THE UNIVERSITY of TENNESSEE KNOXVILLE

Lecture 03: Python Ecosystem, Data, and LLMs

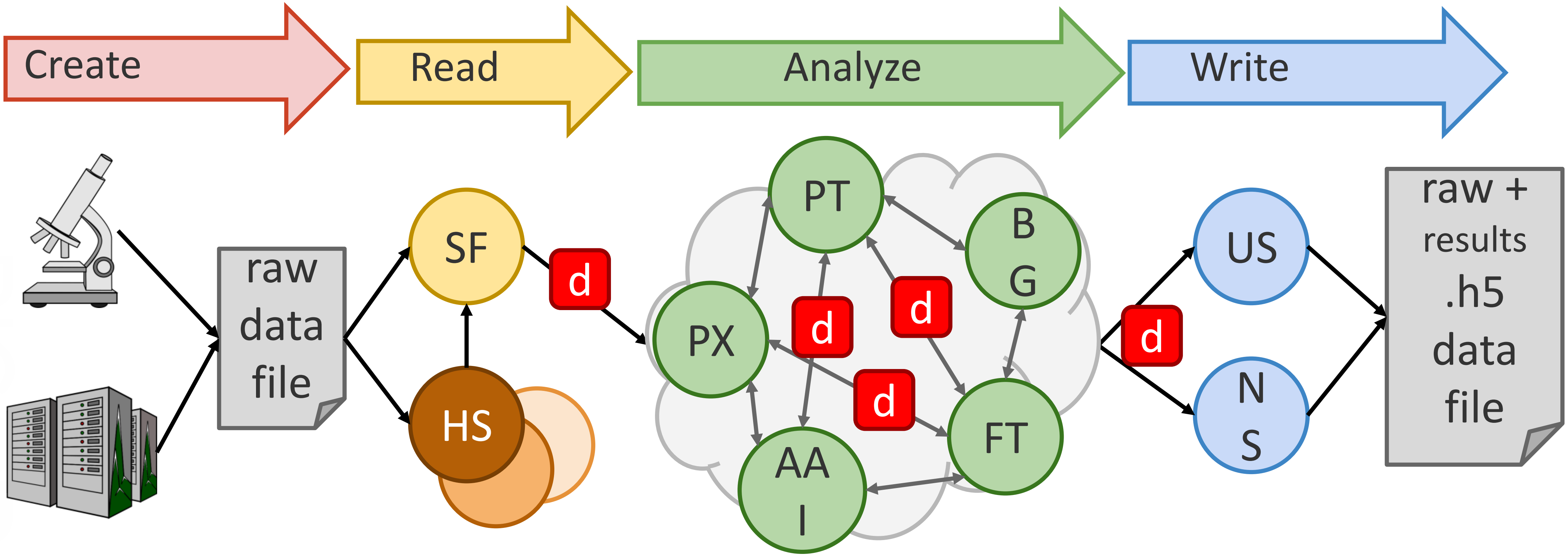
Instructor: Sergei V. Kalinin

- **Machine learning** begins with data—it's the foundation upon which models are built. As it evolves, ML increasingly incorporates heuristics, physical laws, complex data structures, and inferential biases, gradually moving from raw data-driven approaches to more sophisticated, context-aware systems.

- **Physics** starts with fundamental principles and laws—these are the bedrock of our understanding of the natural world. From these principles, physics builds models and theories, which are then validated or refined through experimentation and observation, often leading to new discoveries and deeper insights.

- **Mathematics** begins with axioms and definitions—from these basic building blocks, mathematics constructs a logical framework of theorems and proofs. It provides the rigorous language and tools necessary for describing, analyzing, and solving problems across all scientific disciplines, including physics and machine learning

Solutions: Integrated Ecosystems

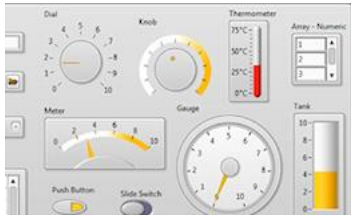


Data from measurements or simulations are read into **sidpy.Dataset** (d) objects directly by **SciFiReaders** (SF). Data are processed using multiple science packages in the PyCroscopy ecosystem that interoperate via **Dataset** objects. **Dataset** objects are written to HDF5 files via **pyUSID** (US) or **pyNSID** (NS).

