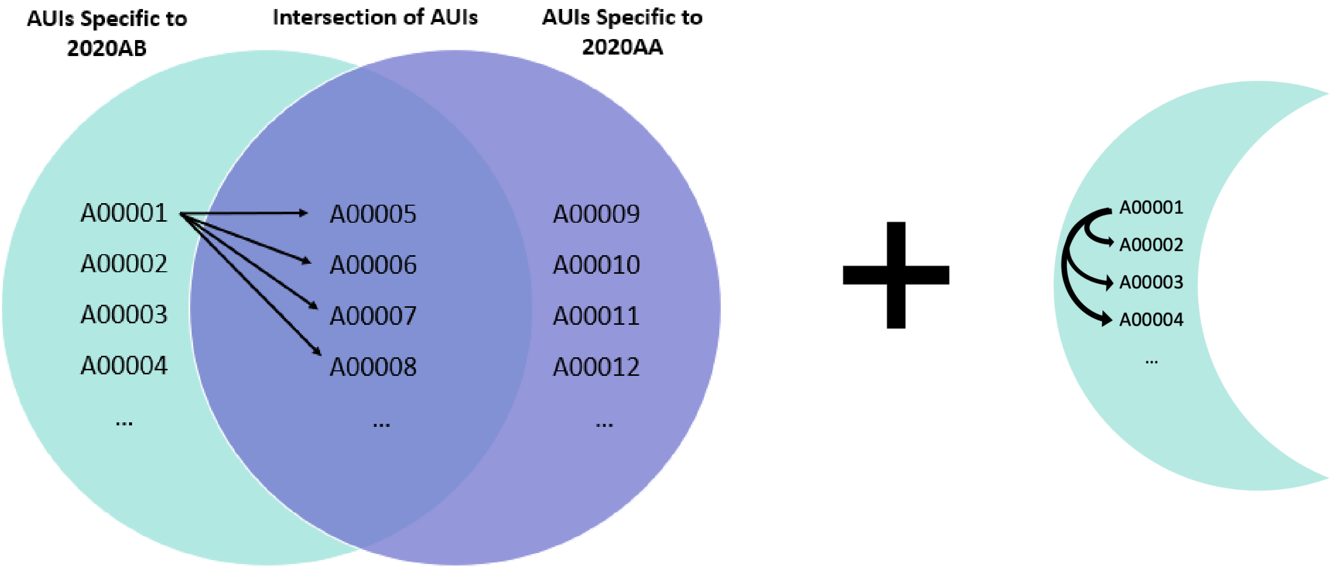
NIH National Library of Medicine
Lister Hill National Center for Biomedical Communications

Dataset

Purpose: Insert new AUIs (2020AB) into prior version of unsuppressed AUIs (Intersection of AUIs)

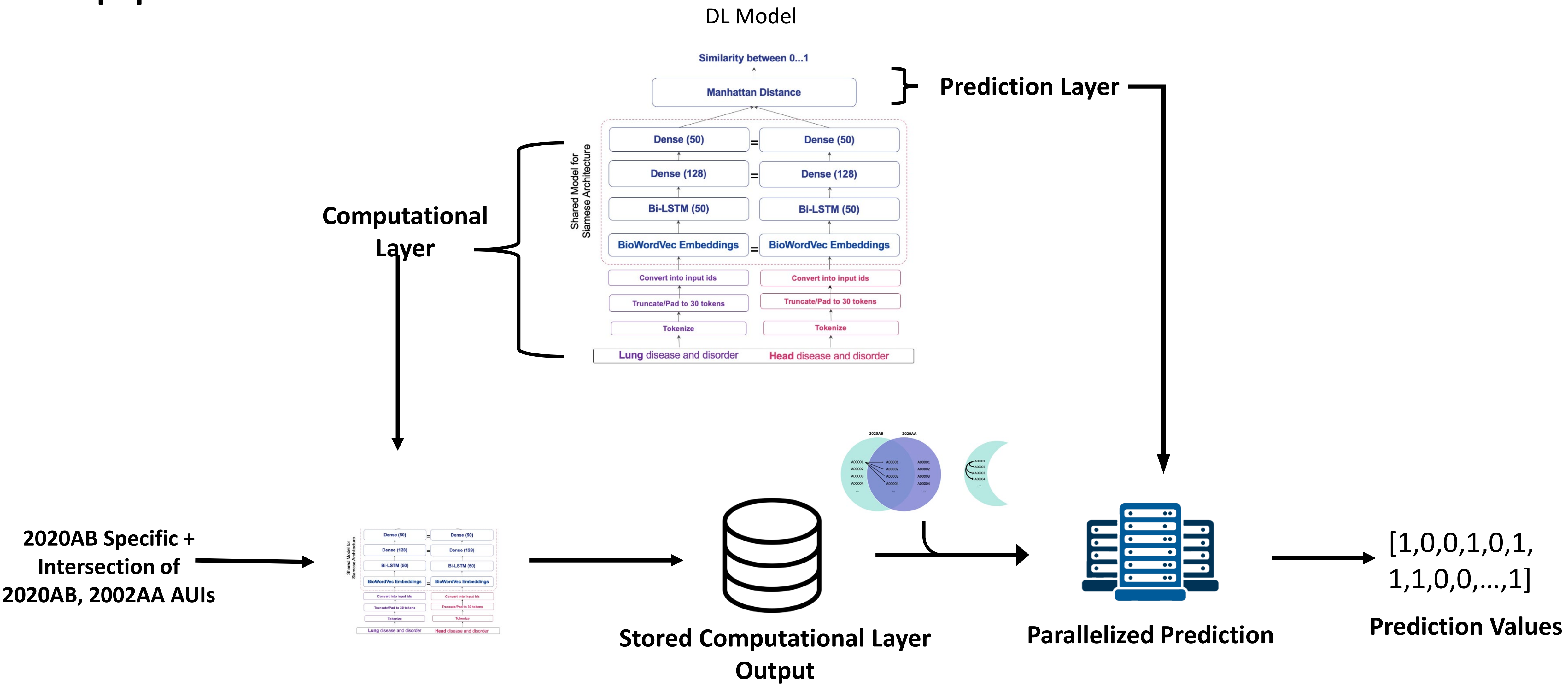
Strategy: Cross product of AUIs specific to 2020AB and Intersection of AUIs + All pairs within AUIs specific to 2020AB

