Storing ICD-10 Codes in Graphs

Team 1 -

Daxton Furniss: Final model creation, (Neo4j)

Nathan Losee: Team coordination, Data Acquisition, Conceptual model

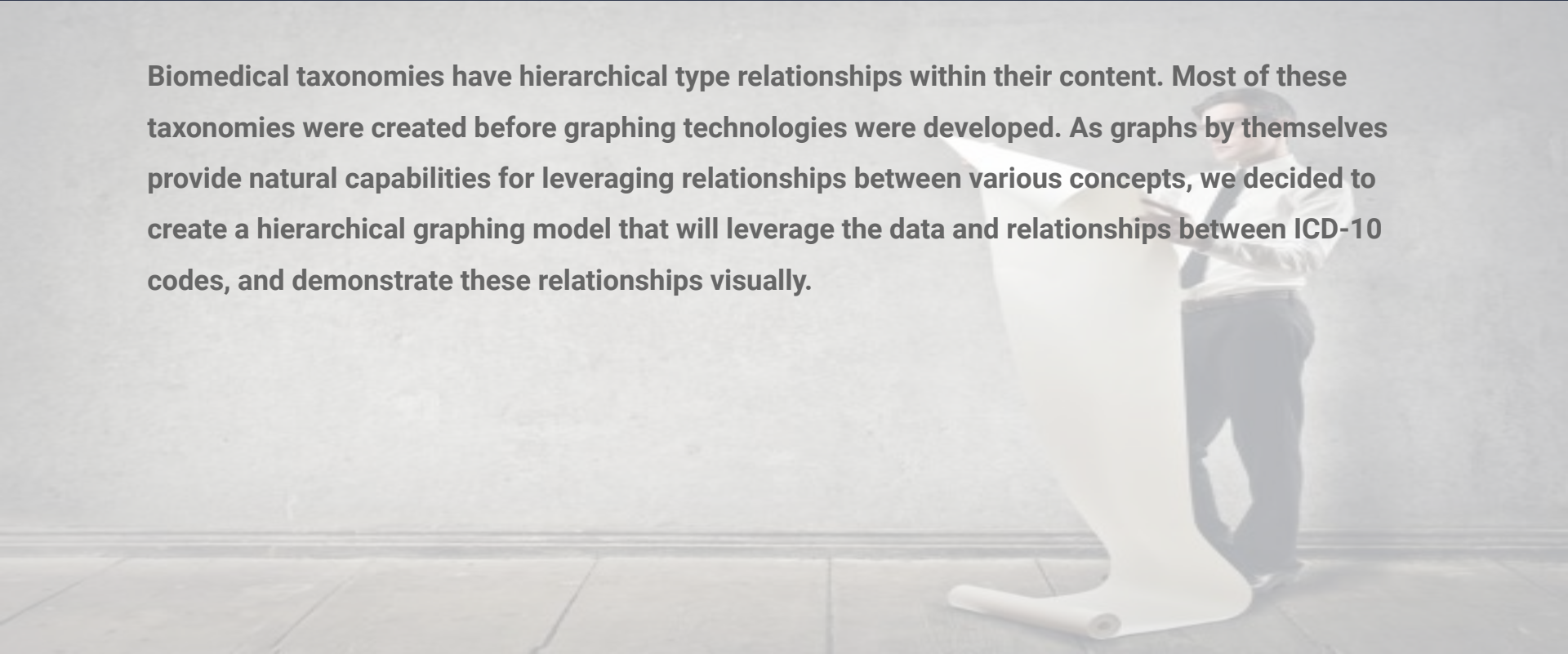
Sakshi Singh: Background/Historical Context

Scott Wardle: Data Wrangling, Final Model Assistance

Tanner Frahm: Data Wrangling, Final Model Assistance

Background

Biomedical taxonomies have hierarchical type relationships within their content. Most of these taxonomies were created before graphing technologies were developed. As graphs by themselves provide natural capabilities for leveraging relationships between various concepts, we decided to create a hierarchical graphing model that will leverage the data and relationships between ICD-10 codes, and demonstrate these relationships visually.