Slide courtesy of R. Vasudevan

Solutions: Integrated Ecosystems


Instrument Tier



Automated, standardized, modularized data acquisition

Instrument-agnostic, self-describing, model in an open friendly file format

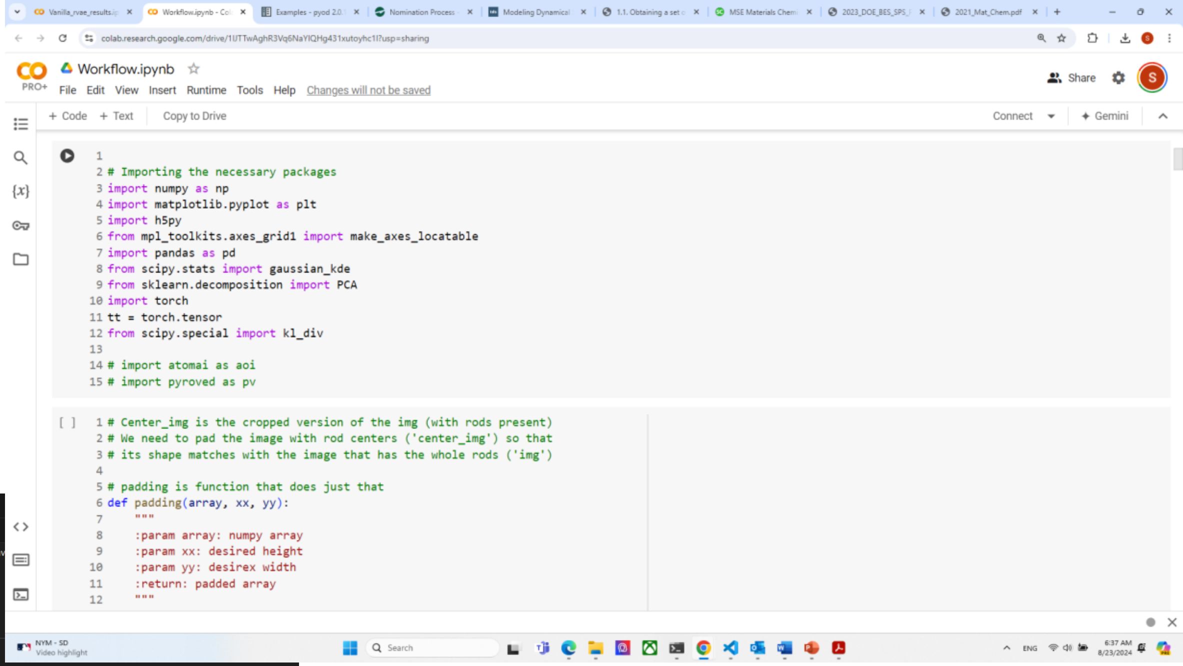
Centralized repository for data processing, analysis