Result: 2×10^{12} pairs



Dataset pairing method using AUIS specific to 2020AB and AUIs present in both 2020AB and 2020AA

Approach



DL Model

Prediction Layer

Computational Layer

2020AB Specific + Intersection of 2020AB, 2002AA AUIs

Stored Computational Layer Output

Parallelized Prediction

[1,0,0,1,0,1,
1,1,0,0,...,1]

Prediction Values

Project Progress

- Completed
  - Testset Generation
  - Precomputing AUIs using computational layer
- In Progress
  - Parallelized Predictions
- To Do:
  - Error Analysis

Conclusion

- Motivation
  - Address the laborious and error-prone UMLS construction process
- Prior Work
  - Used deep learning for synonymy prediction of AUIs
  - Greatly exceeded baseline
- Goals
  - Create framework to apply models to real use case
  - Minimize running time of synonymy prediction
- Challenge and Contribution
  - Scalability
  - Approach to minimize running time on dataset
- Approach
  - Model splitting and precomputing computational layer prior to parallelizing predictions

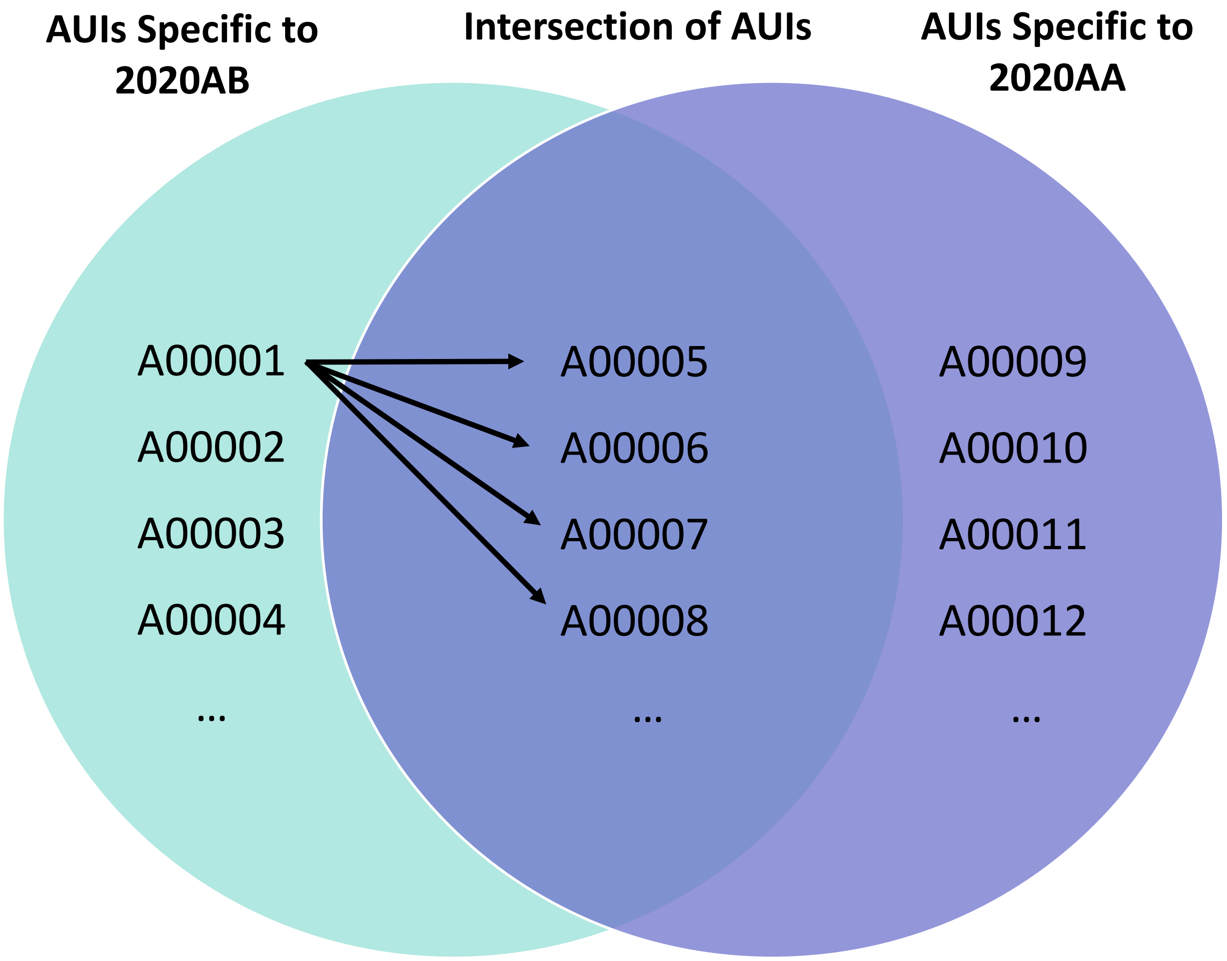
Acknowledgements

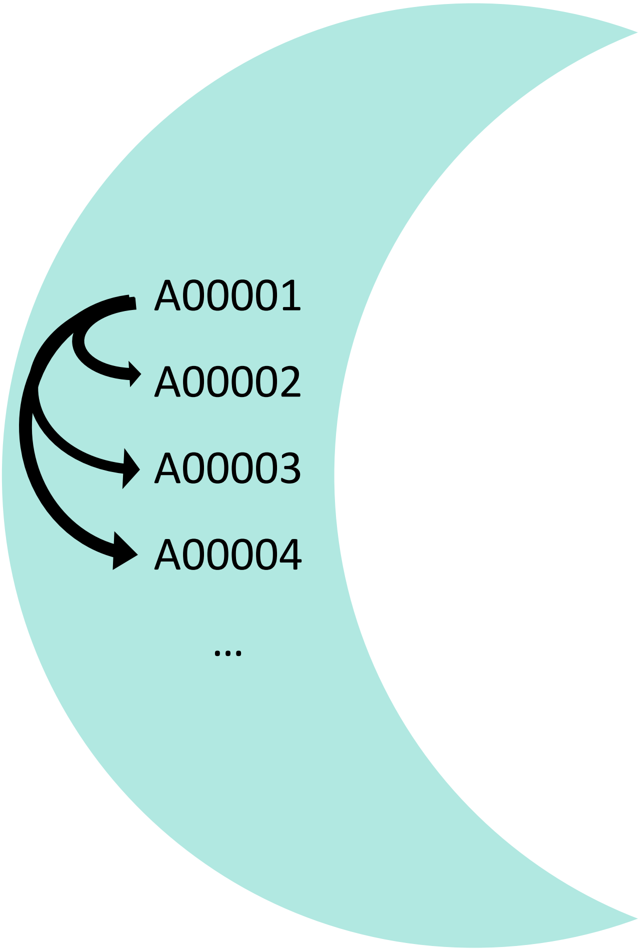
Supported by Graduate Data Science Summer Program (GDSSP) hosted by NIH OITE

Thank you NLM, Dr. Olivier Bodenreider, and Dr. Vinh Nguyen for their mentorship and support

Thank you to Goonmeet and Thilini

Thank you


