Transformative Applications of Data Science:

Spatiotemporal Studies, Graph & Deep Learning To Solve Grand Challenges

Roger H. French

Materials Data Science for Stockpile Stewardship (MDS^3) Center of Excellence
Materials Science & Eng. Dept., Computer & Data Sciences Dept.
Case Western Reserve University, Cleveland OH 44106 USA

A selection of Lifetime Extension Articles

roger.french@case.edu
http://dmseg5.case.edu/people/faculty.php?id=rxf131

A BHEF CASE STUDY

CREATING A MINOR IN

APPLIED DATA SCIENCE

Case Western Reserve University Engages Business Leaders to Produce T-Shaped Professionals

THROUGH THE COLLABORATION of its business and higher education members, the Business-Higher Education Forum (BHEF) launched the National Higher Education and Workforce Initiative (HEWI) to create new undergraduate pathways in high-skill, high-demand fields such as data science and analytics. Data science and analytics must be integrated with T-shaped skills, such as critical thinking, collaboration, and effective communication, which are critical for all graduates entering the 21st century workforce. Knowledge of data science and analytics in recent years has become as fundamental as any other skill for graduates' career readiness. BHEF's Strategic Business Engagement Model with higher education addresses this demand by moving the two sectors from transactional relationships to strategic partnerships through five strategies:

1. **ENGAGE** corporate leadership;
2. **FOCUS** corporate philanthropy on undergraduate education;
3. **IDENTIFY** and tap core competencies and expertise;

undergraduate education.

This case study examines how BHEF member Case Western Reserve University (Case Western Reserve) is integrating T-shaped skills into a minor in applied data science.

PROGRAM OVERVIEW

THE APPLIED DATA SCIENCE (ADS) MINOR AT CASE WESTERN RESERVE serves as a national model for undergraduate education in data science. Available to every undergraduate student across all schools at the university, this program of study requires experiential learning opportunities, embeds T-shaped skills, and allows students to master fundamental ADS concepts in their chosen domain area. From strong leadership engagement to funded undergraduate research opportunities, Case Western Reserve applied BHEF's Strategic Business Engagement Model to create a minor that responds to the fundamental need for data science in today's global business community.

AY 2014-15	AY 2015-16	AY 2016-17	AY 2017-18	AY 2018-19	AY 2019-20	AY 2020-21	AY 2021-22	AY 2022-23	Total
9	36	49	57	100	106	92	159	102	710

Medical Mutual of Ohio
Medtronic
Philips Healthcare
Sherwin-Williams Company
Siemens
Teradata Corporation
Timken Company
University Hospitals

2

http://www.bhef.com/publications/creating-minor-applied-data-science

BHEF BUSINESS HIGHER EDUCATION FORUM
Creating Solutions. Inspiring Action.

lle.case.edu

Components of Applied Data Science Curriculum

Applied Data Science Core Courses

- DSCI351-451: Exploratory Data Analysis
- DSCI353-453: Modeling, Prediction, Machine Learning
- DSCI352-452: Materials Data Science Res. Project
  - For their GitHub “Portfolio”
- DSCI 354-454: Data Visualization & Analytics
- DSCI332-432: Surface & Subsurface Modeling

POSEV Concepts

- Privacy, Openness, Security, Ethics, Value

Taught from “Structure of a Data Analysis” Perspective

Agile Software Development Tools & Approach

Knuth’s Literate Programming Perspective

- Integrate Code and Report Writing
- Rmarkdown, Jupyter Notebooks

Textbooks

- Open Intro Statistics
- Introduction to Statistical Learning with R

Taught using a Practicum Approach

Each class has two parts

- **Foundation:**
  - Statistics, Regression, ML, Time series ...
- **Practicum**
  - Code Style and commenting
  - Pipelines and Pipe operators
  - Data structures and data frames

Coding/Programming Language

- R with Rstudio IDE
- Python with Spyder (or Jupyter notebooks)

Open Data Science Toolchain

- (cross platform: Linux, Mac, Win)
- R, Python
- Rstudio, Spyder
- Markdown, Rmarkdown
  - Jupyter Notebooks for Python, R
- LaTeX engine, TexStudio
- Chrome, Firefox, html

Open Data Science Tool Chain

Using Open-source, Agile Tools

- Manifesto for Agile Software Development

Reproducible Research

- Using Rmarkdown reports
- Python/R Jupyter Notebooks
- When data updates
- Recompile your report
- All new figures and report!
- Well Documented Codes & Reports

High Level Scripting Languages: R, Python

- Use Machine Learning Frameworks
- Such as Keras/TensorFlow for Deep Neural Networks

Rstudio Integrated Development Environment

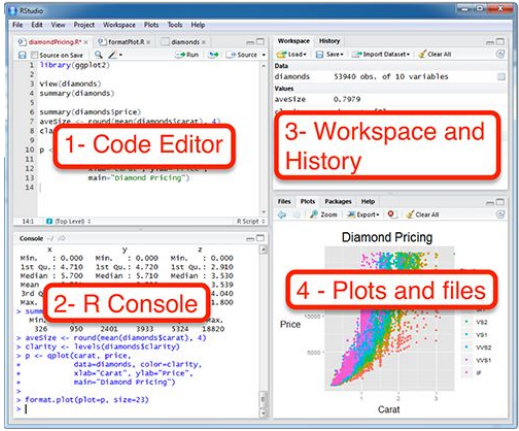
- Spyder IDE for Python

Git Repositories for Code Version Control

- Share code scripts with colleagues
- Share project data and reports with others

Github, BitBucket, GitLab for Collaboration

- Website hosting your Code Repositories



Teaching with Git & The Tools of Agile Software Development

Coursework distributed using Git Repository

- Fork the “Prof” repo
- Students tend their personal repo

Coherent Repo Structure

- Codes using relative pathing
- So codes work cross-platform

Git Sync and Pull

- For each class

Git Add, Commit, Push

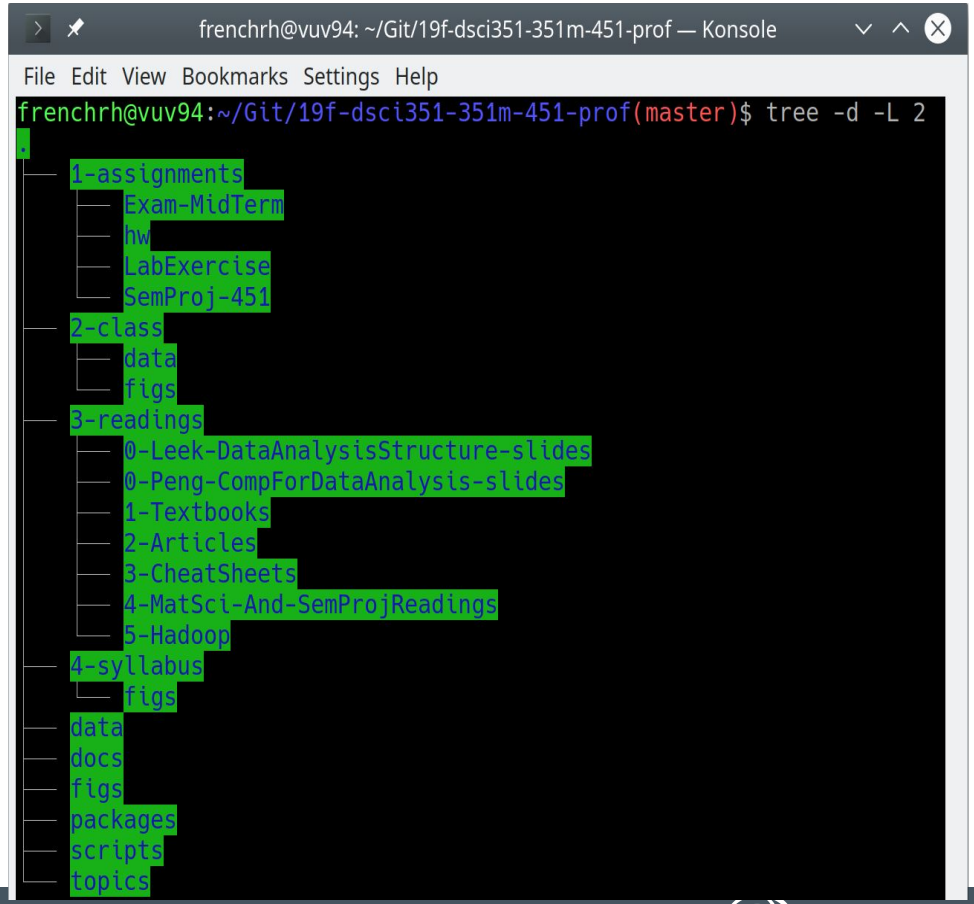
- Students own work

Class Notes in Rmarkdown

- Compiled to pdf
- With Pandoc & LaTeX

Assignments

- Traditional homeworks
- Lab Exercises: Two week assignments



“Structure of a Data Analysis” Perspective, For SemProj’s & Class Practicum

Part a) Define Question

- Background on the research area & critical issues
- Define the question
- Define the ideal data set
- Determine what data you can access
- Define critical capabilities, identify packages you will draw upon
- Obtain the data, define you target data structure
- Clean and tidy the data

Part b) Cleaning and Exploratory Data Analysis (EDA)

- Write your databook, defining variables, units and data structures
- Data visualization and exploratory data analysis
- Observations of trends and functional forms
- Power transformations
- Validate with reference to domain knowledge
- Evaluate the types of Modeling Approaches to take

Part c) Modeling, Prediction, Machine Learning

- Types of modeling to try
- Statistical prediction/modeling
- Model selection
- Cross-validation, Predictive R2
- Interpret results
- Challenge results

Part d) Present Your Final Models and Learnings

- Present your results
- Present reproducible code
- Comparison to literature modeling approaches

Jeff Leek, JHU, Data Analytic Style

Distribution of Water Molecules in a Triboelectric Charging System

Rui Fu

DSCI 451 SemProj 3: Predictors and Responses of

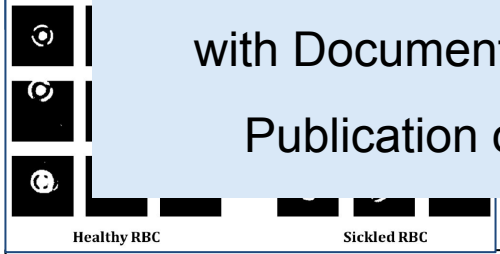
Imag ce

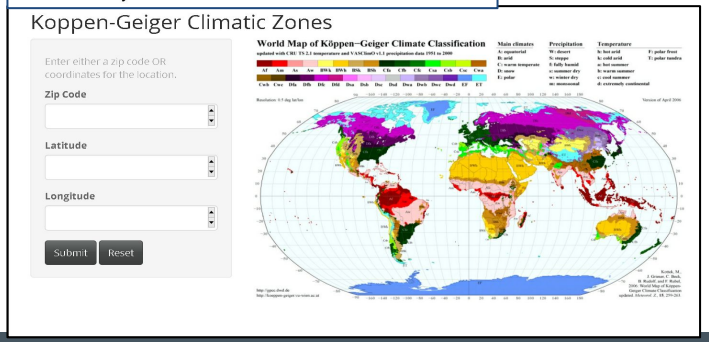
Software Packaging/Engineering

Scripts => Functions => Packages

with Documentation & Vignettes

Publication on CRAN, PyPI





kgc
R Package
C. Bryant

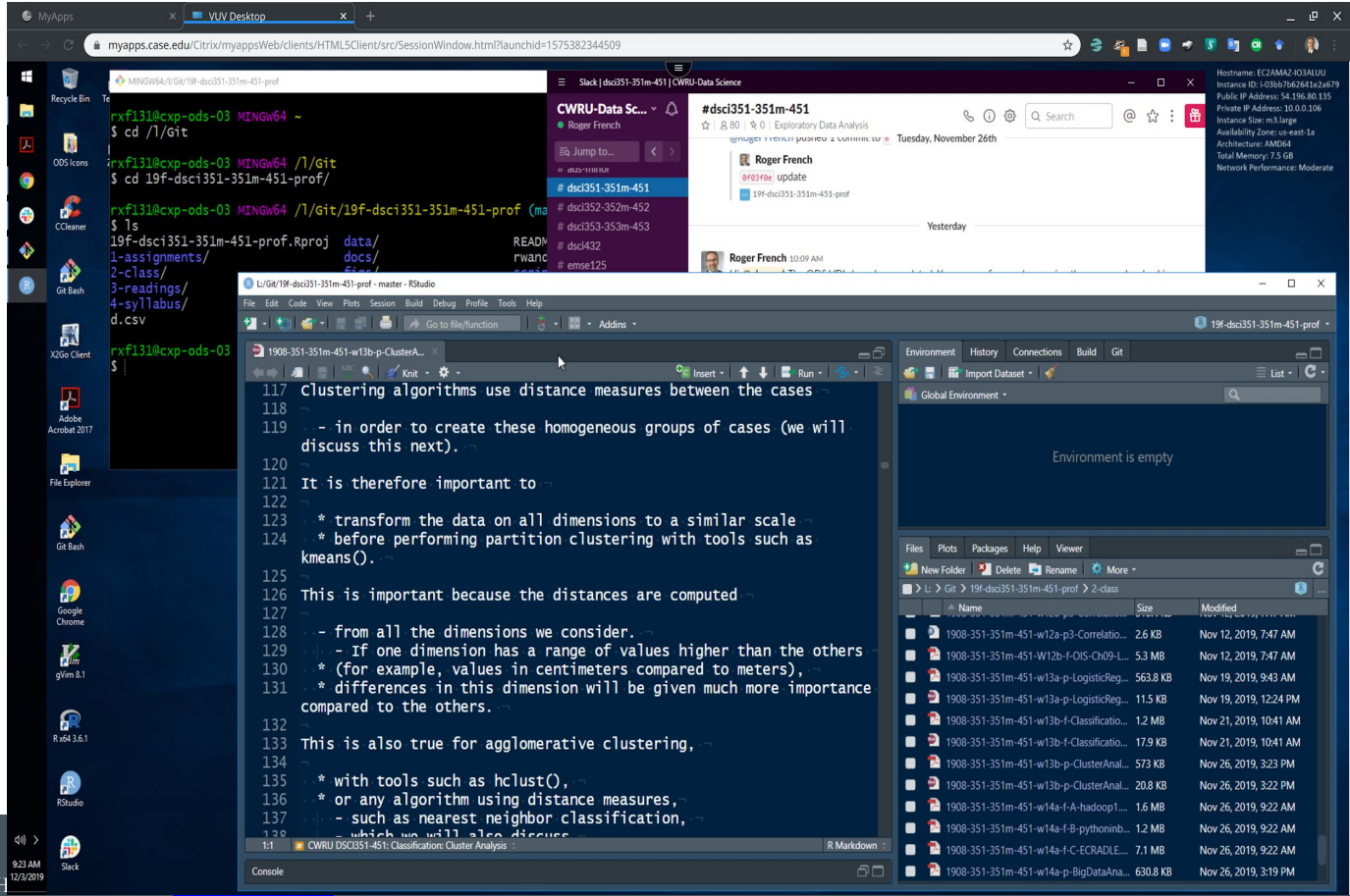
Compute Infrastructure for the ADS program:Citrix Win10 Cloud Computers