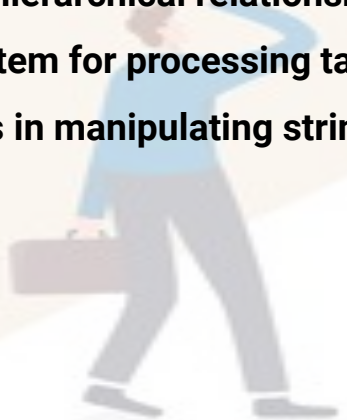


Motivation

- Interest in enhancing comprehension and utility of biological and clinical taxonomies.
- Motivated by experiences with taxonomies like ICD and other biological and clinical EMR codes.
- Aim to develop methodologies for preparing taxonomies for graphical representation.
- Desire to improve understanding of biological systems through graphical representations.
- Utilizes advanced graphing technologies like Neo4j.

Purpose

- **Transform taxonomy data into a format compatible with graphing technologies.**
- **Master preparation of datasets for graphical representation.**
- **Demonstrate hierarchical relationships.**
- **Develop a system for processing taxonomy data for visual representation.**
- **Enhance skills in manipulating string and character data for analysis.**



Data Acquisition

- **Began with SNOMED CT codes, however since SNOMED CT Concepts are organized into a complex ontology with relationships among them (, parent-child, part-of, causative-agent) rather than ICD-10's hierarchical structure with chapters, blocks, and subcategories, we changed gears and focused on ICD-10.**

- **ICD-10 Codes - International Classification of Diseases - Defined by World Health Organization, https://www.cdc.gov/nchs/icd/Comprehensive-Listing-of-ICD-10-CM-Files.htm**

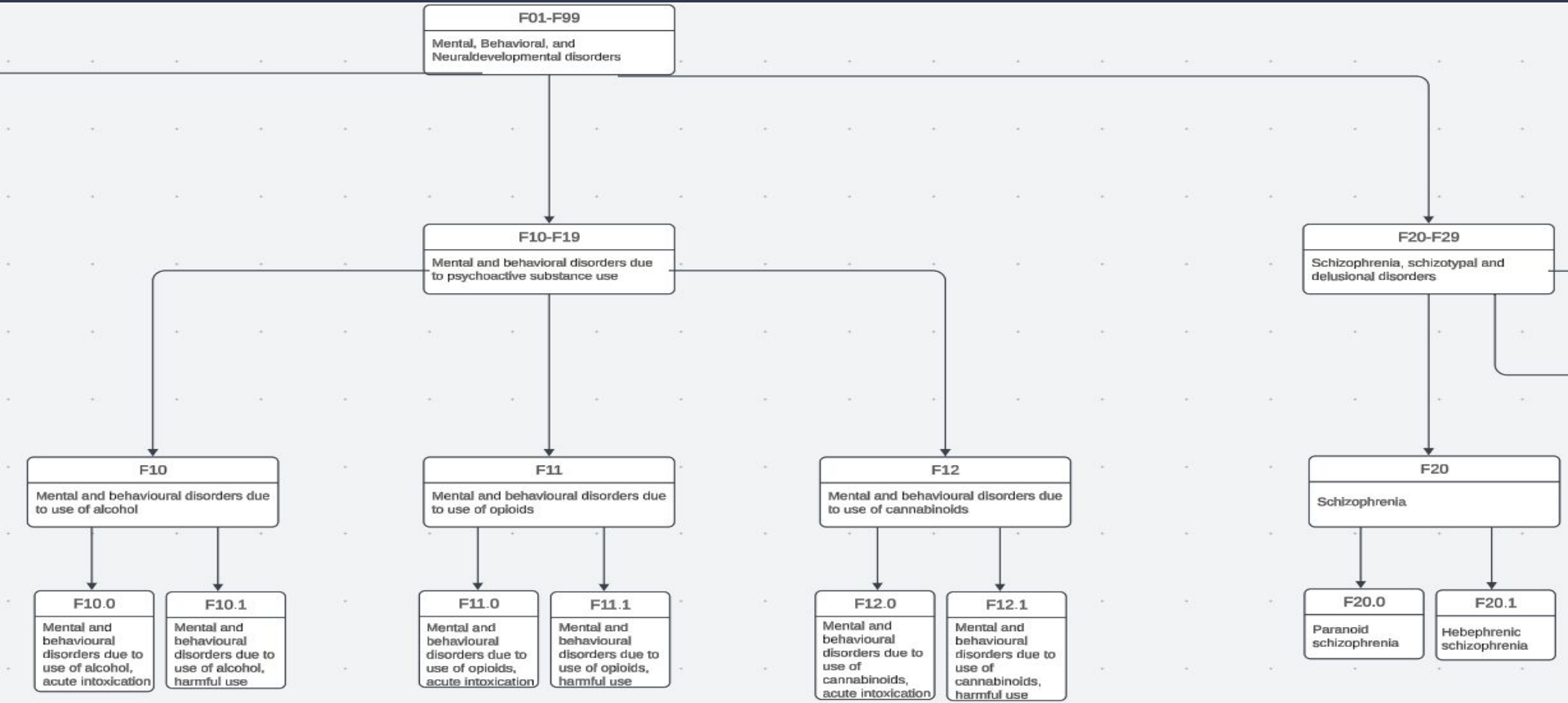
- **Most complete/comprehensive list of ICD-10 codes and information found on: https://www.health.gov.za/icd-10-master-industry-table/**