AUIs Specific to 2020AB
Intersection of AUIs
AUIs Specific to 2020AA
A00001
A00002
A00003
A00004
…
A00005
A00006
A00007
A00008
…
A00009
A00010
A00011
A00012
…


A00001
A00002
A00003
A00004
…

Poster Layout

- Introduction
  - What is the UMLS
  - What is the problem
  - Our solution using DL
  - Our prior experiments leveraged deep learning models to predict pairs of terms as synonymous using lexical and contextual information (eg. semantic group, source synonymy).
  - While these experiments suggest a significant improvement, this new approach remains to be tested on a real use case, e.g., inserting new biomedical terms into an existing version of the UMLS Metathesaurus.
- Goals
  - we evaluate the performance of these deep learning models in predicting the synonymy between terms present in UMLS version 2020AA and 2020AB
  - improve feasibility of model adoption into the UMLS construction process,
- Challenge
  - Scalability is the primary challenge in these experiments as it would take approximately 5 months to evaluate our new testset and 17 years to predict every pair in the UMLS given a single GPU.
  - Parallelizing ith additional GPUS canl linearly reduce running time, but it remains unacceptable for our use case
- Hypothesis

- Methods
  - Dataset
  - Scalability

- Results
  - TBD
- Conclusion
  - TBD
- Future Works
- Acknowledgements