Provide students Open Data Science Computers

- Win10 Cloud Computers (Citrix)
- Hosted on CWRU AWS
  - Scalable,
  - Good Performance
    - Tohoku Univ.,
    - Sendai Japan
- With R, Rstudio,
- Python3, Spyder
  - “standard” R packages
  - “standard” Python packages
- Git, (git bash)
- Pandoc, LaTeX, html
- Slack
- StackExchange

Standard ODS Env.

- No time lost fixing computers
- Full install instruc. provided



CWRU Markov Data Science Cluster: Hosted by [U]Tech Res. Computing

Markov Total = 1120 CPU cores, 174k GPU cores

- 28 nodes: 2 Xeon CPUs (20 cores/CPU).
- 20 nodes: 2 Xeon CPUs with 2 Nvidia RTX2080Ti
- 1120 CPU cores and 174K GPU cores.

Enables Batch & Interactive GUI sessions

- Using either compute or GPU nodes.

Running R/Rstudio and Python3/PyCharm,

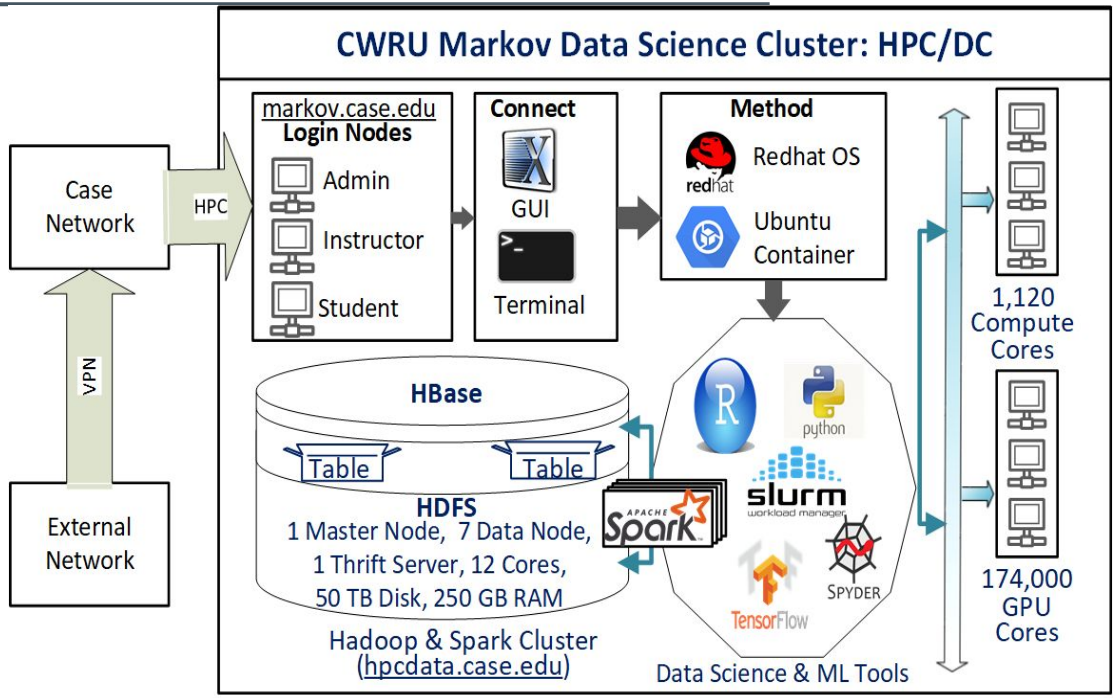
- Along with Keras/TensorFlow2, Torch, Cuda/CDNN

Implemented using the Open Data Science (ODS) Kubuntu Container

- Based on Singularity.

Markov's Hadoop cluster: hpcdata.case.edu

- Loaded with publically available datasets



OnDemand[1] Web Portal for HPC Access

Using Singularity for Containers

And Ubuntu 20.04lts OS

- **Instead of the HPC RHEL 7.9**
- **Better support for new ML tools**

Easy to develop new Containerized Apps

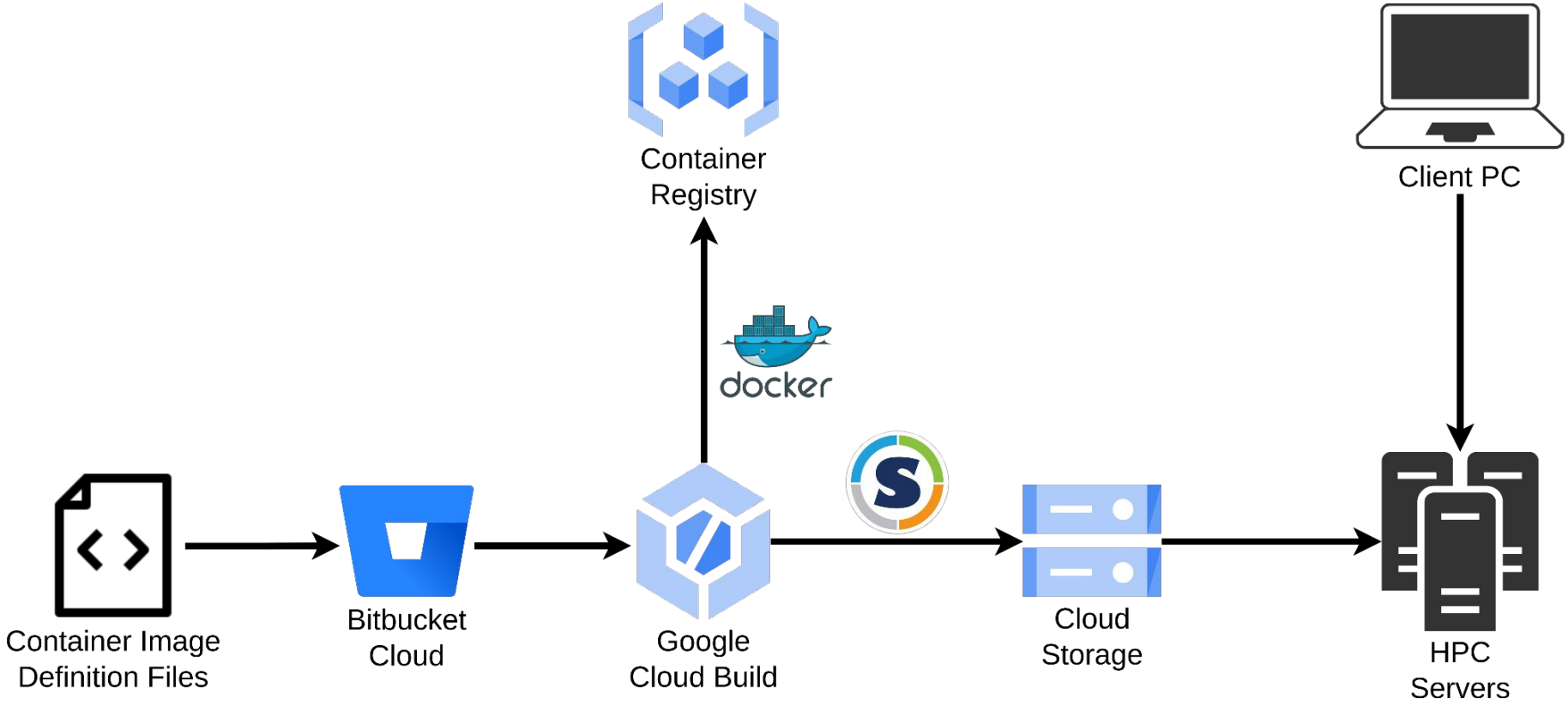
- **Rstudio**
- **PyCharm**
- **Jupyter Labs**
- **LXDE GUI Desktop**

Manage R & Python versions & dependencies

- **Versions**
  - R 4.2.1
  - Python 3.8.10
- **And all package versions**

[1] D. Hudak et al., "Open OnDemand: A web-based client portal for HPC centers," JOSS, vol. 3, no. 25, p. 622, May 2018, doi: 10.21105/joss.00622.

Our Data Science Containers Build to our Google Cloud Container Registry

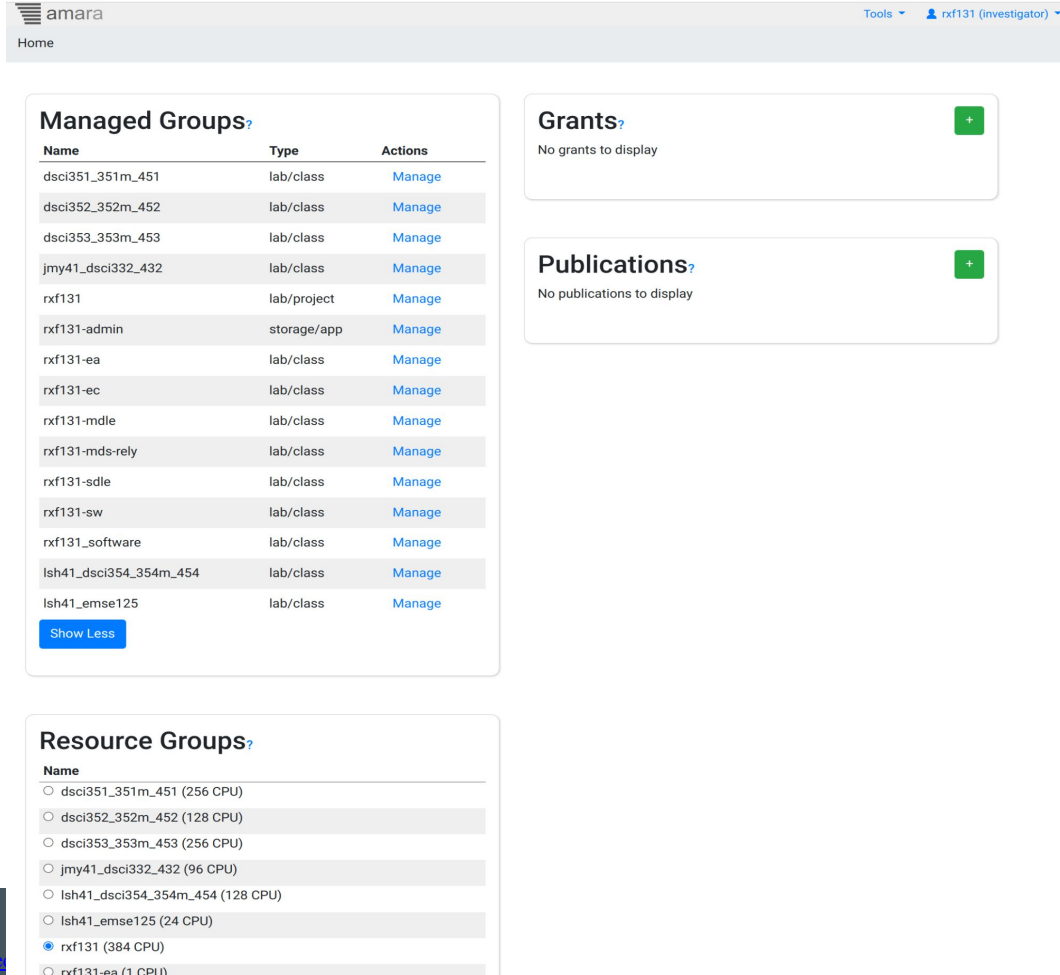


Amara to Manage Linux Groups for HPC, Apps & File Access

Make group permissions management accessible

Enable more complex Cyber security groups

- **To address the security in research**
- **And education**



Common Research Analytics & Data Lifecycle Environment

CRADLE Analytics & Compute Cluster

Distributed & High Performance Computing[1]

FAIRification of Datasets

1. A. Khalilnejad, et al., Automated Pipeline Framework for Processing of Large-Scale Building Energy Time Series Data, PLOS ONE. 15 (2020) e0240461. https://doi.org/10.1371/journal.pone.0240461.

