cs415-milestone-2

October 11, 2024

1 cs415 Milestone 2

Within Neo4j Desktop, install the **Neosemantics (n10s)** plugin which enables Neo4j to store and query RDF data. Restart Neo4j after installing the plugin.

Configure Neosemantics for RDF Import

```
// Create required n10s constraint
CREATE CONSTRAINT n10s_unique_uri FOR (r:Resource) REQUIRE r.uri IS UNIQUE;

// Install the Neosemantics procedures
CALL n10s.graphconfig.init();
```

Upload YAGO Data into Neo4j

```
// Load YAGO TTL data into Neo4j
CALL n10s.rdf.import.fetch("file:///path_to_yago_data/yago-4.5.0_ttl.ttl", "Turtle");
```

Explore YAGO Data in Neo4j

```
MATCH (n) RETURN n LIMIT 10;

MATCH (a)-[r]->(b) RETURN a, r, b LIMIT 10;
```

Optimize Data for Queries

```
CREATE INDEX ON :Entity(name);
```

Under here is a Python implementation to import the .ttl file into our Neo4j database. This currently fails during import but to my understanding, because of cache and RAM constraints on my local machine. It is included here to show the effort, and my be utilized in the future.

Downloaded the YAGO 4.5 dataset. (https://yago-knowledge.org/data/yago4.5/)

[1]:

```
pip install rdflib-neo4j
```

```
Defaulting to user installation because normal site-packages is not writeable
Requirement already satisfied: rdflib-neo4j in
/Users/adams/Library/Python/3.9/lib/python/site-packages (1.1)
```

```
Requirement already satisfied: rdflib>=7.0.0 in
/Users/adams/Library/Python/3.9/lib/python/site-packages (from rdflib-neo4j)
(7.0.0)
Requirement already satisfied: neo4j>=5.0.0 in
/Users/adams/Library/Python/3.9/lib/python/site-packages (from rdflib-neo4j)
(5.25.0)
Requirement already satisfied: pytz in
/Users/adams/Library/Python/3.9/lib/python/site-packages (from
neo4j>=5.0.0->rdflib-neo4j) (2024.2)
Requirement already satisfied: isodate<0.7.0,>=0.6.0 in
/Users/adams/Library/Python/3.9/lib/python/site-packages (from
rdflib>=7.0.0->rdflib-neo4j) (0.6.1)
Requirement already satisfied: pyparsing<4,>=2.1.0 in
/Users/adams/Library/Python/3.9/lib/python/site-packages (from
rdflib>=7.0.0->rdflib-neo4j) (3.1.4)
Requirement already satisfied: six in /Library/Developer/CommandLineTools/Librar
y/Frameworks/Python3.framework/Versions/3.9/lib/python3.9/site-packages (from
isodate<0.7.0,>=0.6.0->rdflib>=7.0.0->rdflib-neo4j) (1.15.0)

[notice] A new release of pip is
available: 24.0 -> 24.2
[notice] To update, run:
/Library/Developer/CommandLineTools/usr/bin/python3 -m pip install

--upgrade pip
Note: you may need to restart the kernel to use updated packages.
```

[2]:

```python
from rdflib import Namespace, Graph, URIRef, RDF, SKOS, Literal
from rdflib_neo4j import Neo4jStore, Neo4jStoreConfig, HANDLE_VOCAB_URI_STRATEGY
from neo4j import GraphDatabase
```

[3]:

```python
# Neo4j credentials
URI = "neo4j://localhost:7687"
AUTH = ("neo4j", "12345678")

with GraphDatabase.driver(URI, auth=AUTH) as driver:
    driver.verify_connectivity()
    print("Connection established.")
```

```
Connection established.
```

[3]:

```python
# Connect to Neo4j DB
DB_URI="neo4j://localhost:7687"
DB_USERNAME="neo4j"
DB_PWD="12345678"

auth_data = {'uri': DB_URI,
             'database': "neo4j",
             'user': DB_USERNAME,
             'pwd': DB_PWD}

# Define your custom mappings & store config
config = Neo4jStoreConfig(auth_data=auth_data,
                          handle_vocab_uri_strategy=HANDLE_VOCAB_URI_STRATEGY.
 ↪IGNORE,
                          batching=True)

file_path = '/Users/adams/wsu/cpts415/yago-4.5.0.2/yago-facts.ttl'

# Create the RDF Graph, parse & ingest the data to Neo4j, and close the␣
 ↪store(If the field batching is set to True in the Neo4jStoreConfig, remember␣
 ↪to close the store to prevent the loss of any uncommitted records.)
neo4j_aura = Graph(store=Neo4jStore(config=config))
# Calling the parse method will implictly open the store
neo4j_aura.parse(file_path, format="ttl")
neo4j_aura.close(True)
```

```
Uniqueness constraint on :Resource(uri) found.


The store is now: Open
```

```
The Kernel crashed while executing code in the current cell or a previous cell.


Please review the code in the cell(s) to identify a possible cause of the␣
 ↪failure.


Click <a href='https://aka.ms/vscodeJupyterKernelCrash'>here</a> for more info.


View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.
```

```
Canceled future for execute_request message before replies were done
```

```
Canceled future for execute_request message before replies were done.


View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.
```

The class below attempts to parse the .ttl file in batches. Currently it produces the same issue, but is a approach we can dive deeper into in the future.

[4]:

```python
class Neo4jHandler:

    def __init__(self, uri, user, password):
        self.driver = GraphDatabase.driver(uri, auth=(user, password))

    def close(self):
        self.driver.close()

    def add_triple_batch(self, triples):
        with self.driver.session() as session:
            session.write_transaction(self._create_triples, triples)

    @staticmethod
    def _create_triples(tx, triples):
        for subj, pred, obj in triples:
            query = """
            MERGE (s:Resource {uri: $subj})
            MERGE (p:Resource {uri: $pred})
            MERGE (o:Resource {uri: $obj})
            CREATE (s)-[:RELATED_TO]->(o)
            """
            tx.run(query, subj=str(subj), pred=str(pred), obj=str(obj))

def batch_parse_and_ingest(file_path, batch_size=1000):
    rdf_graph = Graph()
    neo4j_handler = Neo4jHandler("bolt://localhost:7687", "neo4j", "12345678")

    triples = []
    count = 0
    for subj, pred, obj in rdf_graph.parse(file_path, format="ttl"):
        triples.append((subj, pred, obj))
        count += 1

        if count % batch_size == 0:
            neo4j_handler.add_triple_batch(triples)
            print(f"Processed {count} triples")
            triples = []  # Clear the batch after committing

    if triples:
        neo4j_handler.add_triple_batch(triples)
        print(f"Processed {count} triples (final batch)")

    neo4j_handler.close()

file_path = '/Users/adams/wsu/cpts415/yago-4.5.0.2/yago-facts.ttl'
batch_parse_and_ingest(file_path, batch_size=1000)
```

```
The Kernel crashed while executing code in the current cell or a previous cell.


Please review the code in the cell(s) to identify a possible cause of the␣
  ↪failure.


Click <a href='https://aka.ms/vscodeJupyterKernelCrash'>here</a> for more info.


View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.
```

```
Canceled future for execute_request message before replies were done
```

```
Canceled future for execute_request message before replies were done.


View Jupyter <a href='command:jupyter.viewOutput'>log</a> for further details.
```