A Master Class in Building Production-Grade NLP Pipelines

Presented By:
Sri Krishnamurthy, CFA, CAP
sri@quantuniversity.com
www.quantuniversity.com

10/15/2019
MathWorks Conference
New York, NY
Speaker bio

Sri Krishnamurthy
Founder and CEO
QuantUniversity

• Quant, Data Science & ML practitioner
• Prior Experience at MathWorks, Citigroup and Endeca and 25+ financial services and energy customers.
• Columnist for the Wilmott Magazine
• Teaches Data Science/AI at Northeastern University, Boston
• Reviewer: Journal of Asset Management
About QuantUniversity

• Boston-based Data Science, Quant Finance and Machine Learning training and consulting advisory
• Trained more than 1000 students in Quantitative methods, Data Science, ML and Big Data Technologies
• Building a platform for operationalizing AI and Machine Learning in the Enterprise
Agenda

1. Model Life Cycle Management & Pipelines
2. Productionizing Pipelines: An NLP Case study
Machine Learning Workflow

Data Engineer, Dev Ops Engineer

Data Scraping/Ingestion → Data Exploration → Data Cleansing and Processing → Feature Engineering

Robotic Process Automation (RPA) (Microservices, Pipelines)

Risk Management/Compliance (All stages)

Model Deployment/Inference
- SW: Web/Rest API
- HW: GPU, Cloud
- Monitoring

Model Selection
- AutoML
- Model Validation
- Interpretability

Model Evaluation & Tuning
- Hyper-parameter tuning
- Parameter Grids

Modeling
- Supervised
  - Regression
  - KNN
  - Decision Trees
  - Naive Bayes
  - Neural Networks
  - Ensembles
- Unsupervised
  - Clustering
  - PCA
  - Autoencoder

Data Scientist/Quants

Software/Web Engineer

Analysts & Decision Makers

• RMS • MAPS • MAE • Confusion Matrix • Precision/Recall • ROC

• Regression • KNN • Decision Trees • Naive Bayes • Neural Networks • Ensembles

• Clustering • PCA • Autoencoder
Figure 1: Only a small fraction of real-world ML systems is composed of the ML code, as shown by the small black box in the middle. The required surrounding infrastructure is vast and complex.

Source: Sculley et al., 2015 "Hidden Technical Debt in Machine Learning Systems"
The reproducibility challenge

IS THERE A 
REPRODUCIBILITY 
CRISIS?

A Nature survey lifts the lid on how researchers view the 'crisis' rocking science and what they think will help.

BY MONYA BAKER

http://www.nature.com/news/1-500-scientists-lift-the-lid-on-reproducibility-1.19970
Repeatability (Same team, same experimental setup)
- The measurement can be obtained with stated precision by the same team using the same measurement procedure, the same measuring system, under the same operating conditions, in the same location on multiple trials. For computational experiments, this means that a researcher can reliably repeat her own computation.

Replicability (Different team, same experimental setup)
- The measurement can be obtained with stated precision by a different team using the same measurement procedure, the same measuring system, under the same operating conditions, in the same or a different location on multiple trials. For computational experiments, this means that an independent group can obtain the same result using the author’s own artifacts.

Reproducibility (Different team, different experimental setup)
- The measurement can be obtained with stated precision by a different team, a different measuring system, in a different location on multiple trials. For computational experiments, this means that an independent group can obtain the same result using artifacts which they develop completely independently.

https://www.acm.org/publications/policies/artifact-review-badging
Many choices

Languages

Frameworks

Platforms
Processes are chaotic

Discovery | Design | Development | Deployment

---|---|---|---

Planning

Reality
### Multiple stakeholders

<table>
<thead>
<tr>
<th>Engineering/IT</th>
<th>Quants/Data Scientists</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Scaling</td>
<td>• New Algorithms</td>
</tr>
<tr>
<td>• Structuring</td>
<td>• Try new methods</td>
</tr>
<tr>
<td>• Design of Experiments</td>
<td>• Effect of Parameters and Hyper Parameters</td>
</tr>
<tr>
<td>• Data Parallel/Task Parallel</td>
<td></td>
</tr>
</tbody>
</table>
Which Model to choose?

Client Objective:
• Build the best forecasting model that has a MAPE of 5% or less

Result:
• Regression – 7% MAPE
• Neural Networks – 4% MAPE
• Random Forest – 5% MAPE

Client choice:
• Regression despite being the worst of the top-3 models
• ”I won’t deploy anything that I don’t understand”

Elements of Model Risk Management

1. **Model Governance structure**: Addresses regulatory requirements, roles, responsibilities, oversight, control and escalation procedures.

2. **Model Lifecycle management**: Addresses the processes involved in the design, development, testing, deployment and use of models. Also addresses testing and documentation plans and change management.

3. **Model Review and Validation Process**: Addresses internal and external model review, verification, validation and ongoing monitoring of models (both qualitative and quantitative).
AI Governance is gaining focus

AI system: An AI system is a machine-based system that can, for a given set of human-defined objectives, make predictions, recommendations, or decisions influencing real or virtual environments. AI systems are designed to operate with varying levels of autonomy.