Data Processing Framework: CRADLE

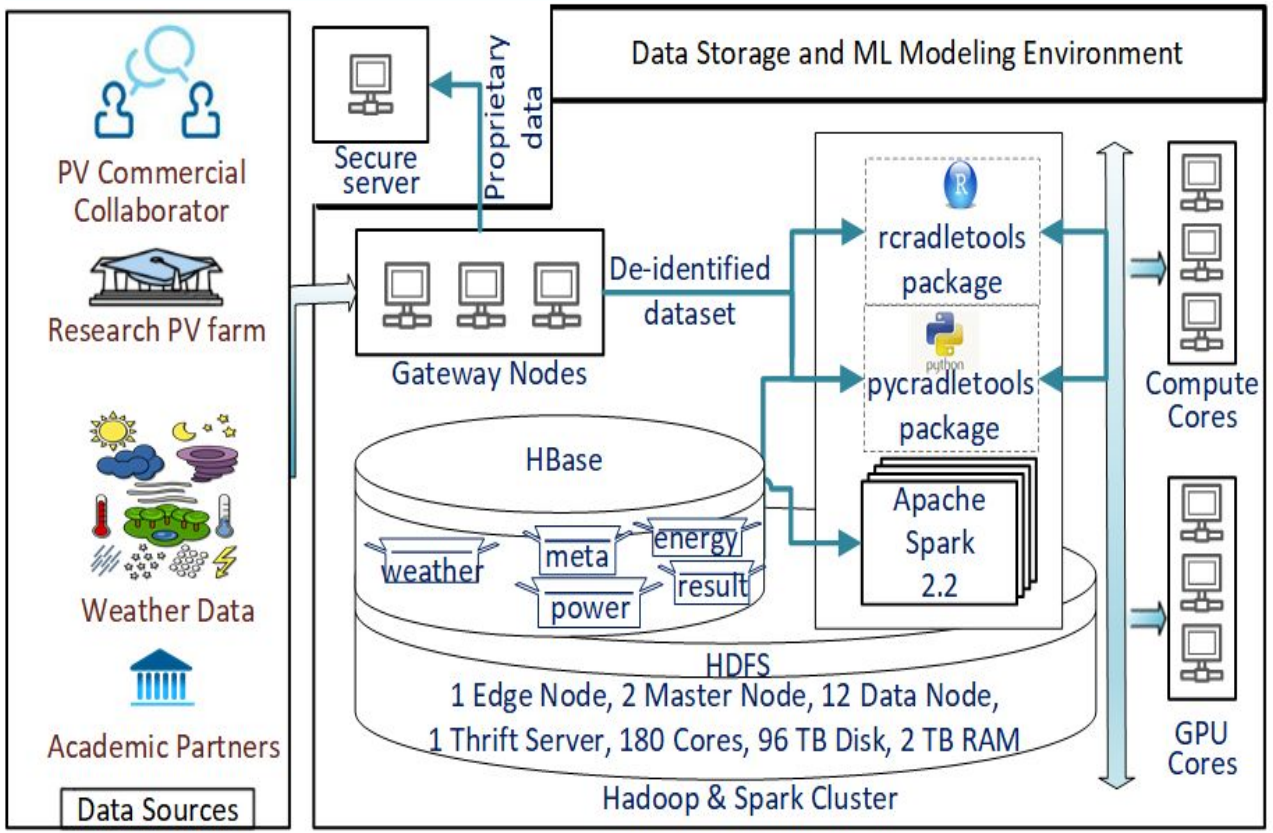
Data acquisition
- Diverse sources
- Anonymization
- Pre-processing
- Metadata

NoSQL Database system
- HBase

Computation
- Data Ingestion & Analysis
- Impute missing values
- R & Python3
- Tensorflow2
- Torch & Pytorch

CRADLE tools
- R & Python package
- Containerized Applications

Common Research Analytics & Data Processing Environment



CRADLE Compute Environment

Running in CWRU HPC
- **Rider (RHEL7)**
- **Pioneer (RHEL8 OS)**
- **Markov (Teaching Cluster)**

OnDemand Containerized Apps
- Using Ubuntu 20.04 OS

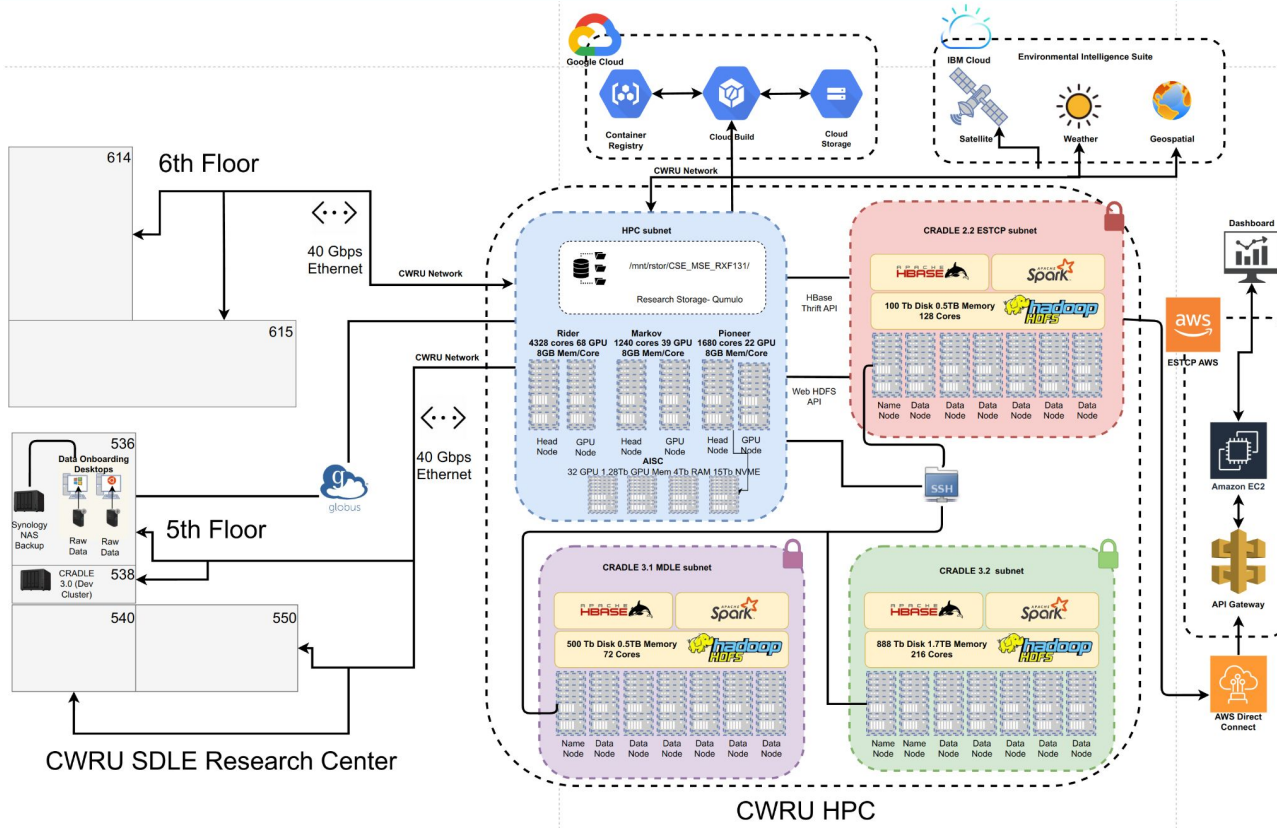
Cloudera Data Platform
- Comm. supported distribution
- Of Apache Hadoop/Hbase/Spark/….

2 Petabytes of Distrib. Comp. Storage
- **Meeting NIST 800-171 level 1**
- **For Controlled Unclassified Data**

Nvidia AISC
- 32 A100 GPU (40 Gb vRAM each)
  - = 1.28 Tb of GPU vRAM
- With 15 Tb of NVME Fast Storage

Able to train 100s of Deep Learning Models

Common Research Analytics & Data Processing Environment



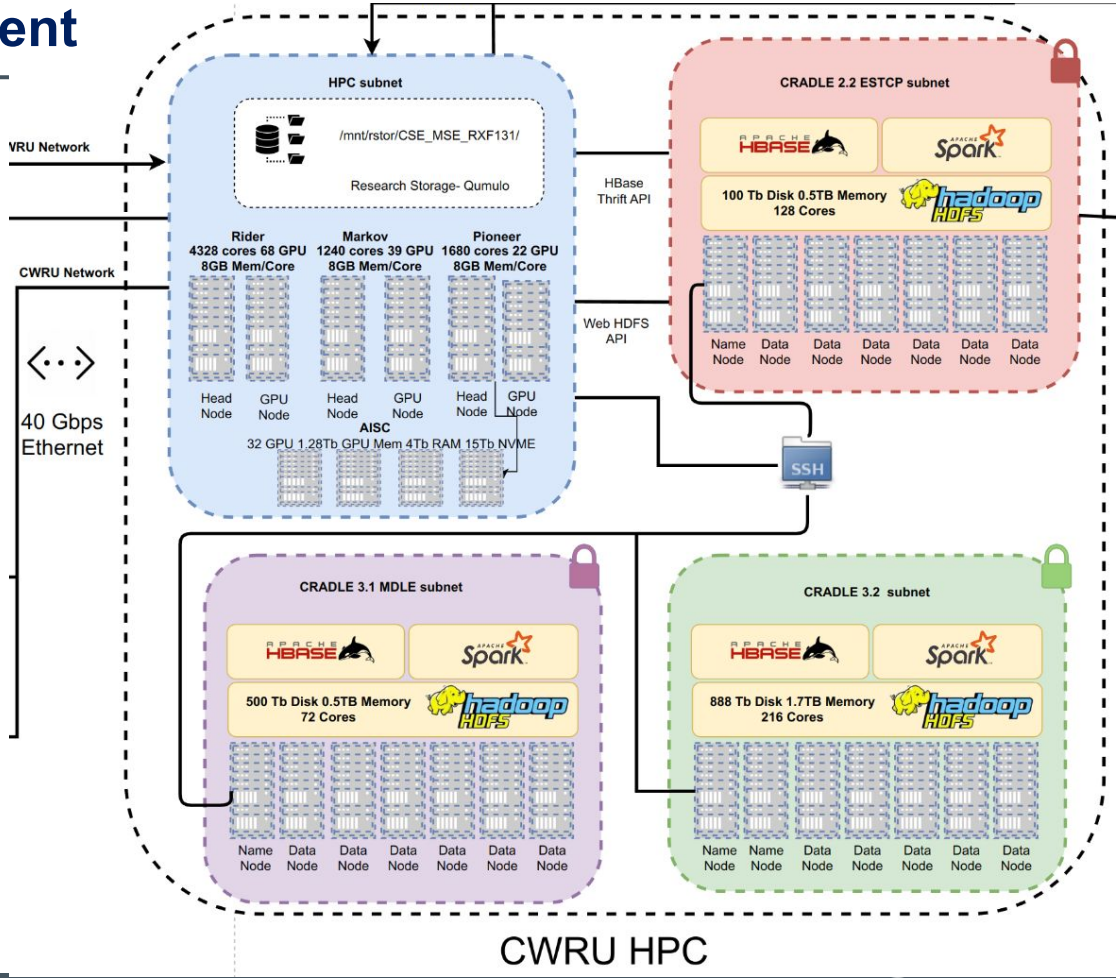
CRADLE Analytics Environment

CRADLE 2.3
- 128 Cores
- 0.5 Tb RAM
- 100 Tb Storage
- CDH 5.16.2, SP-800-171

CRADLE 3.1
- 72 Cores
- 576 Gb RAM
- **480 Tb Storage**
- CDP 7.1, SP-800-171

CRADLE 3.2
- 216 Cores
- 1.8 Tb RAM
- **1.5 Pb Storage**
- CDP 7.1, SP-800-171

NVIDIA GPU AISC
- 32 DGX A100 GPUs
  - 40 Gb RAM each
- **1.28 TB GPU RAM**
- **15 Tb NVME storage**



