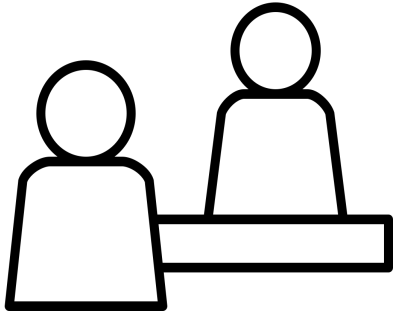
Knowledge Graph For Researchers

Kyle Vincent and Emma Meno
CS 4624: Multimedia, Hypertext, and Information Access
Dr. Edward Fox
Virginia Tech, Blacksburg, VA 24061
April 28, 2020

OUTLINE

Deliverables

Project Team

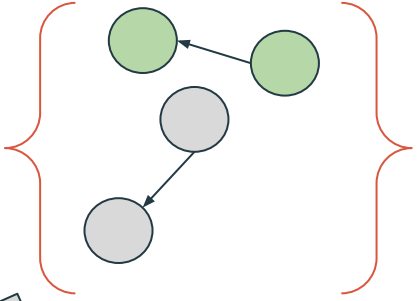


Reads

Project Reports

To Generate

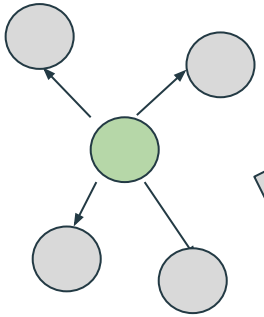
Dataset - Algo - Dataset Tuples



Stored In

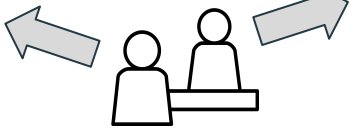
DATABASE

Knowledge Graph



And Make Up

QUERY SERVICE

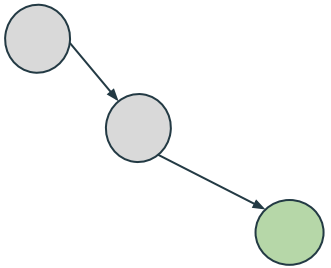


(Input Data Set, Output Data Set)