AI system lifecycle: AI system lifecycle phases involve: i) ‘design, data and models’; which is a context-dependent sequence encompassing planning and design, data collection and processing, as well as model building; ii) ‘verification and validation’; iii) ‘deployment’; and iv) ‘operation and monitoring’. These phases often take place in an iterative manner and are not necessarily sequential. The decision to retire an AI system from operation may occur at any point during the operation and monitoring phase.

AI knowledge: AI knowledge refers to the skills and resources, such as data, code, algorithms, models, research, know-how, training programmes, governance, processes and best practices, required to understand and participate in the AI system lifecycle.

AI actors: AI actors are those who play an active role in the AI system lifecycle, including organisations and individuals that deploy or operate AI.

Stakeholders: Stakeholders encompass all organisations and individuals involved in, or affected by, AI systems, directly or indirectly. AI actors are a subset of stakeholders.

NLP pipeline

Stage 1: Data Ingestion from Edgar
Stage 2: Pre-Processing
Stage 3: Invoking APIs to label data
Stage 4: Compare APIs
Stage 5: Build a new model for sentiment Analysis

- Amazon Comprehend API
- Google API
- Watson API
- Azure API
Components that needs to be tracked

- Model specific
- Tests
- Data versions

- Model params
- Hyper parameters
- Pipeline specifications

- Dependencies
- Lineage/Provenance of individual components

- Programming environment
- Execution environment
- Hardware specs
  - Cloud
  - GPU
Provenance and Lineage of pipelines

Figure 3: Running multiple pipeline versions

Schemas proposed

Schemas proposed

MLFlow

Default  >  Run bf34330b7ebc4a07abe335b9a2a6ce2a

Date: 2019-10-05 20:56:09
Source: `sklearn_elasticnet_wine`
Entry Point: `main`
Duration: 1.6min
Run Command
```
mlflow run file:///Users/srimacpro/mlflow#examples/sklearn_elasticnet_wine -v 60c71fae70aef9841d60f63e023f129c02dec19 -P alpha=0.5 -P l1_ratio=0.1
```

Notes
None

Parameters

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>alpha</td>
<td>0.5</td>
</tr>
<tr>
<td>l1_ratio</td>
<td>0.1</td>
</tr>
</tbody>
</table>
DVC tracks ML models and data sets

DVC is built to make ML models shareable and reproducible. It is designed to handle large files, data sets, machine learning models, and metrics as well as code.
GoCD

FREE & OPEN SOURCE CI/CD SERVER

Easily model and visualize complex workflows with GoCD.

TEST DRIVE GOCD

GoCD supports Pipelines as Code. See the Benefits
Implementation Approaches
Current approaches

Current approaches

Related work

Figure 1: ProvChain System Interaction.


Focus on Cloud data provenance using Blockchain
Related work

DataProv: Built on top of Ethereum, the platform utilizes smart contracts and open provenance model (OPM) to record immutable data trails.

Figure 2: Voting procedure for a Document change.

Ramachandran, Aravind & Kantarcioglu, Dr. (2017). Using Blockchain and smart contracts for secure data provenance management.
Related work


Trusted AI and provenance of AI models
NLP Case study
Goal

• Understanding sentiments in Earnings call transcripts

CORPORATE PARTICIPANTS
Dana Quattrochi athenahealth, Inc. - IR
Jonathan Bush athenahealth, Inc. - Chairman and CEO
Tim Adams athenahealth, Inc. - CFO
Andy Hurd Epocrates - President and CEO
Rob Cosinuke athenahealth, Inc. - Chief Marketing Officer

CONFERENCE CALL PARTICIPANTS
Sean Wieland Piper Jaffray & Co. - Analyst
Jamie Stockton Wells Fargo Securities, LLC - Analyst
George Hill Citigroup - Analyst
Greg Bolan Sterne, Agee & Leach - Analyst
Ryan Daniels William Blair & Company - Analyst
Rich Close Avondale Partners - Analyst
Sandy Draper Raymond James - Analyst
David Bayer Northland Securities - Analyst
Dave Windley Jefferies & Co. - Analyst
Charles Rhyee Cowen and Company - Analyst
Bret Jones Oppenheimer & Co. - Analyst
Michael Cherny ISI Group - Analyst
Tony Bartsch Park West Asset Management - Analyst

PRESENTATION
Operator
Welcome to the athenahealth conference call. I would now like to turn the call over to Ms. Dana Quattrochi. You may now begin.

Dana Quattrochi - athenahealth, Inc. - IR
Good morning and thank you for joining us. With me on the call today is Jonathan Bush, our Chairman and CEO; Tim Adams, our Chief Financial Officer; Rob Cosinuke, our Chief Marketing Officer; and Andy Hurd, President and CEO of Epocrates.
Challenges

- Interpreting emotions
- Labeling data

Options

- APIs
- Human Insight
- Expert Knowledge
- Build your own
NLP pipeline

Stage 1: Data Ingestion from Edgar
Stage 2: Pre-Processing
Stage 3: Invoking APIs to label data
Stage 4: Compare APIs
Stage 5: Build a new model for sentiment Analysis

- Amazon Comprehend API
- Google API
- Watson API
- Azure API
QuSandbox- The platform for governing Data Science and AI workflows in the Enterprise
QuSandbox research suite

- Productionize and share
- Standardize workflows
- Track experiments
- Prototype, Iterate and tune
The four components that need to be encapsulated for reproducible pipelines

- Code
- Data
- Environment
- Process
QuSandbox

MATLAB
Twitter Data Mining
This project provides sample project to show users how to mine, store and process data from Twitter. Users can see an exploratory data analysis on the tweets shared all across the globe with the hashtag #2020.