CRADLE Analytics Environment

CRADLE 2.3
- 128 Cores
- 0.5 Tb RAM
- 100 Tb Storage
- CDH 5.16.2, SP-800-17

CRADLE 3.1
- 72 Cores
- 576 Gb RAM
- **480 Tb Storage**
- CDP 7.1, SP-800-171

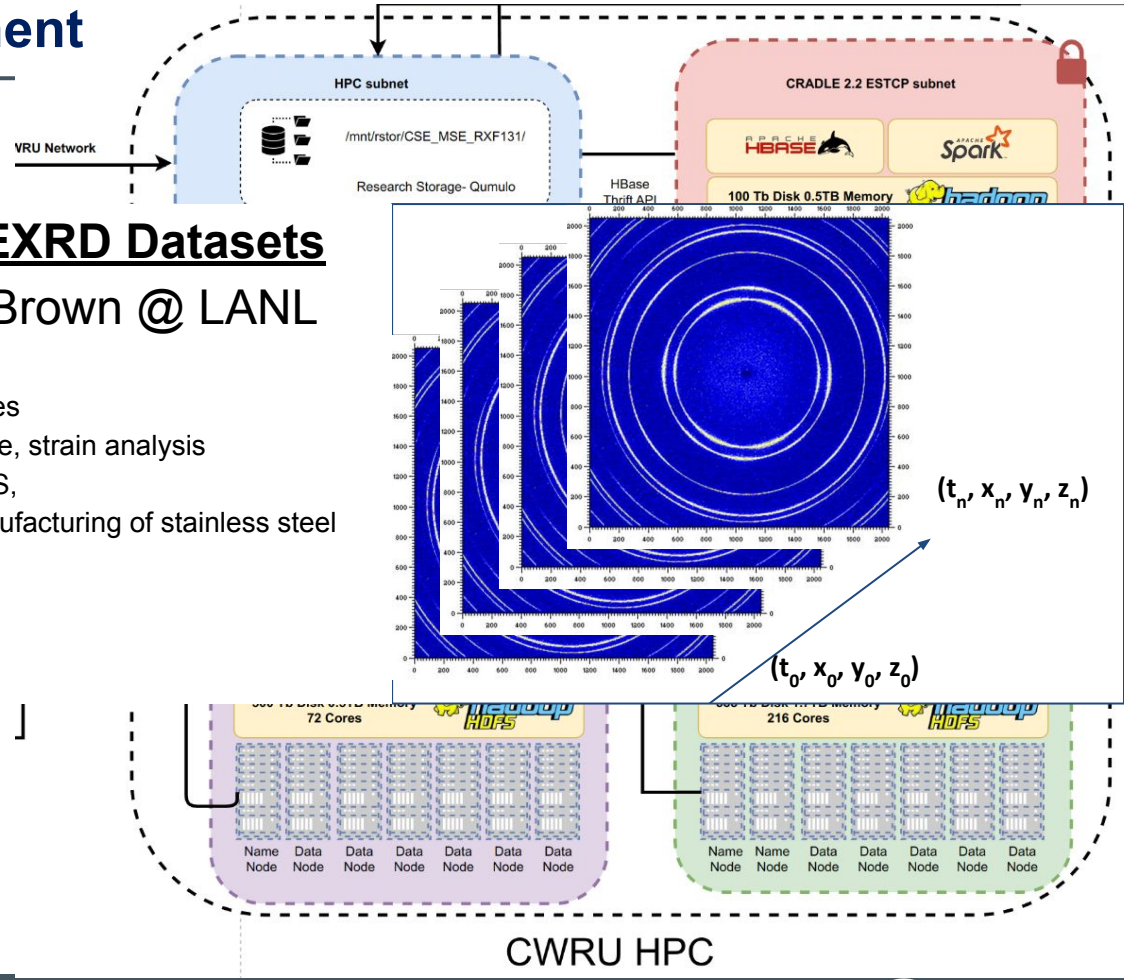
CRADLE 3.2
- 216 Cores
- 1.8 Tb RAM
- **1.5 Pb Storage**
- CDP 7.1, SP-800-171

NVIDIA GPU AISC
- 32 DGX A100 GPUs
  - 40 Gb RAM each
- **1.28 TB GPU RAM**
- **15 Tb NVME storage**

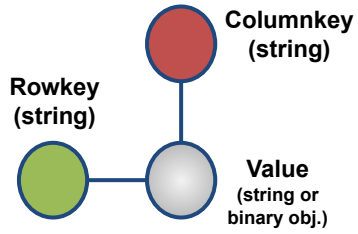
Current 2D-HEXRD Datasets

from Don Brown @ LANL

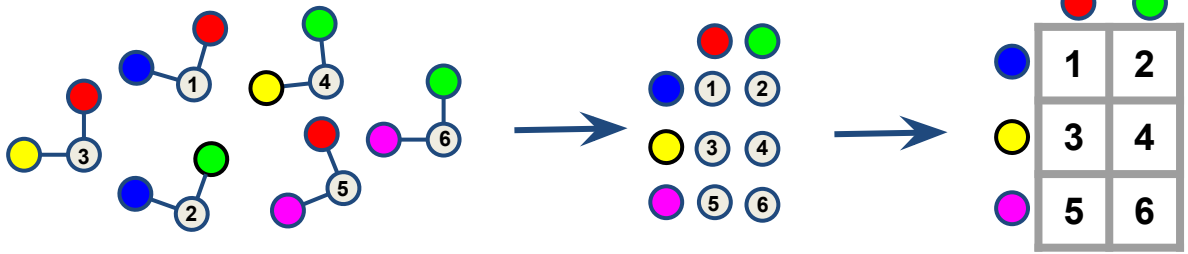
- ~ 13 Tb
- ~ 3 million XRD images
- In-situ heating, texture, strain analysis of Ti-6Al-4V at CHESS,
- Wire arc additive manufacturing of stainless steel etc.



The “NoSQL” Database Abstraction of Hadoop/Hbase: RDF Triples



HBase RDF ‘Triple’



Combines Lab data (Spectra, Images, Videos etc.)

With Geospatiotemporal Data (PV Power Plant Data)

Distributed & High Performance Computing:

Petabyte Data Lake In A Petaflop HPC Environment

- In-place Analytics: Distributed Spark Analytics in Hadoop/HDFS/Hbase
- In-memory Data Extraction: To Separate HPC Compute Nodes

A non-relational data warehouse for the analysis of field and laboratory data from multiple heterogeneous photovoltaic test sites

Yang Hu, *Member, IEEE,* Venkat Yashwanth Gunapati, Pei Zhao, Devin Gordon, Nicholas R. Wheeler, Mohammad A. Hossain, *Member, IEEE,* Timothy J. Peshek, *Member, IEEE,* Laura S. Bruckman, Guo-Qiang Zhang, *Member, IEEE,* and Roger H. French, *Member, IEEE*

Automated pipeline framework for processing of large-scale building energy time series data

Arash Khalilnejad[1,5], Ahmad M. Karimi[2,5], Shreyas Kamath[1,5], Rojiar Haddadian[2,5], Roger H. French[2,4,5]*, Alexis R. Abramson[3,6¤]

Compute and data

HPC Cluster

Rider

- 4328 cores
- 68 GPU
- 8 Gb Mem/core

Markov

- 1240 cores
- 39 GPU
- 8 Gb Mem/core

Pioneer

- 1680 cores
- 22 GPU
- 8 Gb Mem/core

Hadoop Cluster

CRADLE 2.1

- 75 TB
- 192 cores
- 1TB memory
- 15 nodes

CRADLE 2.2

- 100 TB
- 128 cores
- 500GB memory
- 4 nodes

CRADLE 3.1

- 500 TB
- 72 cores
- 564GB memory
- 3 nodes

Data

Temporal data:

- ~2000 PV power plant data
- ~1000 Buildings energy consumption
- ~1000 Locations with weather data

Geospatial data:

- ~1200 Well data
- ~1000 Locations with subsurface temperature

XCT data: 50 TB

- Diamond Light Source
- LLNL

XRD data: ~12TB

Singularity containers

Portable and reproducible containers

- **Cybersecure**
- **Admin privileges not required**
  - Different from Docker Containers

Easy to resolve software dependency conflict

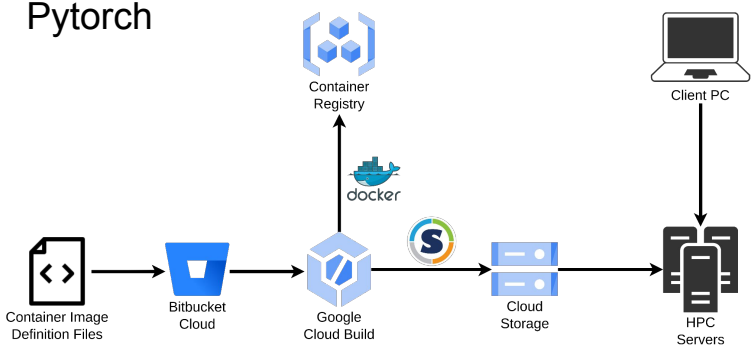
- NVIDIA CUDA
- Python version

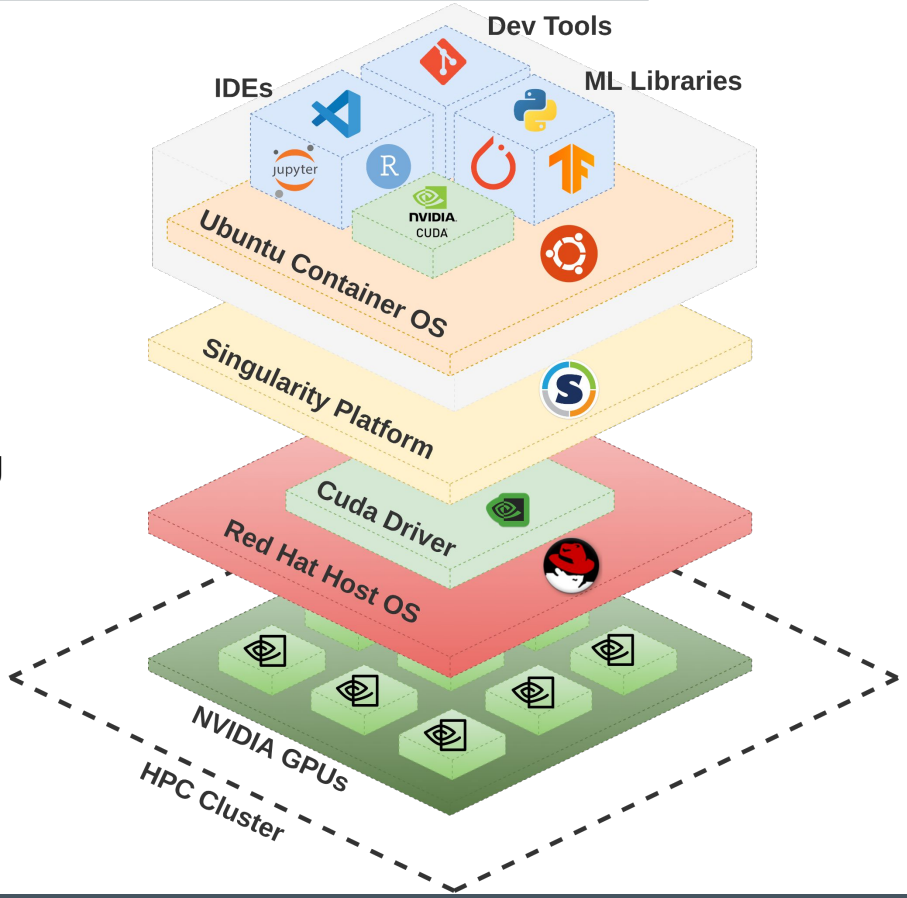
Ubuntu 20.04lts Based container

- Supports all major data science packages

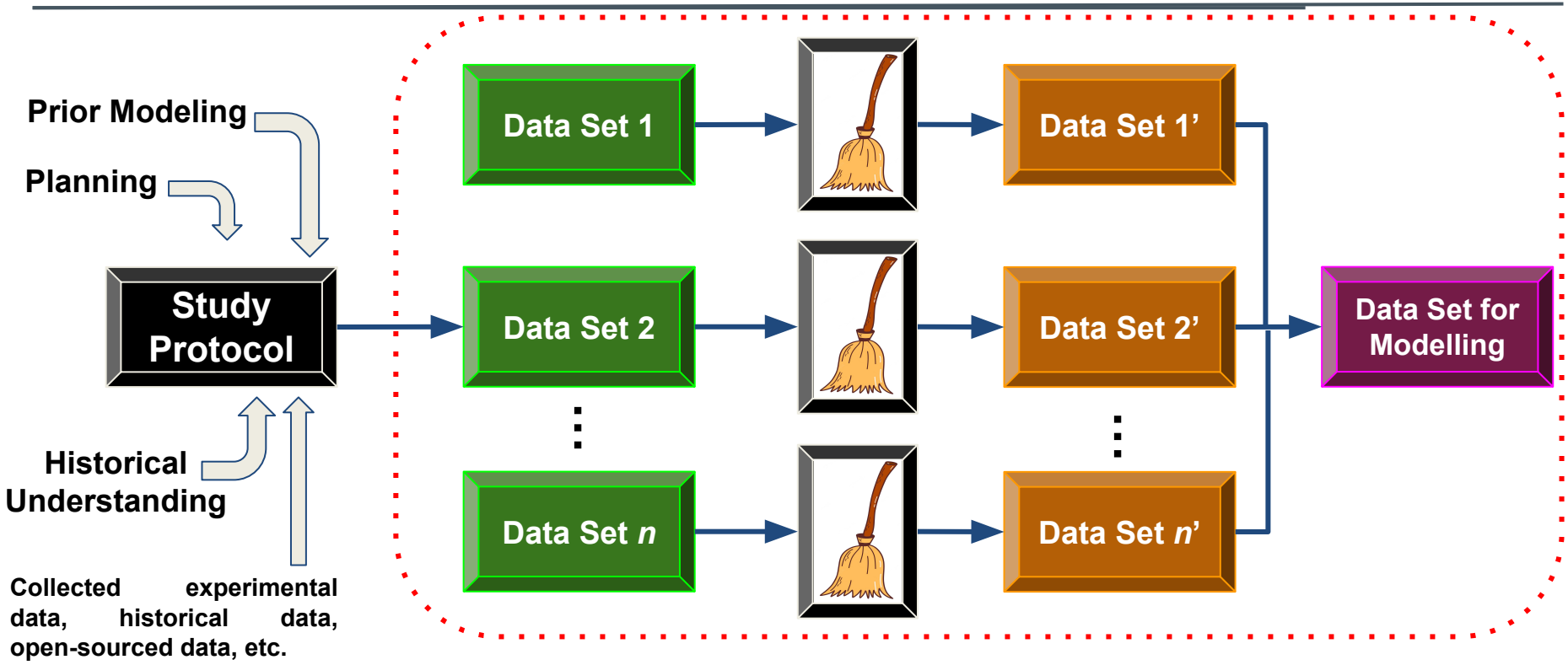
Enables multi-GPI Deep neural network training