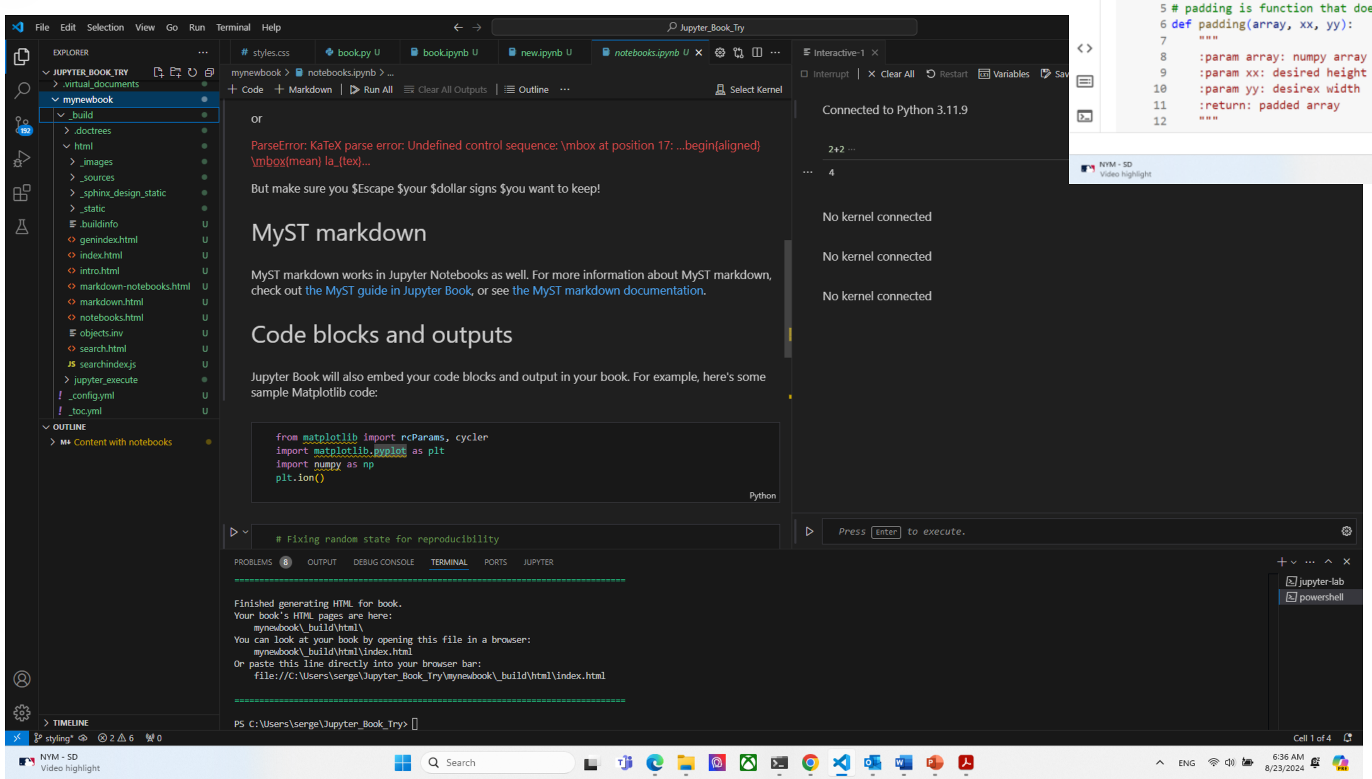
Interactive visualization + analysis + storage on the cloud

Slide courtesy of R. Vasudevan

How we can run code:

- Google Colabs
- AWS SageMaker notebooks
- IDE: Spyder, PyCharm, etc.
- Command line interface





Where would the code run?

- Cloud GPU
- Local computer
- Selected HPC (ISAAC for UTK)

What language do we use:

Main Python libraries we will use:

1. NumPy
2. MatPlotlib
3. Scikit-learn
4. Keras

Other libraries we may use:

1. Seaborn
2. GPax
3. SciPy

- This course
- Read the docs
- Blogs (e.g. Medium)
- Packt, Manning, etc.
- Papers with codes
- GitHub repos

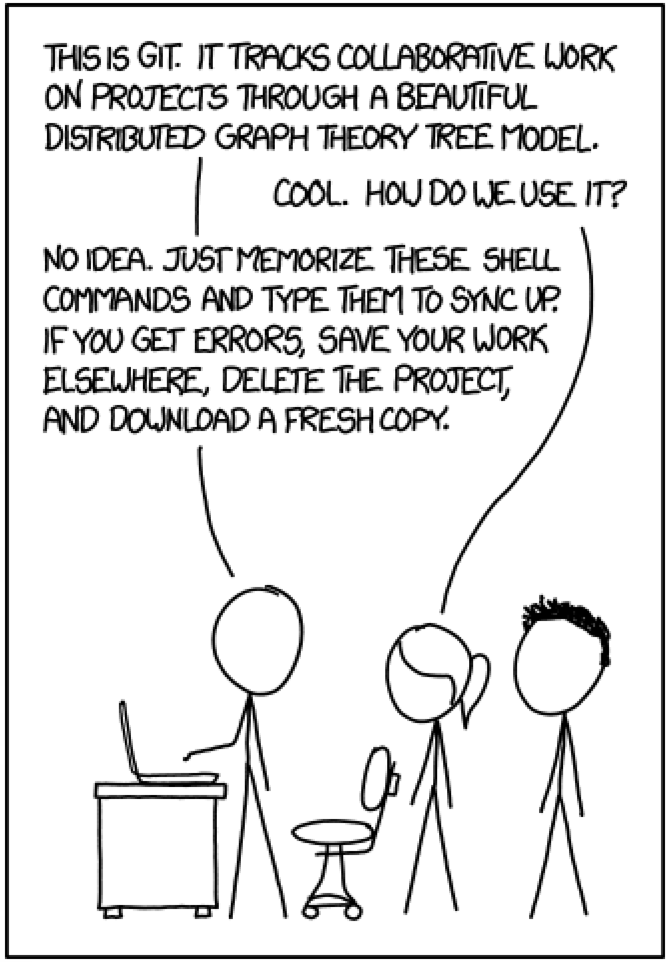
We will learn these as we need them!