Subject Matter Experts

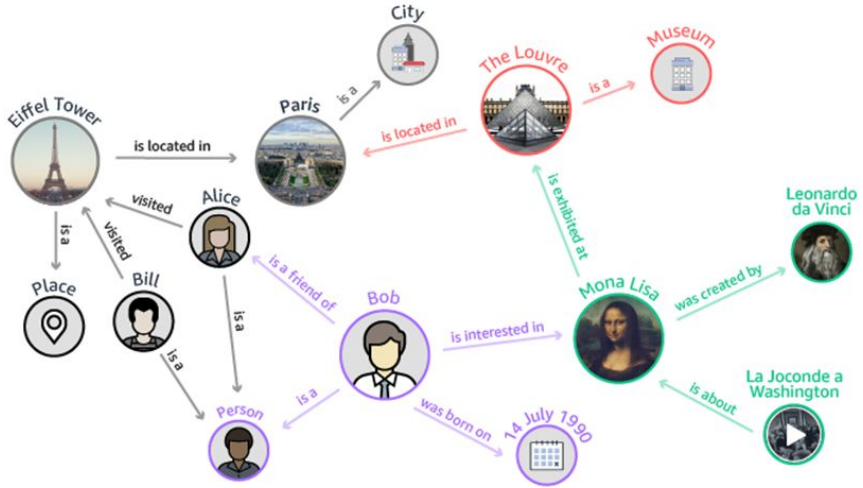
To Return

Workflow Diagram



Project Deliverables

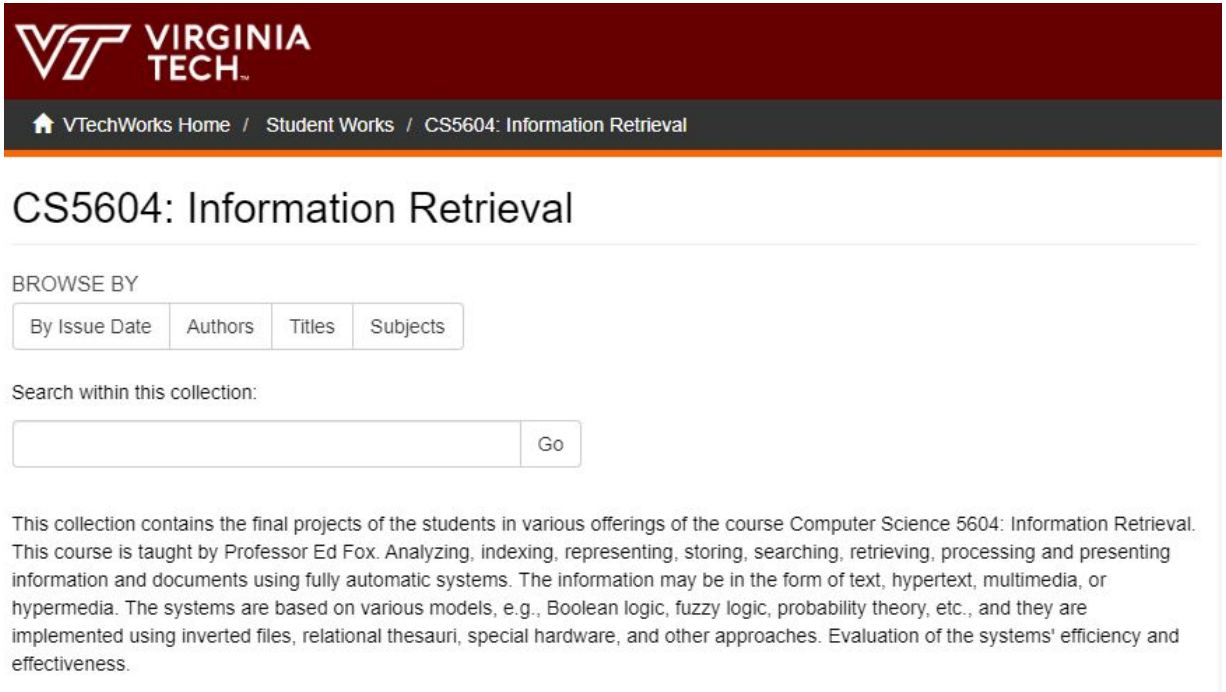
- Build ontology for Twitter-based project info an SME might want to query, answer or calculate
- Build knowledge graph based on created ontology
- Build set of APIs to trigger set of network algorithms based on info queried to graph



Accomplishments

Literature Review: Twitter Data Source

- 2016
- 2017



Built Ontology

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	Task ID	Task Name	Input File(s)	Output File(s)	Dependent Task ID	Functions / Libraries	Manual (Y/N)	Data Collection	Command Line Parameters	Project			Report Names
2	**2016 Reports (CS 5604: Information Retrieval)**												
3	0	Remove stop words and lemmatize	Cleantext from HBase (RDD of tweets)	Cleaned Lemmatized Tweets		StanfordNLP	Y		table name data type for each row-key desired columns	https://vtechworks.lib.vt.edu/handle/10919/73713			CS5604 Fall 2016 Classification Team Final Report
4	1	Manually Classify Tweets	Cleaned Lemmatized Tweets AND Set of Events Classes	Tweet Training Data	0	N/A	Y		collection ID batch size	https://vtechworks.lib.vt.edu/handle/10919/73713			CS5604 Fall 2016 Classification Team Final Report
5	2	Feature Selection	Tweet Training Data	Word Feature Representation	1	Word2Vec Model	N			https://vtechworks.lib.vt.edu/handle/10919/73713			CS5604 Fall 2016 Classification Team Final Report
6	3	Feature Selection	Tweet Training Data	Word Feature Representation	1	Association Rules Based Classifier (provided by Dr. Pereira)	N		path_to_jar_file support threshold block size cluster node base_dir_on-HDFS path_to_training_file path_to_test_file stopwords_file output_directory	https://vtechworks.lib.vt.edu/handle/10919/73713			CS5604 Fall 2016 Classification Team Final Report
7	4	Tweet Classification	Word Feature Representation	Predicted Class of the Given Tweet // Associated Probabilities of Each Class	2,3	Multi-class logistic regression classifier	N		collection number batch size --retrain --metric<training file> <test file>	https://vtechworks.lib.vt.edu/handle/10919/73713			CS5604 Fall 2016 Classification Team Final Report
8	5	Convert to CSV	Cleaned Tweets in HBase	CSV Format Tweets	37	Pig Script	N			https://vtechworks.lib.vt.edu/handle/10919/73739			CS5604: Information and Storage Retrieval Fall 2016 - CMT (Collection Management Tweets)
9	6	Record User, URL, and Tweet Relationships	CSV Format Tweets	Social Network Matrix	5		Y			https://vtechworks.lib.vt.edu/handle/10919/73739			CS5604: Information and Storage Retrieval Fall 2016 - CMT (Collection Management Tweets)
10	7	Calculate Importance Factors for Nodes	Social Network Matrix	Weighted Social Network Matrix	6		Y			https://vtechworks.lib.vt.edu/handle/10919/73739			CS5604: Information and Storage Retrieval Fall 2016 - CMT (Collection Management Tweets)