Data Wrangling

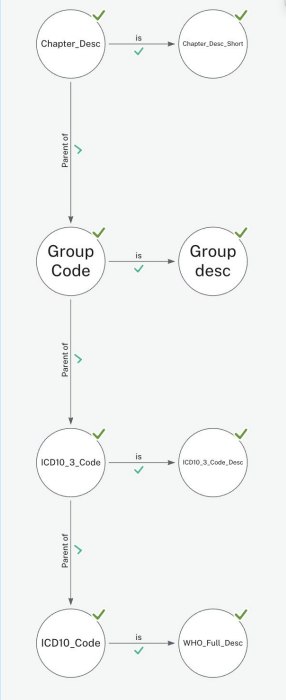
- Used Python to read in data from Excel
- Filtered data into dataframe selecting just the chapter of study (Chapter V - Disease)
- Removing clinically irrelevant rows by subsetting data frame
- Split the dense, existing fields into several shorter fields to better extract and display multi-layered data
- Used dictionaries to take the longer terminology present in the data into abbreviated versions for easier display in graph technology
- Created new, easier to digest fields from the now translated longer terminology with dictionaries functions to use in the graph
- Exported the resultant data frame into a csv file to be read in to Neo4j

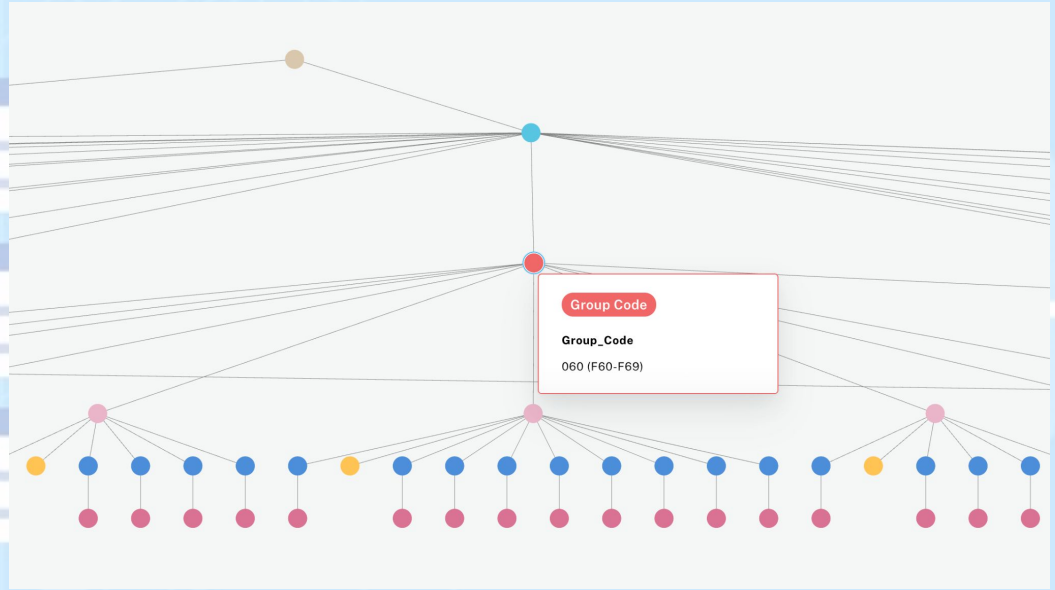
Data Modeling (Conceptual)