- Tensorflow
- Pytorch



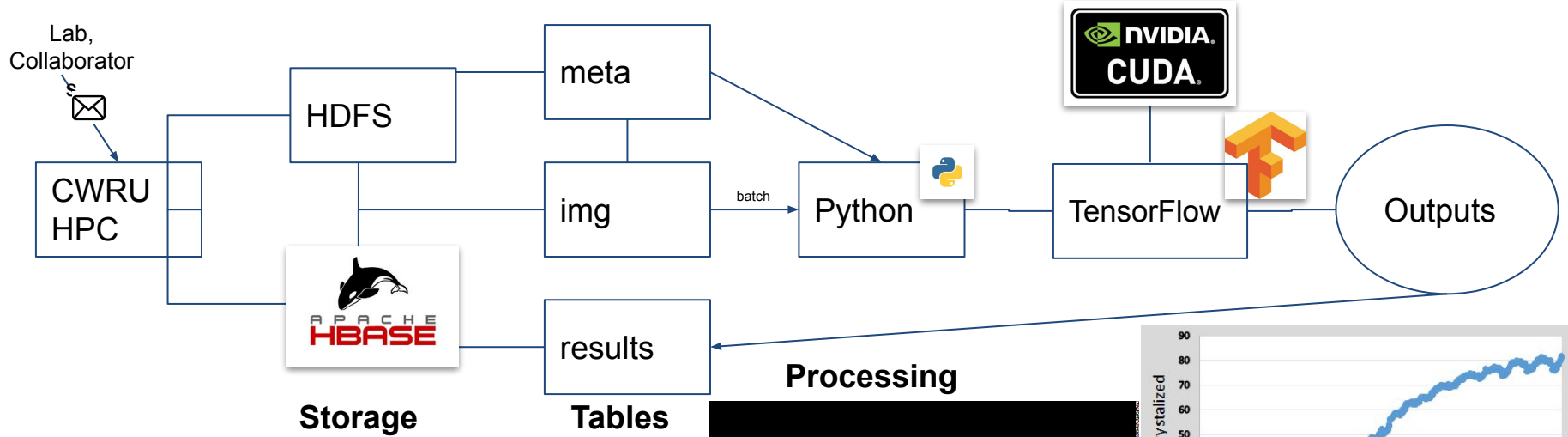


Data Science Pipelining



Poor data and experimental planning can have embarrassing/catastrophic outcomes: [1], [2]

Data Processing Infrastructure: A Data Analysis Pipeline (Python or R)



CRADLE infrastructure

NoSQL database

- Apache HBase

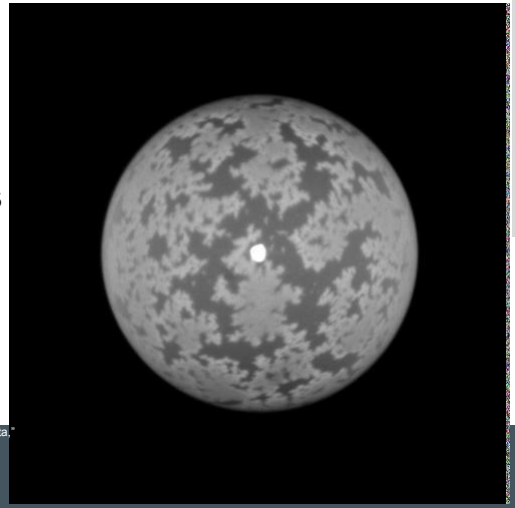
HPC environment

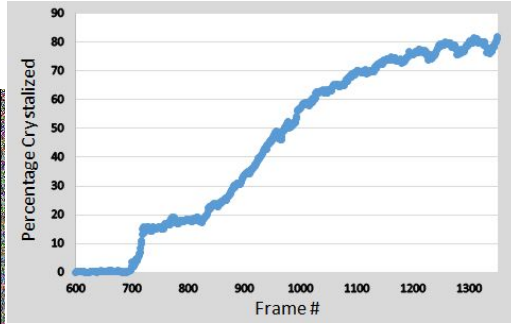
- Nvidia GPU acceleration for deep learning

Python/TensorFlow language

Nucleation & Growth of AlN Crystals

- From Al/Ni Melt
- 1 Million Images





M. Adachi, S. Hamaya, D. Morikawa, B. G. Pierce, A. M. Karimi, Y. Yamagata, K. Tsuda, R. H. French, H Fukuyama, Temperature dependence of crystal growth behavior of AlN on Ni–Al using electromagnetic levitation and computer vision technique", Mat. Sci. in Semicon. Proc., 153, 2023, 107167, ISSN 1369-8001, https://doi.org/10.1016/j.mssp.2022.107167

FAIRification of Datasets & Models, Enables AI learning

Making Datasets & Models FAIR

- By "FAIRification"

Enables Models to find Data

- And Data to find Models

So that they can advance

- Without human intervention

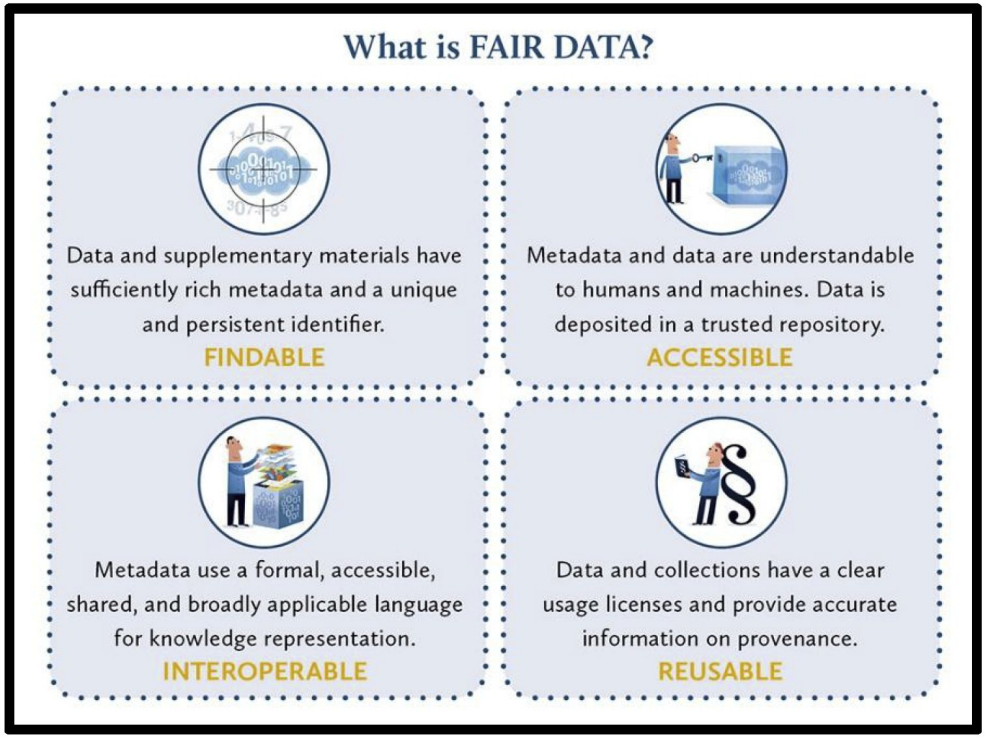
This is an aspect of the Semantic Web

- And Resource Description Framework
- Hbase triples are an example of RDF

FAIRification essential to
DOE SETO AI awards

- For st-GNN, that includes FAIRification
- And PV Multiscale

DOE-NNSA: Materials Degr. & Life Ext.
IEA-PVPS Task 13



M. D. Wilkinson et al., "The FAIR Guiding Principles for scientific data management and stewardship," Sci. Data, vol. 3, no. 1, pp. 1–9, Mar. 2016, doi: 10.1038/sdata.2016.18.
https://en.wikipedia.org/wiki/Resource_Description_Framework

FAIRification of Datasets and Models, Enables AI learning

Making Datasets & Models FAIR

- By “FAIRification”

Enables Models to find Data

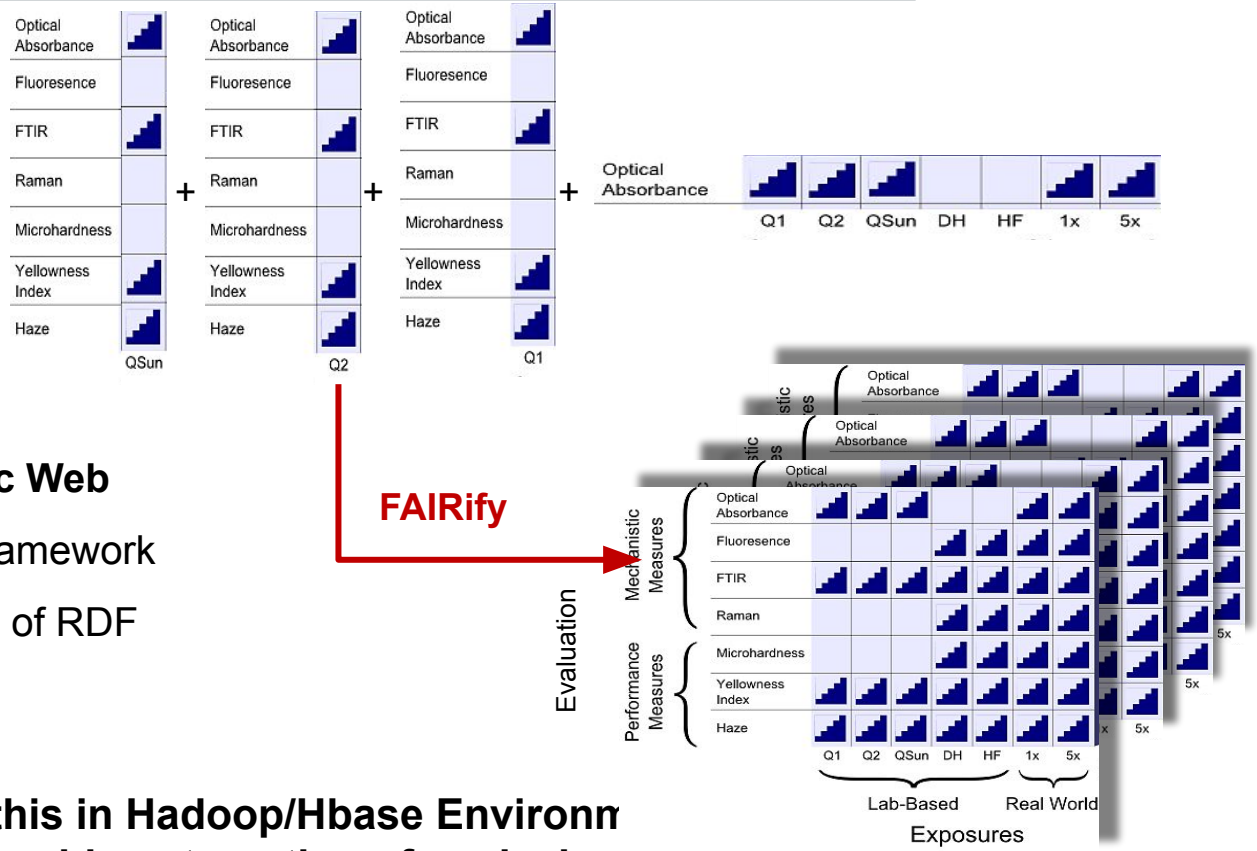
- And Data to find Models

So that they can advance

- Without human intervention

This is an aspect of the Semantic Web

- And Resource Description Framework
- Hbase triples are an example of RDF



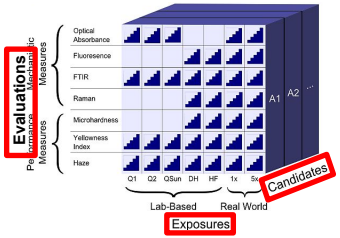
Enabling this in Hadoop/Hbase Environm

- **Can enable automation of analysis**

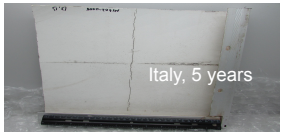
M. D. Wilkinson et al., “The FAIR Guiding Principles for scientific data management and stewardship,” Sci. Data, vol. 3, no. 1, pp. 1–9, Mar. 2016, doi: 10.1038/sdata.2016.18.
https://en.wikipedia.org/wiki/Resource_Description_Framework

Lifetime & Degradation Science Framework and Thrusts

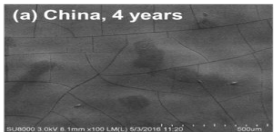
Exposures & Evaluations of Fielded Materials & Components

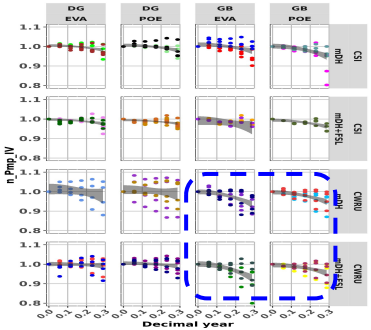


Forensic Studies on Field-Retrieved Components



PV Module Cracked Backsheets





Lifetime Performance of eight PV Mini-module Variants

Lab-based Accelerated Studies

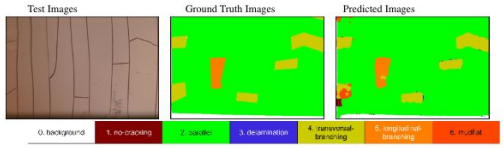
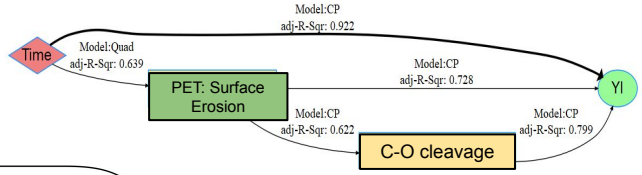
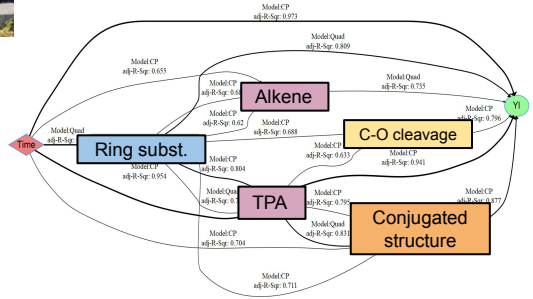


Figure 8. Six examples of crack inspection task performed on the test images using the trained Model O. The different colors in the (b) and (c) column images indicated different crack classes shown in the color bar.