Nodes.csv

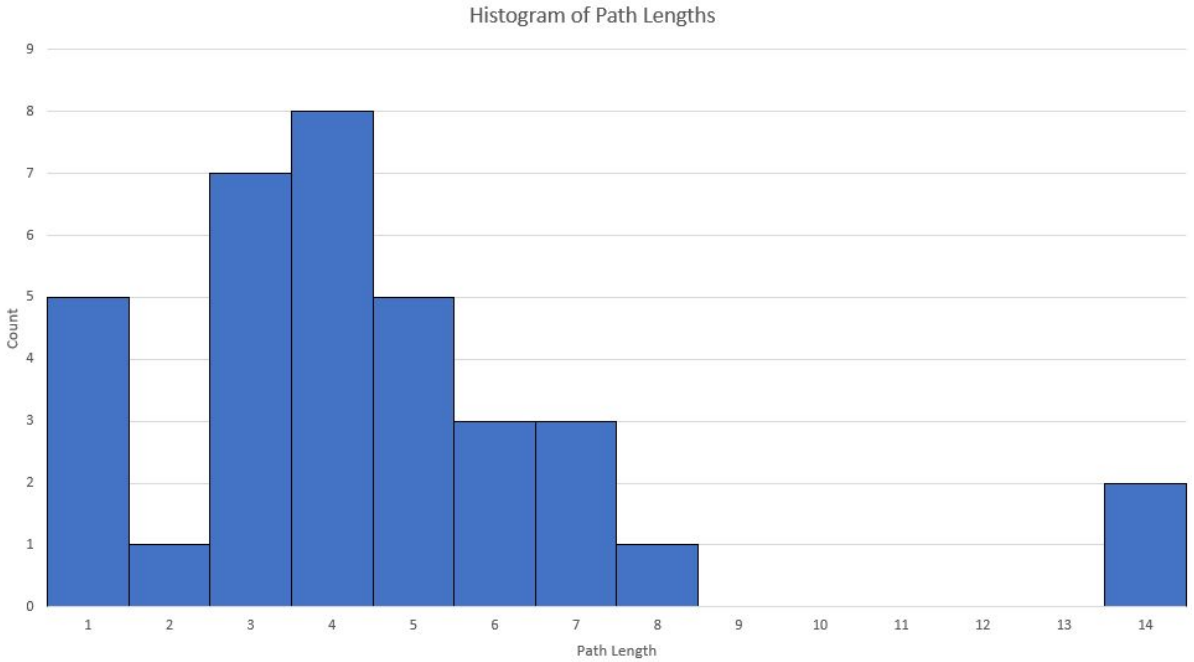
	A	B
1	fileId	name
2	1	Cleantext from HBase (RDD of tweets)
3	2	Cleaned Lemmatized Tweets
4	3	Set of Events Classes
5	4	Tweet Training Data
6	5	Word Feature Representation
7	6	Predicted Class of the Given Tweet
8	7	Associated Probabilities of Each Class
9	8	Cleaned Tweets in HBase