TensorFlow
Base TensorFlow Notebook
This is designed for easily diving into TensorFlow, through examples. For readability, it includes both notebooks and source codes with explanation.

TensorFlow
Deep Q Learning
The experiment shows a simple implementation of Deep Q learning and how to apply it to play cartpole.

MATLAB
Lending Club Clustering
This project shows users how to implement clustering analysis based on LendingClub data.

handson-ml/1510696922847
A series of Jupyter notebooks that walk you through the fundamentals of Machine Learning and Deep Learning in python using Scikit-Learn and Tensorflow.

Github URL -
DockerHub URL -
Qu Credits: 79

Run on QuSandbox
Run from Command Line

Amazon Web Service
Choose the AWS Machine Type?

Do you want to load one project or all projects?

Single Project:

Duration (in hours):

Terms & Conditions | Privacy Policy | Cookie Policy
Model Management Studio

QuSandbox Model Management Studio

QuSandbox-Edgar-Pipeline

QuSandbox-Phase-1-Env
QuSandbox-Model-Phase-1
BLOCK-PHASE-1-QUSSandbox
ADD ENTITY

QuSandbox-Phase-2-Env
QuSandbox-Model-Phase-2
BLOCK-QUSSandbox-2
ADD ENTITY

QuSandbox-Phase-3-Env
QuSandbox-Model-Phase-3
BLOCK-QUSSandbox-3
ADD ENTITY

Stage 1

Stage 2

Stage 3
Terms

- **JDF**: Job Definition File; A DSL for representing Model Pipelines
- **Stage**
- **Entity**
  - Model
  - Data
  - Environment
- **Version format**
Sentiment Analysis with QuSandbox

QuantUniversity Team

Abstract: EDGAR, the Electronic Data Gathering, Analysis, and Retrieval system, performs automated collection, validation, indexing, acceptance, and forwarding of submissions by companies and others who are required by law to file forms with the U.S. Securities and Exchange Commission (the "SEC"). The database contains a wealth of information about the Commission and the securities industry which is freely available to the public via the Internet (HTTPS)[2]. In this project, we intend to analyze the sentiment of each paragraph in a form-425 file of a specific company downloaded from Edgar. This form-425 file is phone call transaction which contains a lot of dialogues between operator and clients. Thus, Sentiment analysis will be applied on each paragraph in the file and return either a sentiment label or a sentiment score or both of them to the user. Also, we want to test and compare the performances of different NLP APIs in this project. Amazon Comprehend, Microsoft Azure, Google Cloud and IBM Watson will be used in the project to analysis the same file. Moreover, before the actual analysis, a crawler and a preprocessing phase need to be designed in order to have a analysable data set.

This sprint involved integration of the Model management Studio, QuSandbox and the ResearchHub

Additional Links

1. **Model Management workflow**
   - We illustrate how QuSandbox can be used to set up a pipeline to enable crawling, pre-processing and prediction of sentiments. The workflow is an illustration of the key concepts and terminologies used to structure the pipeline.

2. **Using CLI tools to automate the workflow**
   - This is to illustrate how the CLI can be used to invoke actions on the QuSandbox. The CLI tools enable access to the QuSandbox without the need of the Model Management Studio. This enables integration with third party scheduling tools.

3. **Sentiment analysis workflow**
   - This experiment has 3 stages. In the first stage, we crawl the form 425 data from the Edgar website and store it to a Amazon S3 bucket. In the second stage, we pre-process the data and store it back to the Amazon S3 bucket. In the third stage, we let the Quant perform sentiment analysis using the API of his/her choice. We provide Jupyter notebooks for 4 APIs: 1. Amazon's Comprehend API, 2. Google API, 3. IBM Watson API, 4. Microsoft API.
Architecture: What’s tracked?

Metadata
- Data about the information to be tracked
- Includes version number, timestamps, user information, MD5 of the artifacts and high-level notes

Data
- Pipelines, custom DSL, standard formats for representing models
- Events (Updates, rollbacks)
- JSON, Amazon ION, YAML,

Artifacts
- Model Pickle files, ONYX, COREML, Model params
- Data, blobs etc.
Architectures supported

Blockchain-based:
• QLDB
• Ethereum

Non-Blockchain-based:
• MongoDB
Demo
Future work

- Support for ONYX, CoreML
- Integration with:
  - MLFlow, DVC, GoCD
- Integration with SCM systems
  - Github, SVM
- Tracking Back tests
- Push Architecture -> Event-Driven Architecture
- Enriched Analytics
- Roles and Authorization