twiml

AI Enterprise Workflow Study Group

Course 1, Week 2

2/29/2020

Agenda

- Check in
- C1W2 key concepts
- Discussion
- Next steps

About the Courses

The AI Enterprise Workflow specialization on Coursera consists of a series of six courses. Each of the courses is designed to be completed in two weeks, except for course six, which has two weeks of study and two weeks for a capstone project.

The courses aim to give learners with some data science experience an understanding of real-world processes and workflows used to deliver AI and ML models in the enterprise.

Our study group will meet weekly on Saturdays at 9:30 am Pacific Time and will continue for 14 weeks.

You should plan to spend around 8 hours a week to complete the material.

Course & Study Group Schedule

AI Enterprise Workflow Study Group		
Session	**Topic**	**Date**
Overview Webinar	Webinar with instructor, Ray Lopez	15-Feb
Course 1 Week 1	Course intro	22-Feb
Course 1 Week 2	Data ingestion, cleaning, parsing, assembly	29-Feb
Course 2 Week 1	Exploratory data analysis & visualization	7-Mar
Course 2 Week 2	Estimation and NHT	14-Mar
Course 3 Week 1	Data transformation and feature engineering	21-Mar
Course 3 Week 2	Pattern recognition and data mining best practices	28-Mar
Course 4 Week 1	Model evaluation and performance metrics	4-Apr
Course 4 Week 2	Building machine learning and deep learning models	11-Apr
Course 5 Week 1	Deploying models	18-Apr
Course 5 Week 2	Deploying models using Spark	25-Apr
Course 6 Week 1	Feedback loops and monitoring	2-May
Course 6 Week 2	Hands on with OpenScale and Kubernetes	9-May
Course 6 Week 3	Captsone project week 1	16-May
Course 6 Week 4	Captsone project week 2	23-May

Course 1 Week 2 learning objectives

1. Articulate the use case for sparse matrices as a target destination for data ingestion
2. Explain the purpose of testing in data ingestion
3. Know the initial steps that can be taken towards automation of data ingestion pipelines
4. Know where data science and data engineering have the most overlap in the AI workflow
5. Explain the fundamental aspects of the data ingestion process

Key concepts

- Data ingestion
- ETL, etc.
- Data science vs data engineering
- Sparse matrices
- Data testing
- Automation of data ingestion pipelines

Data ingestion

“The collection of data followed by the readying of that data for use in the AI workflow.”

- Extract data
- Ensure quality
- Load to target destination

“Make it work, then make it better”

ETL etc.

- Extract, Transform & Load
- Historically used to load data into EDW
- More recently, model has shifted to ELT and data lakes
- Data lake => large collections of data in natural formats
- “Data ingestion” used vs. ETL to cover all these alternatives, emphasizes use of data in place

Data science vs data engineering

- Data engineering => developing, constructing, testing and maintaining architectures to deploying large scale processing systems and databases.
- In a large enterprise (or fast-growth startup), data engineers may be available to assist with building or hardening/productionalizing a data ingestion pipeline.
- Roles vary widely company to company. New roles have emerged, e.g. machine learning engineers.

Sparse matrices

A matrix in which most elements are zero. Often found in e.g. word frequency (TF-IDF), one-hot encoding, user-term matrices for recommendations.

Sparse matrices can be used during dev/test as an alternative to complete data pipelines. Middle ground between comprehensive data warehouse with extensive test coverage and collection of text files.

- Require less memory
- Can be faster to work with

Sparse matrix formats

SciPy supports these and other sparse matrix formats:

- coo_matrix - COOrdinate format sparse matrix
- csc_matrix - Compressed Sparse Column removes redundancy in columns
- csr_matrix - Compressed Sparse Row removes redundancy in rows

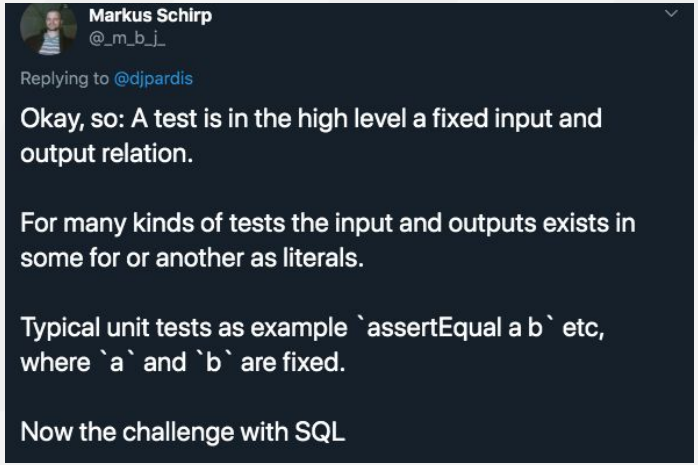
Use tocoo, tocsc, tocsr, todense to transform between formats

Use vstack, hstack to append to (stack) sparse matrices vertically, horizontally

Use save_npz, load_npz to save, load

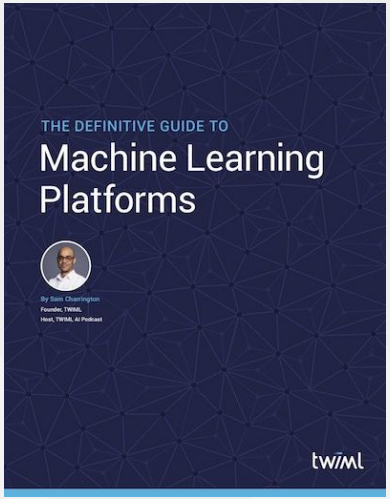
Data testing

- Quality/sanity check your data to avoid “garbage in, garbage out”
- “Unit Testing” for data is an immature discipline, but lots of folks working on it.
- Interesting thread on unit tests for SQL: https://twitter.com/_m_b_j_/status/1232760813380829185



Automation of data ingestion pipelines

- Lots of work happening here.
- Progressively more complex approaches like:
  - Executable file vs notebook
  - Cookiecutter Data Science (https://github.com/drivendata/cookiecutter-data-science/)
  - DAG-based tools



https://twimlai.com/aiplatforms/

Case Study #1 - Data Ingestion

- Zip files have been updated to correct missing data.
- Data assets vs local files accessible to notebook. See my comment in Slack.

Discussion

What did you learn?

What stumbling blocks did you run into?

How do these lessons relate to your experience?

What did you learn/find interesting in this week's lesson?

What are you doing as homework?

What interesting resources have you found?

Other?

Next steps

Congratulations on completing Course 1!

Next week we move on to Course 2, Week 1 (EDA & dataviz)

Chime in on Slack if you run into any issues or want to share any observations

Prepare your questions, discussion points, etc. for next week's meetup