Edges.csv

	A	B	C	D	E	F	G	H	I	J
1	taskId	name	inputId	outputId	functionsAndLibraries	manual	commandLineParameters	reportUrl	reportName	domainCollection
2	1	Remove stop words and lemmatize	1	2	StanfordNLP	Y	table name; data type for each row-key; desired columns	https://vtechworks.lib.vt.edu/handle/10919/73713	CS5604 Fall 2016 Classification Team Final Report	Kentucky Shooting, Newton Shooting, Firefighter Shooting, Hurricane Arthur, Hurricane Sandy, Hurricane Isaac, China Factory Explosion, Texas Plant Explosion, Manhattan Explosion
3	2	Manually Classify Tweets	2	4	N/A	Y	collection ID; batch size	https://vtechworks.lib.vt.edu/handle/10919/73713	CS5604 Fall 2016 Classification Team Final Report	Kentucky Shooting, Newton Shooting, Firefighter Shooting, Hurricane Arthur, Hurricane Sandy, Hurricane Isaac, China Factory Explosion, Texas Plant Explosion, Manhattan Explosion
4	3	Manually Classify Tweets	3	4	N/A	Y	collection ID; batch size	https://vtechworks.lib.vt.edu/handle/10919/73713	CS5604 Fall 2016 Classification Team Final Report	Kentucky Shooting, Newton Shooting, Firefighter Shooting, Hurricane Arthur, Hurricane Sandy, Hurricane Isaac, China Factory Explosion, Texas Plant Explosion, Manhattan Explosion
5	4	Feature Selection	4	5	Word2Vec Model	N		https://vtechworks.lib.vt.edu/handle/10919/73713	CS5604 Fall 2016 Classification Team Final Report	Kentucky Shooting, Newton Shooting, Firefighter Shooting, Hurricane Arthur, Hurricane Sandy, Hurricane Isaac, China Factory Explosion, Texas Plant Explosion, Manhattan Explosion
6	5	Feature Selection	4	5	Association Rules Based Classifier (provided by Dr. Pereira)	N	path_to_jar_file; support threshold; block size; cluster node; base_dir_on-HDFS; path_to_training_file; path_to_test_file; stopwords_file; output_directory	https://vtechworks.lib.vt.edu/handle/10919/73713	CS5604 Fall 2016 Classification Team Final Report	Kentucky Shooting, Newton Shooting, Firefighter Shooting, Hurricane Arthur, Hurricane Sandy, Hurricane Isaac, China Factory Explosion, Texas Plant Explosion, Manhattan Explosion