CNN Classification of Accelerated Exposed Backsheet Cracks

Network models: <*S*|*M*|*R*> Mechanistic Degradation Pathway





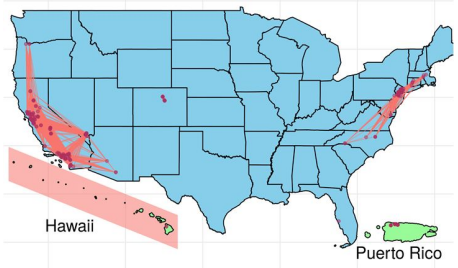
PET Degradation

Under accelerated exposures with water spray (top) and humidity (bottom)

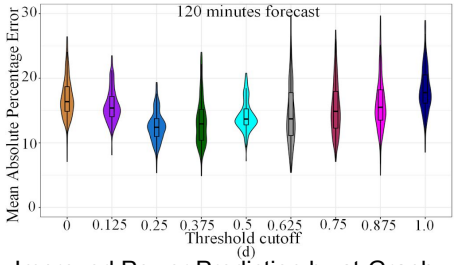
Chemical degradation mechanisms Identified along pathways

Surface erosion with spray.

Deep Learning with spatiotemporal-Graph Modeling



st-Graph Neural Network Models



Improved Power Prediction by st-Graph Neural Network Model

Lyu, Y., et. al. *Solar Energy*, 199, 425-436. Zhang, B., et al. (2019). *Scientific reports*, 9(1), 1-13. J. Liu *et al.*, *2020 47th IEEE Photovoltaic Specialists Conference (PVSC)*, Calgary, OR, 2020, pp. 1709-1714. A. Karimi, et al., *Proc. IAAI-21*, 2021. Yang, H. E., French, R., & Bruckman, L. (Eds.). (2019). *Durability and Reliability of Polymers and Other Materials in Photovoltaic Modules*. William Andrew.

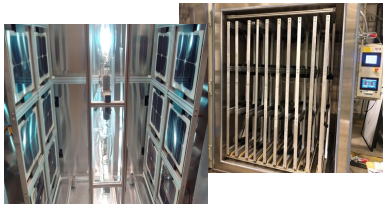
Service Lifetime Prediction (SLP): a PV example

Accurate PV SLP, is crucial to LCOE, and is the basis of PV lifetime performance

- Challenge: **PV modules are complex systems** with **multiple degradation mechanisms**

Requires:comprehensive study protocol development and data on variants and exposures

Our solution: mapping of degradation mechanisms and pathways with network modeling



Service Life Prediction:
In Lab ⬆️ & Field ⬇️

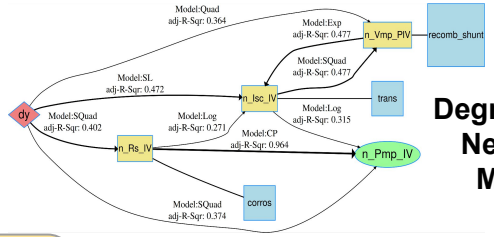
PV module variants :
BOM
Quality
Design

Weathering exposure stressors:
Climate
Racking

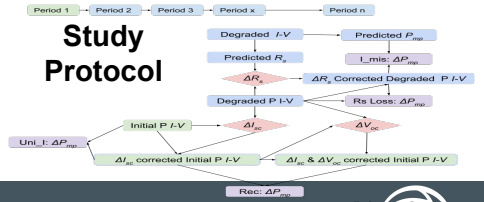
Cell technology
Encapsulation
Glass/Backsheet
Brand/Suppliers
Irradiance/UV
Moisture
Temperature

Degradation Network models

ΔR_s, corrosion
ΔI_{sc}, optical
ΔV_{mp}, *recombination*



Degradation Network Model



Study Protocol

Stream Processing Example: Tesla, ML for "Autonomous Driving"

Automated Data Analysis Pipelines

- Enable Terabyte Dataset Analysis
  - In-situ Manu. Datastreams (7 Tb)
  - Beamline XRD (12 Tb)

Write-back All Models & Results

- Future Analysis Builds On Priors
- Datasets & AI/ML Models Get Smarter

Minimize Large Data Transfer

- Prefer In-place Analytics (Hadoop/Spark)

Focus on Fast/Efficient Modeling

- Such as YOLO CNN
  - for Autonomous Driving

Knowledge Graphs

Spatiotemporal Graphs

And Their Role in

Deep Representation Learning

Knowledge graphs: Nodes & Edges Define Relationships

What is a knowledge Graph?

- **Nodes:** entities w. types & attributes;
- **Edges:** relations
- capture (factual) knowledge as graphs

Where do KGs come from?

- Structured data: sensors, tables, Wiki infoboxes,
  databases, social nets, …
- Unstructured data: text, images, videos

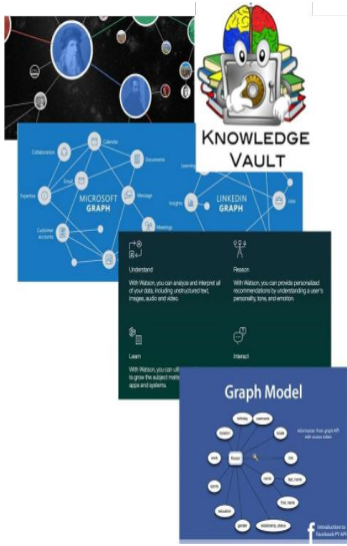
Why (Knowledge) Graphs?

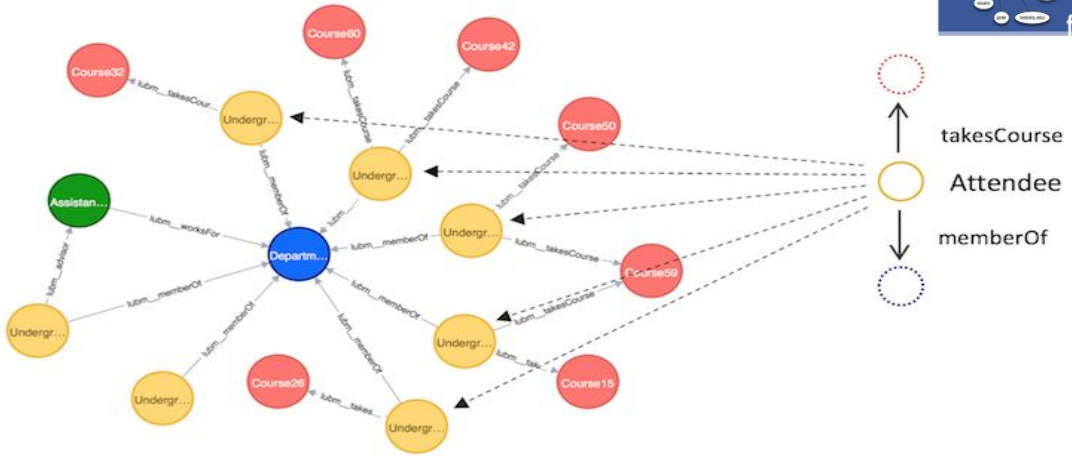
Humans:

- Explore data via intuitive/processible structure
- Combat information overload
- Tool for supporting knowledge-driven tasks

Also:

- Key ingredient for many AI tasks
- Bridge from data to human semantics
- Use decades of work on graph analysis



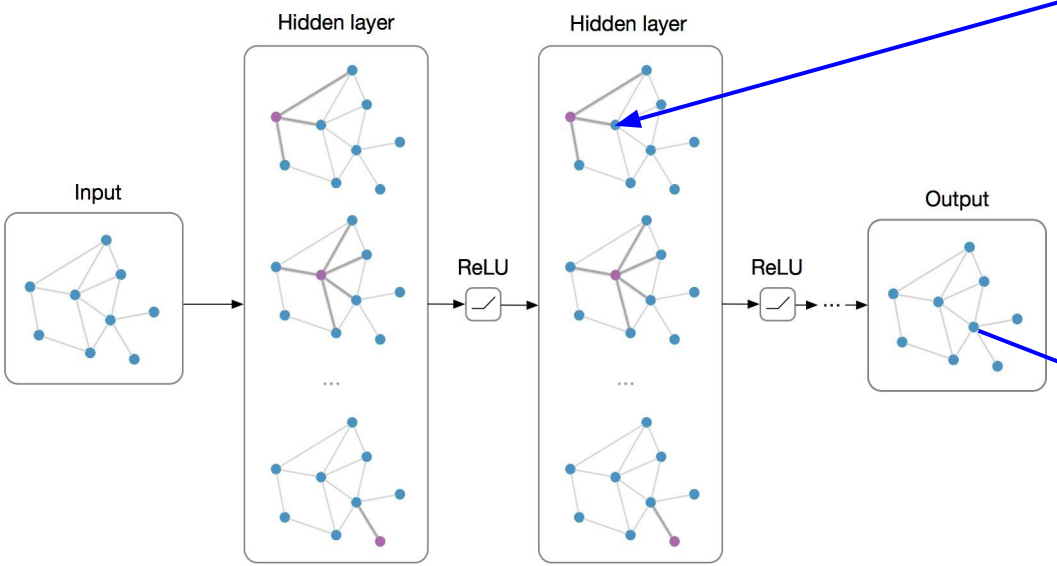


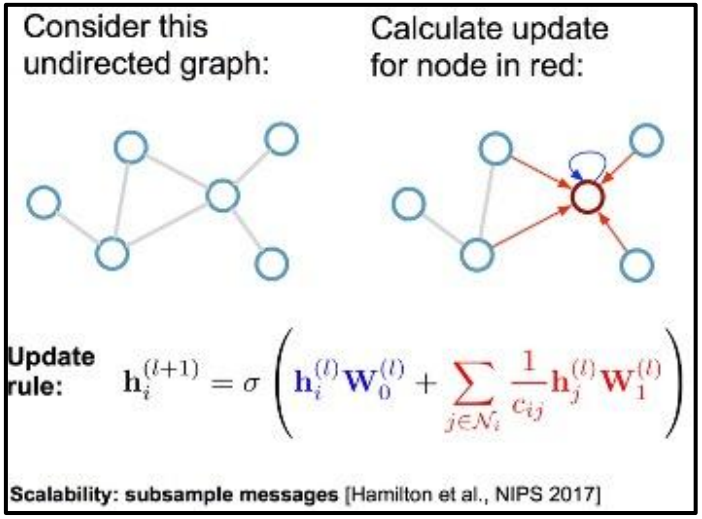
Graph Representation Learning: Enable Deep Learning on Graphs

Graph Neural Networks (GNNs)

Notation: $\mathcal{G} = (\mathbf{A}, \mathbf{X})$

- Adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$
- Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$





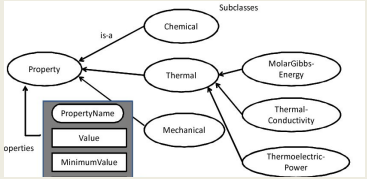
Downstream tasks:

- Node classification, link prediction…
- Graph classification, graph clustering…
- Anomaly detection, data imputation

Idea: Pass messages between nodes and agglomerate to refine node/edge representations

MDS^3-COE: A Knowledge Graph Learning Framework

Metadata, Ontologies
RDF triples, JSON
(abstraction/semantics/constraints)



... to data standardization & knowledge sharing

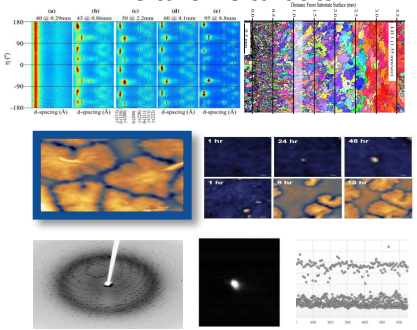
1 *validate* / *curate*

FAIRification

Featurization

2

Raw datasets
Image/videos, design data, in-situ/ex-situ data..