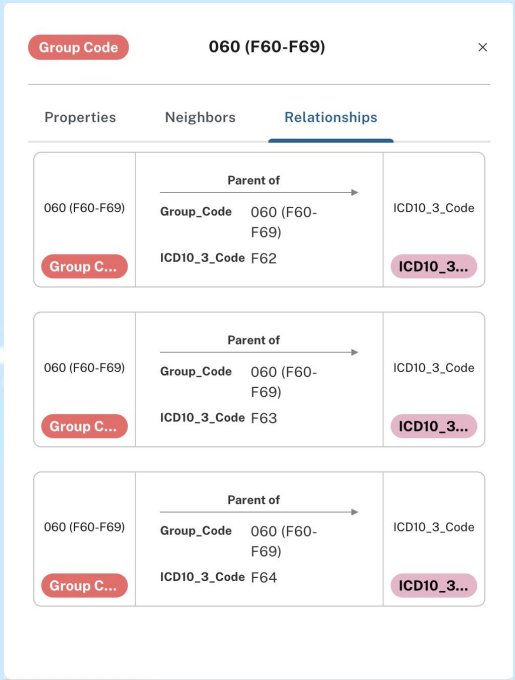
Created using LucidSpark



Data Modeling (Neo4j)







Results

- **Able to create an hierarchical graph database with Neo4j in order to parse and query our dataset in an interactive visualization.**
- **Data points are displayed as nodes, with edges connecting them representing the attributes between nodes.**
- **Allows for quick and effective querying using Neo4j's Cypher Query language.**
- **Neo4j can handle large datasets, making it useful for the creation of a complicated and expansive database.**
- **When querying ICD10 codes, there is a hierarchy and relationship between each subset of codes, and descriptions for each code.**

Challenges:

Challenges faced:

- **Selecting the correct taxonomy and subsection was an important first step in our project. Taxonomies with self-referencing loops would be problematic for the graphing dataset and not entirely intuitive to wrangle.**
- **Further, taxonomies can be massive, so choosing a subsection became important to match the scope of the project at hand. Prior knowledge and experience were relied upon to determine clinical relevance and data points to extract. The dictionaries were time consuming and might not be universally understood by all educational backgrounds.**

Conclusion

Neo4j is a powerful tool for education and hierarchical summarization of dense taxonomies. In order to prepare the taxonomies for Neo4j, we wrote a program that selects a subsection of the taxonomy, removes irrelevant information, splits fields with multiple layers of meaning, and parses the most dense data down to more meaningful phrases. We used this wrangling script on an ICD-10 subsection. Then, we imported the wrangled taxonomy into Neo4j to show a proof-of-concept for how the visualization can occur.

- **Looking Forward:**
  - **Not all taxonomies will be similar to ICD-10 in structure or size. Dealing with the potential loops or external references will require more robust integration with code. Further, some sort of Natural Language Processing might be useful in extracting meaningful layers and data from the dense taxonomy fields to save time and more robustly approach the larger taxonomies.**