Path Metrics

Path Length	Count
1	5
2	1
3	7
4	8
5	5
6	3
7	3
8	1
14	2
Total	35



Interim Project Toolkit - The GRANDstack

GraphQL

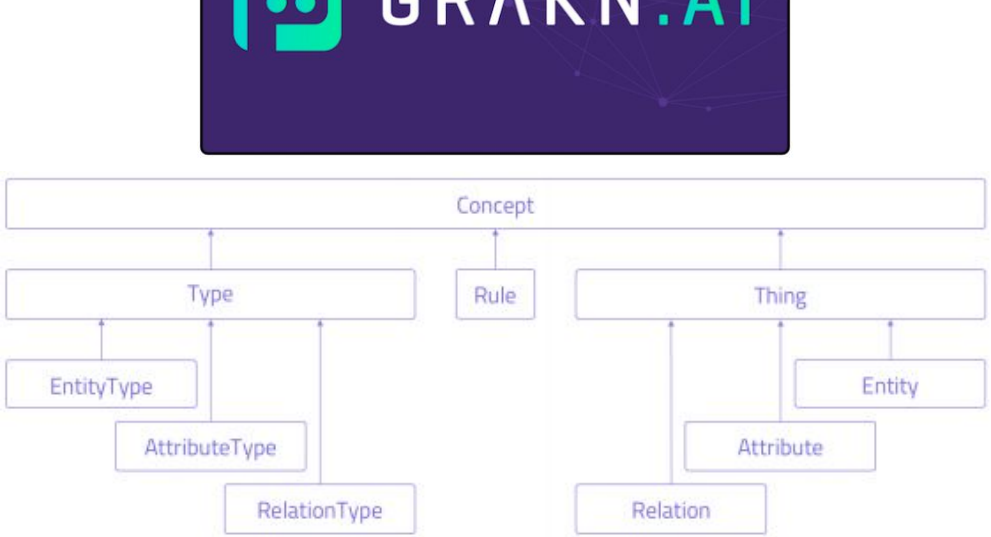
React

Apollo

Neo4j Database

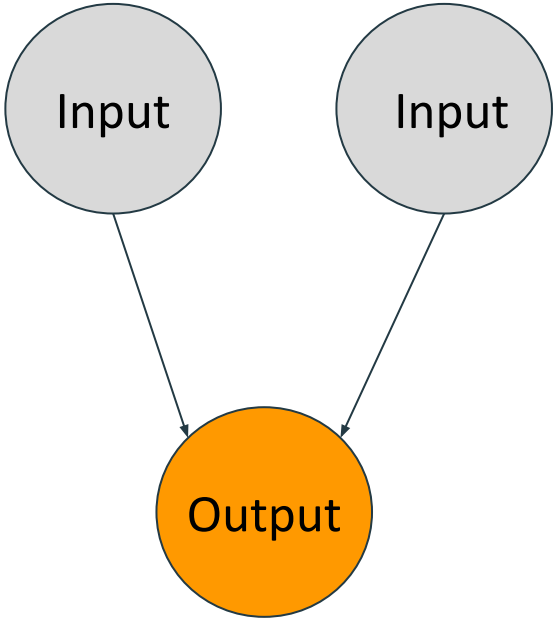
Implemented Project Toolkit - Grakn

- Database storage/interface
- Graql queries for reads, writes, & analysis
- Three Client APIs:
  - Node.js
  - Python
  - Java



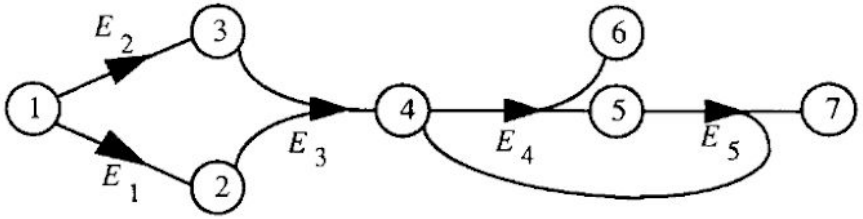
The And/Or Problem

- With traditional node-edge-node relationships, we weren't sure how to differentiate between two scenarios present in our data:
  - A certain output requires Input 1 AND Input 2 to be produced
  - You can use either Input 1 OR Input
- This means the difference between returning 2 paths and 1 path



Solution: The Hypergraph

- Recommended by our Client
- A hyperedge can connect any number of entities
- Now we can distinguish between hyperedges and normal edges



Grakn Database Schema

- Our client recommended a tool called Grakn.
  - Designed for Knowledge Graphs specifically
  - Open Source
  - Allows Hyper-Relationships
- Sample of our Schema:

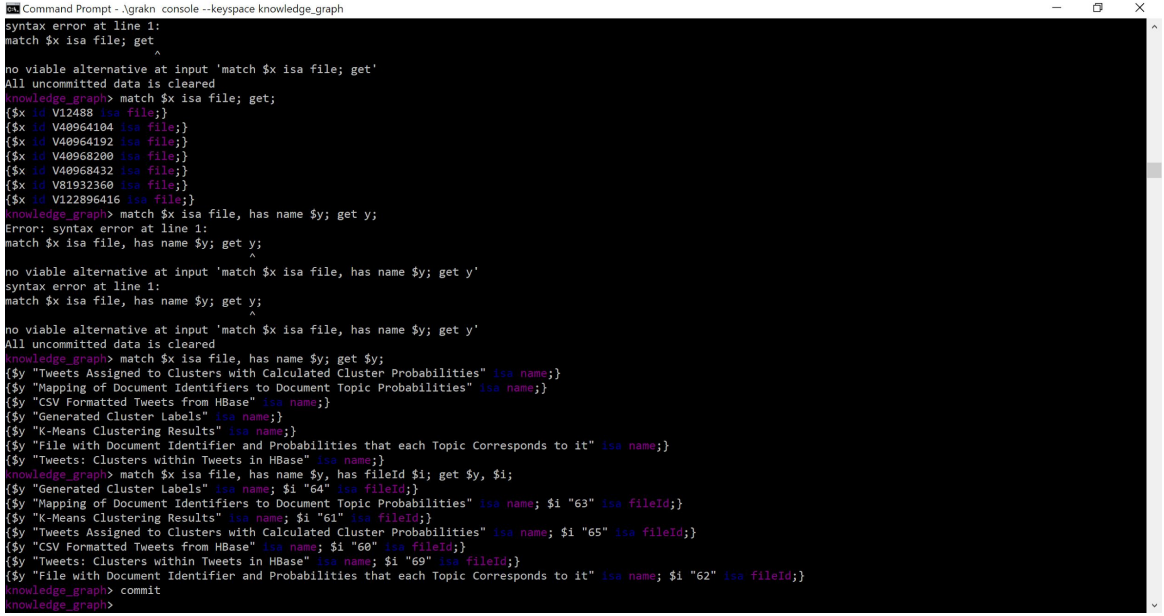
```
define
#Our dataset nodes
file sub entity,
  key fileId,
  has name,
  plays inputfile,
  plays outputfile;

#The tasks that connect the nodes
task sub relation, abstract,
  key taskId,
  has name,
  has inputId,
  has outputId,
  has functionsAndLibraries,
  has manual,
  has commandLineParameters,
  has reportUrl,
  has reportName,
  has domainCollection,
  relates inputfile,
  relates outputfile,
  plays ancestor,
  plays decendent;

#points from input to output
produces sub task,
  relates inputfile,
  relates outputfile;

need-two-inputs sub task,
  relates outputfile,
  relates inputfile;
```