ML/AI models

physical con-straints

linked entities

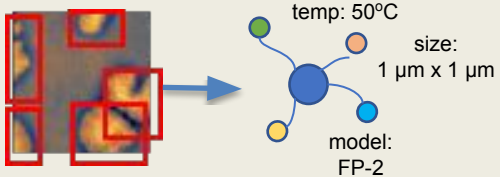
enriched features

Deep Learning

Summarization

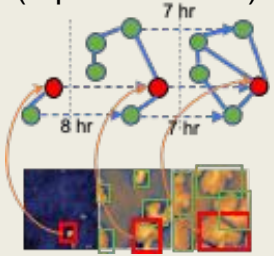
4 *cost-effective learning* / *efficient access & interpretation* **3**

Objects, Observations & Properties
(instances)



... to create AI/ML ready data resources

Scenes or st-Graphs
(representations)



Deep st-graph representation learning
... to inferential & predictive models

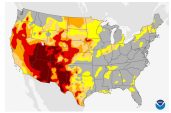
Summary Graphs
(patterns)

... to cost-effective data access, analysis & interactive exploration

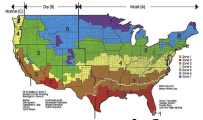
[1] Roger H. French et al., "Fairmaterials." The Python Package Index (PyPI), Oct. 08, 2021, https://pypi.org/project/fairmaterials/ .
[2] Willam C. Oltjen, et al., "FAIRmaterials: Make Materials Data FAIR." CRAN, Sep. 14, 2021, https://CRAN.R-project.org/package=FAIRmaterials .
[3] A. M. Karimi, et al., "Spatiotemporal Graph Neural Network for Performance Prediction of Photovoltaic Power Systems," in Proc. of AAAI., 2021, https://ojs.aaai.org/index.php/AAAI/article/view/17799

Degradation Science with Spatiotemporal-Graph Models

image/satellite



climate/environment

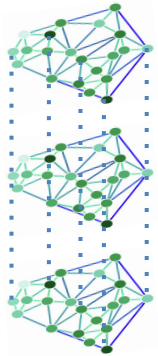


geospatial

featurizer

st-mapper

featurized network



time

querying/pattern mining

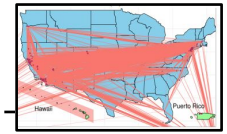
Metadata

Field & Lab Data

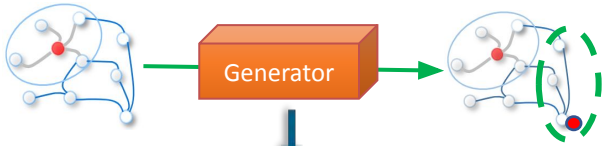
Manufacturing Data

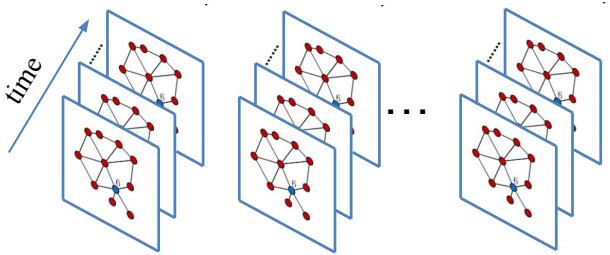
Material data storage: CRADLE

Domain/physical models & constraints



st-data augmentation





Graph Representation Learning:
message passing with spatiotemporal attention in graph convolution layers, incorporating non-local spatiotemporal dependencies

transform embeddings to downstream analysis

Classification

Clustering

Prediction/ forecasting

Decision making

Yinghui Wu, CWRU

[1] Chelsey Bryant, Nicholas R. Wheeler, Franz Rubel, and Roger H. French, “kgc: Koeppen-Geiger Climatic Zones.” The Comprehensive R Archive Network, Dec. 20, 2017 [Online]. Available: https://cran.r-project.org/package=kgc . [Accessed: May 24, 2021]
[2] W.-H. Huang et al., “netSEM: Network Structural Equation Modeling.” The Comprehensive R Archive Network, Nov. 28, 2018 [Online]. Available: https://CRAN.R-project.org/package=netSEM . [Accessed: Dec. 02, 2018]
[3] A. M. Karimi, B. G. Pierce, J. S. Fada, N. A. Parrilla, R. H. French, and J. L. Braid, “PVimage: Package for PV Image Analysis and Machine Learning Modeling.” May 08, 2020 [Online]. Available: https://pypi.org/project/pvimage/ . [Accessed: Feb. 28, 2020]
[4] Alan J. Curran, Tyler Burleyson, Sascha Lindig, David Moser, and Roger H. French, “PVplr: Performance Loss Rate Analysis Pipeline.” The Comprehensive R Archive Network, Oct. 07, 2020 [Online]. Available: https://CRAN.R-project.org/package=PVplr . [Accessed: Oct. 18, 2020]
[5] Wei-Heng Huang et al., “ddiv: Data Driven I-V Feature Extraction.” The Comprehensive R Archive Network, Apr. 14, 2021 [Online]. Available: https://CRAN.R-project.org/package=ddiv . [Accessed: Jul. 30, 2019]
[6] Menghong Wang et al., “SunsVoc: Constructing Suns-Voc from Outdoor Time-series I-V Curves.” The Comprehensive R Archive Network, Apr. 30, 2021 [Online]. Available: https://CRAN.R-project.org/package=SunsVoc
[7] Willam C. Oltjen, Liangyi Huang, Roger H. French, and Liangyi Huang, “FAIRmaterials: Make Materials Data FAIR.” The Comprehensive R Archive Network, Sep. 14, 2021 [Online]. Available: https://CRAN.R-project.org/package=FAIRmaterials
[8] Roger H. French et al., “Fairmaterials.” The Python Package Index (PyPI), Oct. 08, 2021 [Online]. Available: https://pypi.org/project/fairmaterials/
[9] Kris Zhao and Roger H. French, “hbspark: Package to pipe data from HBase to Spark (2+).” 2021 [Online]. Available: https://github.com/kxz167/hbspark . [Accessed: 29-Jan-2022]
[10] R. F.-0002-6162-0532) Huang(ORCID:0000-0003-0845-3293) Liangyi, “pointextract: Extract points information from 2d images to build a 3D object.” Jun-2022 [Online]. Available: https://pypi.org/project/pointextract/ . [Accessed: 22-Apr-2022]

FAIRification: make raw datasets FAIR

Project datasets FAIRified
to key-value and triple representa

- non-image data, metadata & ontolog
- standardized to materials KGs

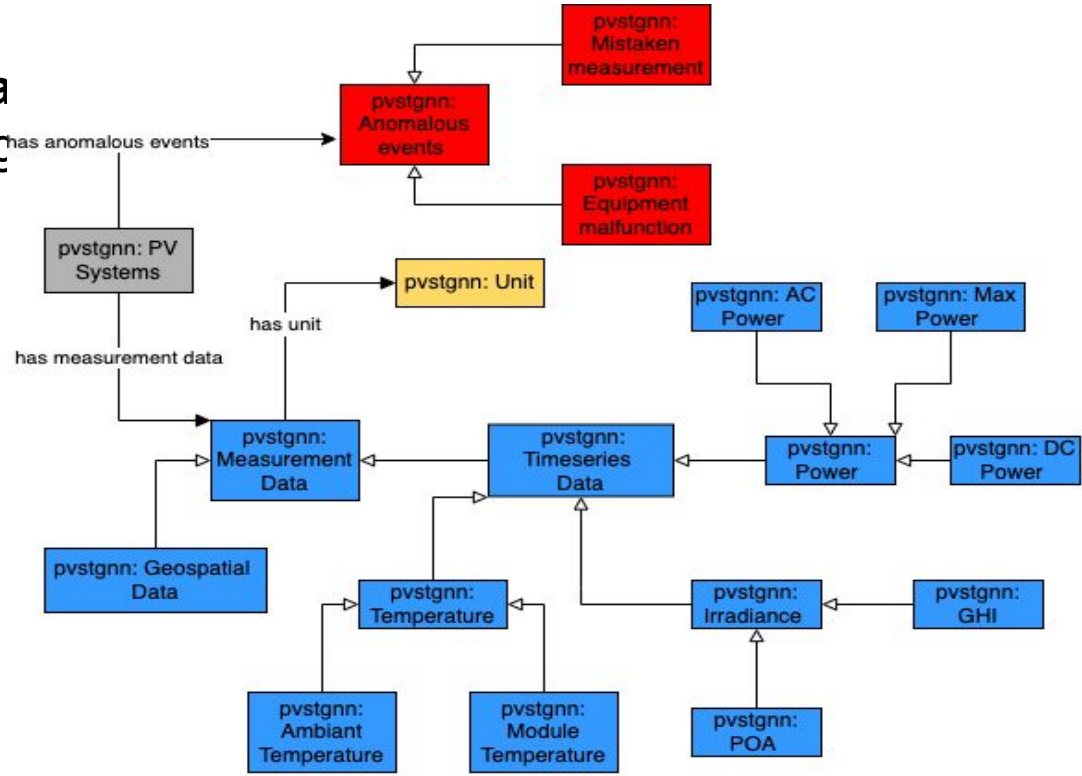
Metadata is "information about data"

Example: Split metadata from PV Power Plants
Into human readable "chunks"

- Location
- Array
- Inverters
- Time series

And define key-value pairs

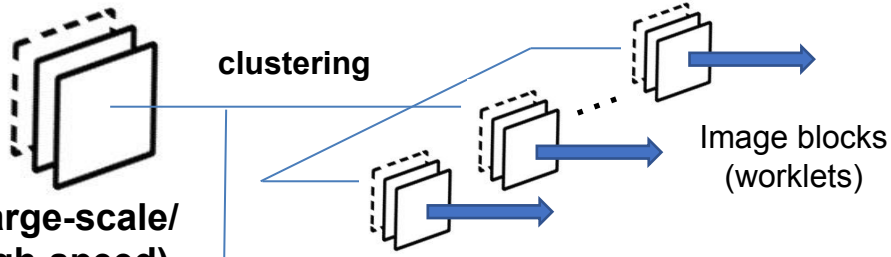
- Each "key" is a defined word
- Defined by schema.org of W3C
- Using JSON-ld



"PV System Measurement Data Ontology"
(DoE SETO "AI for PV" st-GNN)

Feature extraction: from FAIR data to (attributed) objects

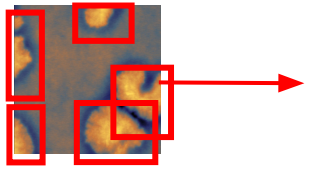
"Fast Image/Feature Detection: YOLO on AFM images (MDLE)"



(Large-scale/ high-speed) Image/video data

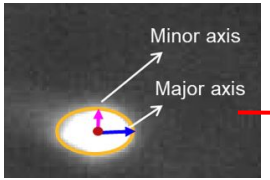
Image/video metadata (size, method, mode, device, environment, materials…)

Supporting techniques:
- Image/video segmentation
- Image/video clustering
- object detection (Mask R-CNN, Faster R-CNN, YOLO family)



Parallel object detection

"Stars"
(entity + attributes)



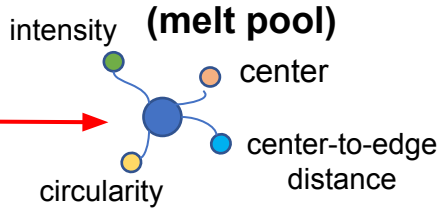
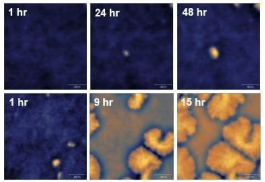
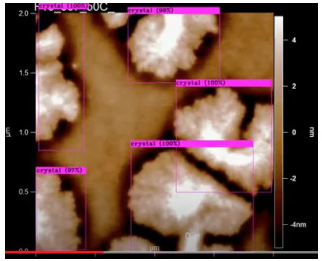


Image source: Gaikwad, Aniruddha, Brian Giera, Gabriel M. Guss, Jean-Baptiste Forien, Manyalibo J. Matthews, and Prahalada Rao. "Heterogeneous sensing and scientific machine learning for quality assurance in laser powder bed fusion–A single-track study." Additive Manufacturing 36 (2020): 101659.