Migration Scripts

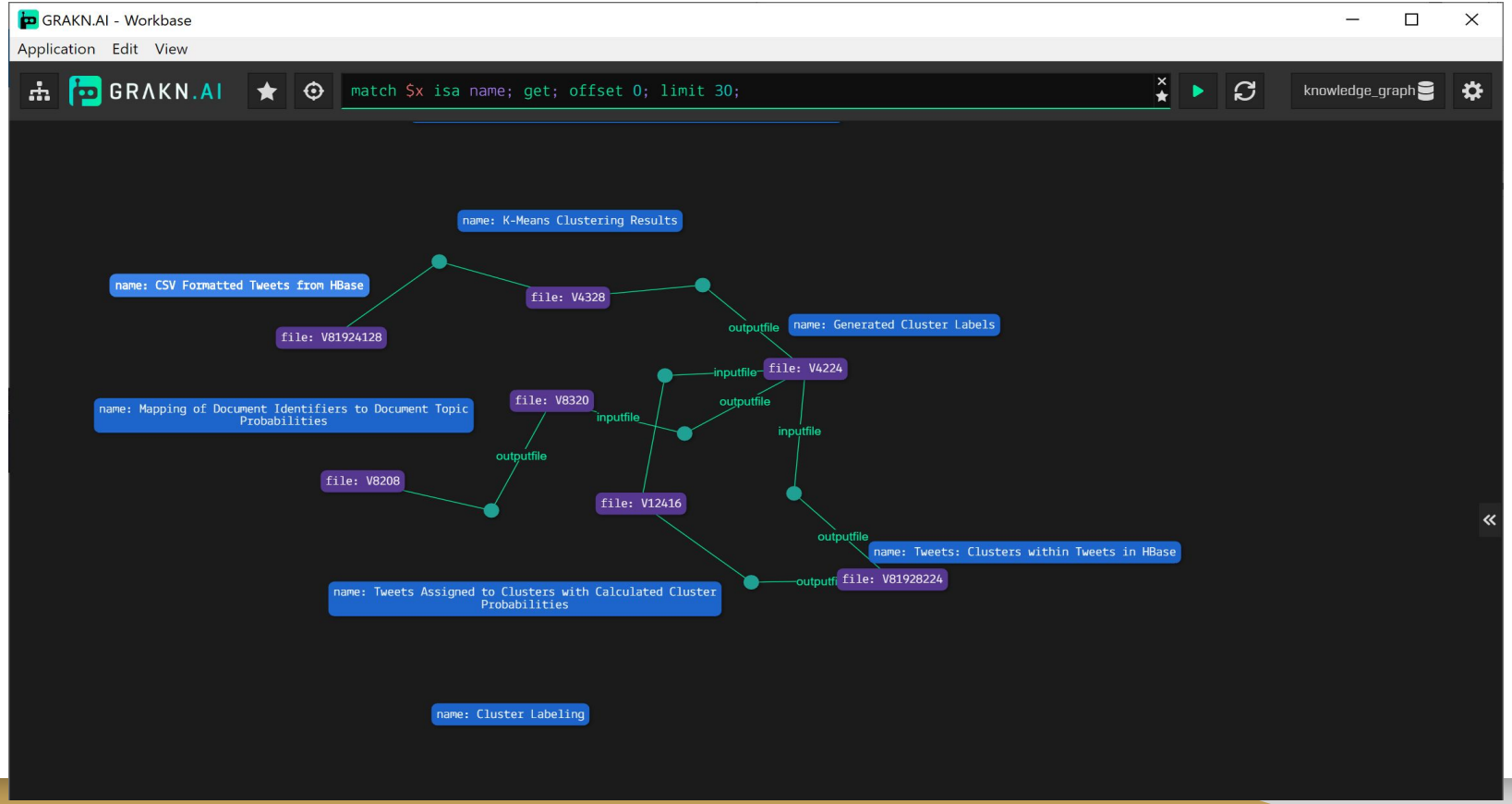
- With schema established, starting migrating the data
- Grakn has three clients:
  - Python
  - Java
  - Node.js
- Wrote data migration scripts that could use these clients and migrate from CSV → Grakn DB
- This proved way more challenging than it had been with Neo4j

Possible User Interface: Console

- One possible way to query the data is the Grakn Console
- It is included with the download of the software
- Uses the Graql language.



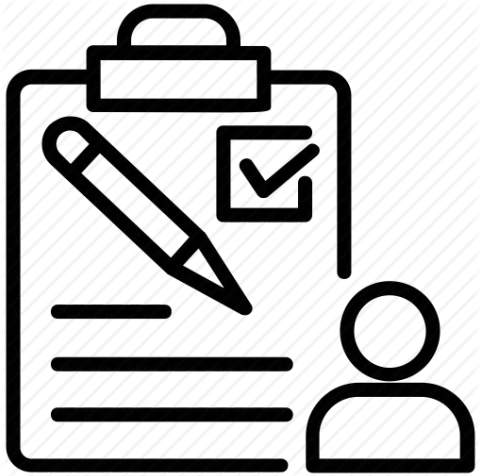
Grakn Workbase UI



Testing and Evaluation

Testing and Evaluation

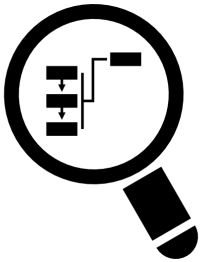
- First test to be conducted by client shortly after project submission
- Next test will be to have a subject matter expert use both the console and the Workbase UI



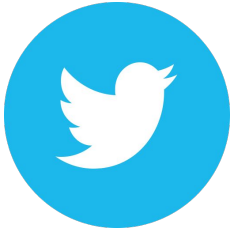
Lessons Learned

Challenges Faced

- Waiting on network science data
- Gathering sufficient material from project reports
- Selecting an appropriate project toolkit
- Transition to virtual learning



Solutions



- Focus on different data ⟶ Twitter
- Including reports from multiple years ⟶ 2016 & 2017
- Shift to Grakn
- Use of Zoom and other online platforms

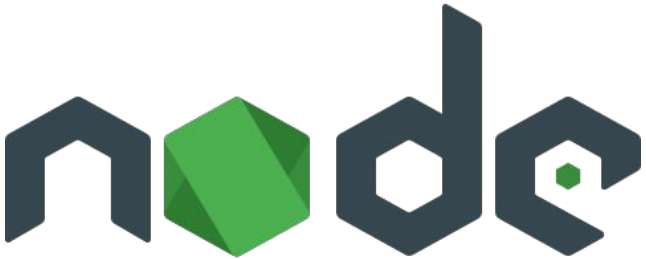
Overall Lessons

- Asking more questions earlier in the design process
- Communicating with the professor, client, and team
- Documentation and project tools

Future Work

Future Work

- Upload the rest of the data to the graph
- Build upon schema to aid better path queries
- Use Node.js Cient to establish a website as front end
- Automate process of parsing through reports



Acknowledgements

Client: Prashant Chandrasekar

- Fifth-year PhD student
- Works with Dr. Fox
- *Our Work*: Proof of Concept for PhD

Professor: Dr. Edward Fox

- Director of Digital Library Research Laboratory

References

”Grakn.AI – Text Mined Knowledge Graphs”, https://www.stockwerk.co.at/event/grakn-ai-textmined-knowledge-graphs/

”Grakn Schema Overview”, https://dev.grakn.ai/docs/schema/overview

Directed Hypergraph and Applications: https://www.sciencedirect.com/science/article/pii/0166218X9390045P

Grakn Documentation: https://grakn.ai/

Questions?