Summary Graph Generation: make individual objects “linked”

(Training)
Image/video data





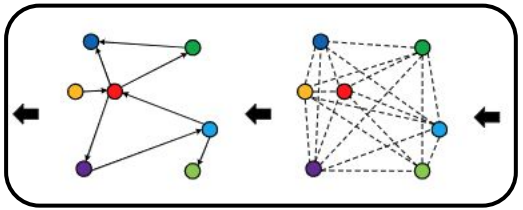
objects/image features

Annotation
(a node classification task)

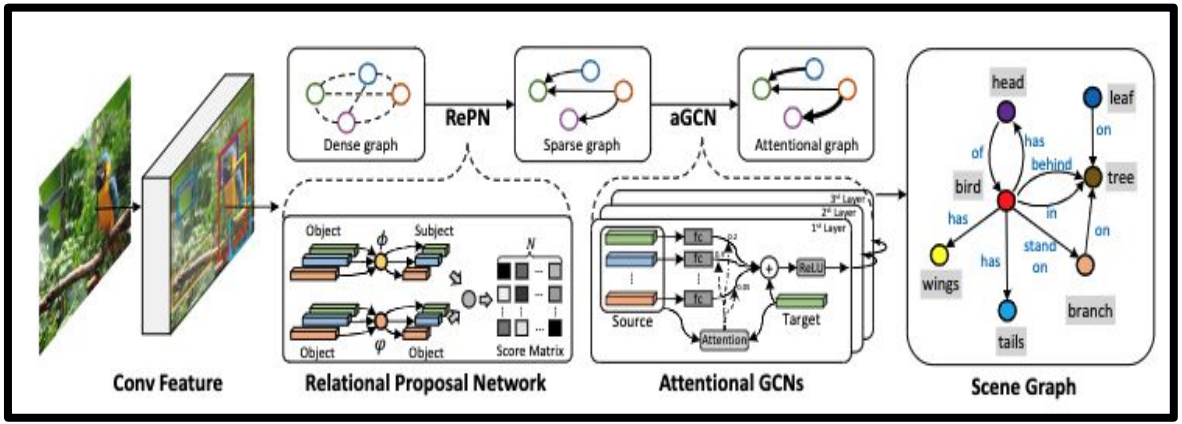
few-shot detection

Relation Extraction
(a link inference prediction task)

Link inference & classification

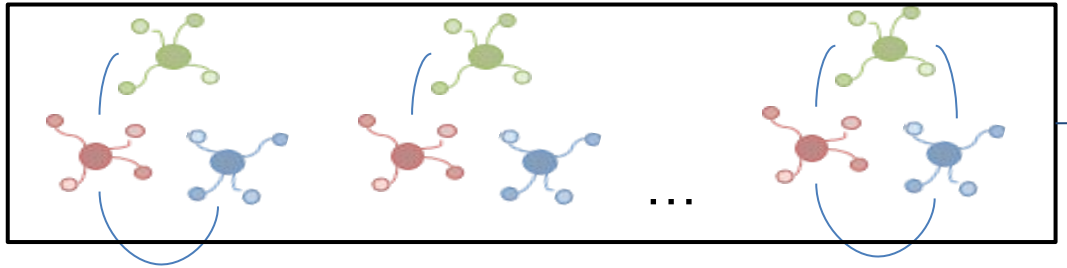


Graph R-CNN (GT & Facebook AI, ECCV ’18)
(Relational Proposal Network + Attention GCN)

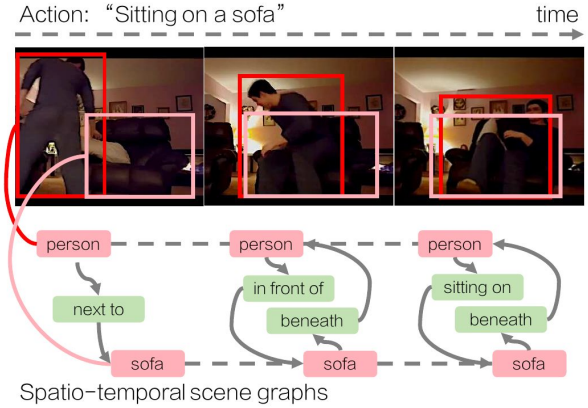


Spatiotemporal graphs: from static data to evolving sequences

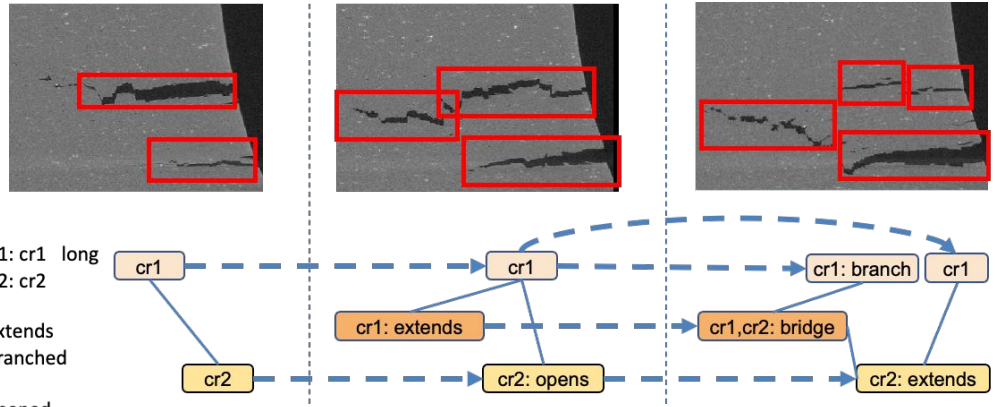
A sequence of (summary) graph “snapshots” (st-graph)



cost-effective
st-data search
& learning



“Action Genome: Actions as Composition of Spatio-temporal Scene Graphs”, J.Ji, et.al. CVPR 2020



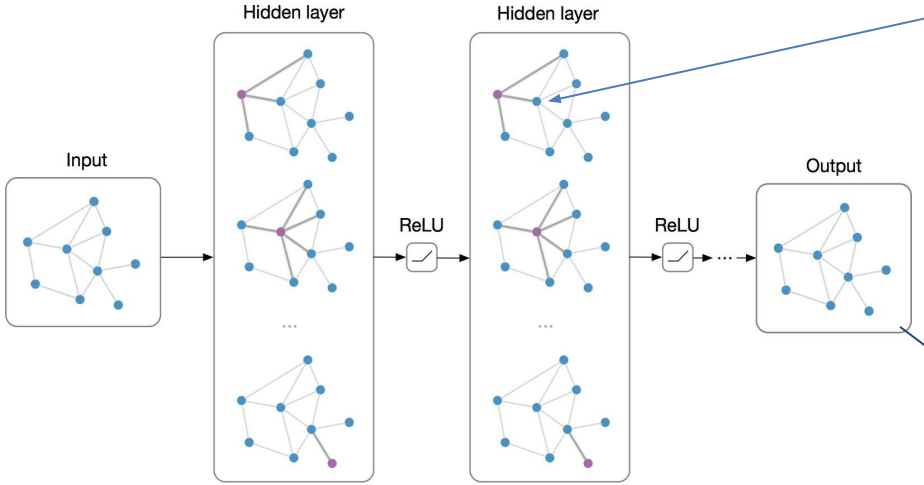
In the environmental cracking in 5XXX Alloys, dataset XCT-2, a st-scene graph captures the spatiotemporal correlation among the image features in real-space and their evolution through time. This verifies the growth of cracks 1 (cr1) and 2 (cr2) which, proceeding from left to right, cr1: extends and branches while cr2: opens and extends. Finally, cr1 and cr2 interact leading to bridging and the two cracks connect through the sample volume.

Graph Representation Learning: enable deep learning on graphs

Graph Neural Networks (GNNs)

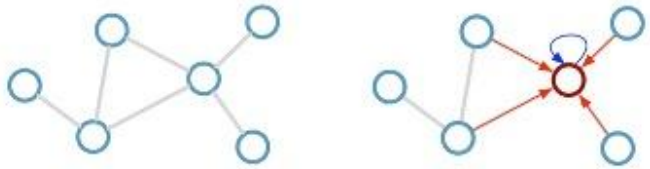
Notation: $\mathcal{G} = (\mathbf{A}, \mathbf{X})$

- Adjacency matrix $\mathbf{A} \in \mathbb{R}^{N \times N}$
- Feature matrix $\mathbf{X} \in \mathbb{R}^{N \times F}$



Consider this undirected graph:

Calculate update for node in red:



Update rule: $\mathbf{h}_i^{(l+1)} = \sigma\left(\mathbf{h}_i^{(l)}\mathbf{W}_0^{(l)} + \sum_{j \in \mathcal{N}_i} \frac{1}{c_{ij}} \mathbf{h}_j^{(l)}\mathbf{W}_1^{(l)}\right)$

Scalability: subsample messages [Hamilton et al., NIPS 2017]

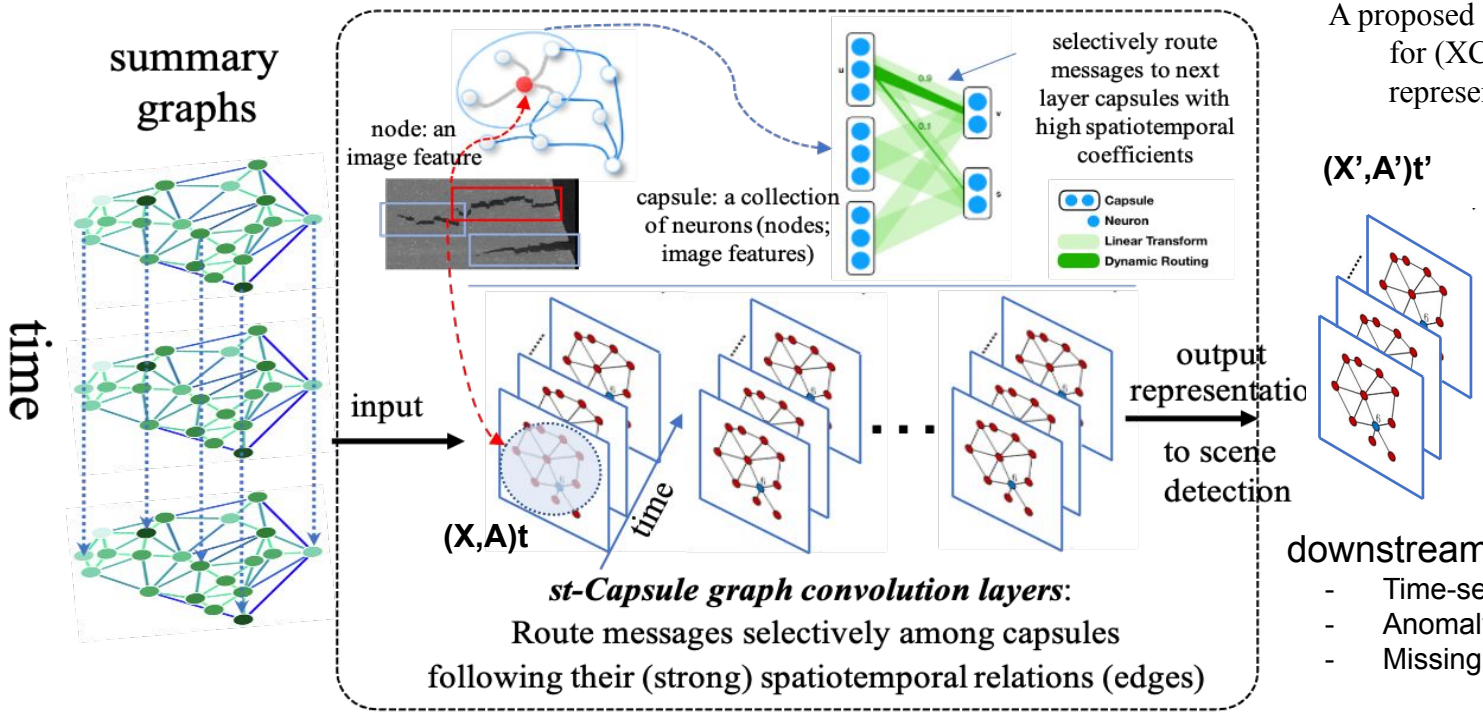
downstream tasks:

- node classification, link prediction…
- graph classification, graph clustering…
- anomaly detection, data imputation

Idea: Pass messages between nodes and agglomerate to refine node/edge representations

Spatiotemporal GNNs: deep learning on st-graphs

- **Jointly learn node-level, spatial-, and temporal representation: GT = {G1, …, Gt, …}**
- **Capture dynamic nature of evolving networks**



A proposed st-GNN architecture for (XCT/XRD) image representation learning

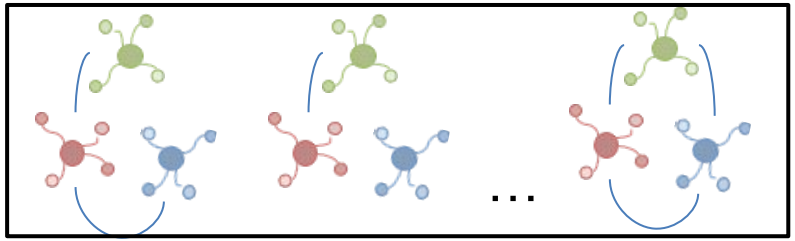
downstream tasks:

- Time-series data prediction
- Anomaly detection & classification
- Missing data imputation

st-Scene Generation: from st-graphs to "scenes"

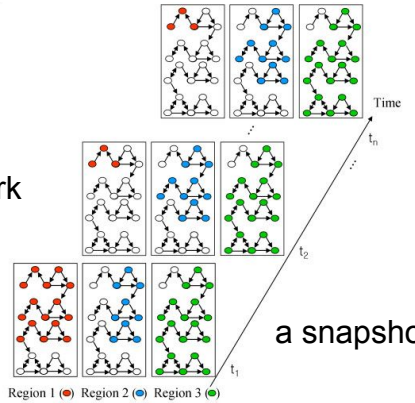
a snapshot

Summary sequences



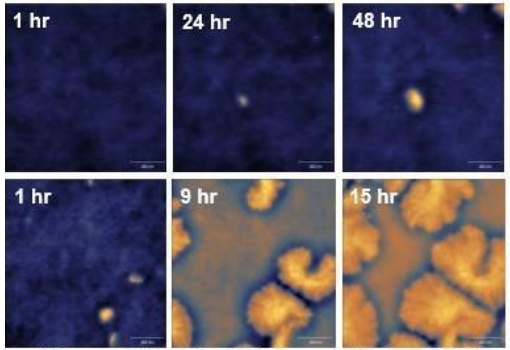
pattern detection

a spatiotemporal network pattern mining task



a snapshot

Recurrent, statistically significant small subgraph patterns



sequence segmentation & scene recognition (train predictive st-GNN models)

st-graph pattern mining/ association analysis:

- Value dependencies * ($X \rightarrow Y$)
- Dense *
- Frequent *
- Burst *
- Periodic ("Blinking")
- Durable*
- Co-occurrences ($A \rightarrow B,\ \Delta T$)*
- Negative correlated

Pattern-wise association analysis

"genome" scenes e.g., association rules