We will also start to learn how to select sources

Code Repositories and Version Control

- Sharing scripts between users can be workable for immediate or short-term needs, but is not scalable nor lasting
- For reproducibility, it is better to have codes that reside in packages that are documented and well tested
- Most of you are familiar with python packages; but many are probably new to version control
- Version control systems such as git enable multiple people to work on a single software project at the same time to speed up development and ensure consistency
- Git is an open-source distributed version control system. It maintains a history of changes that have occurred in the project and allows for updates as well as reversions to older 'commits'.

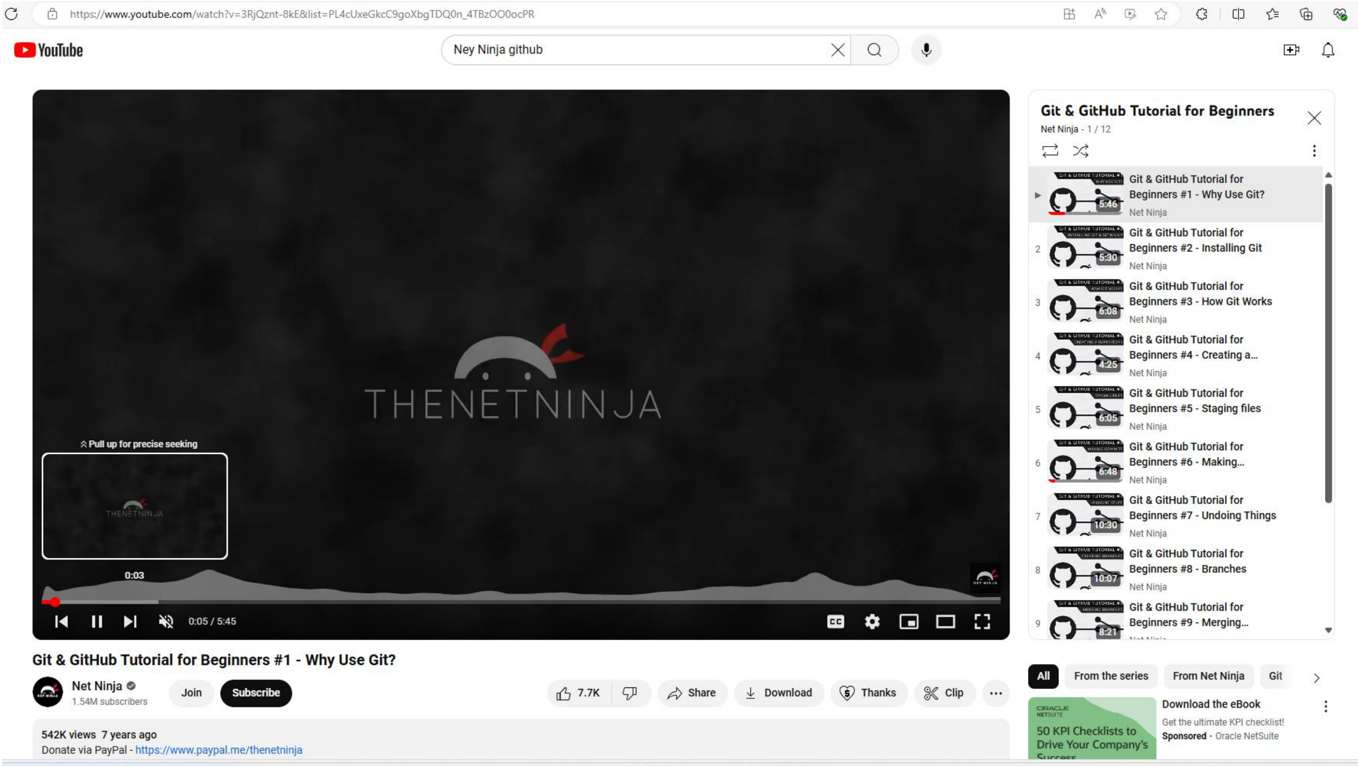
How we can share code:



Git would take a significant amount of time to explain in detail. However, there are plenty of online tutorials, e.g.

https://www.atlassian.com/git/tutorials

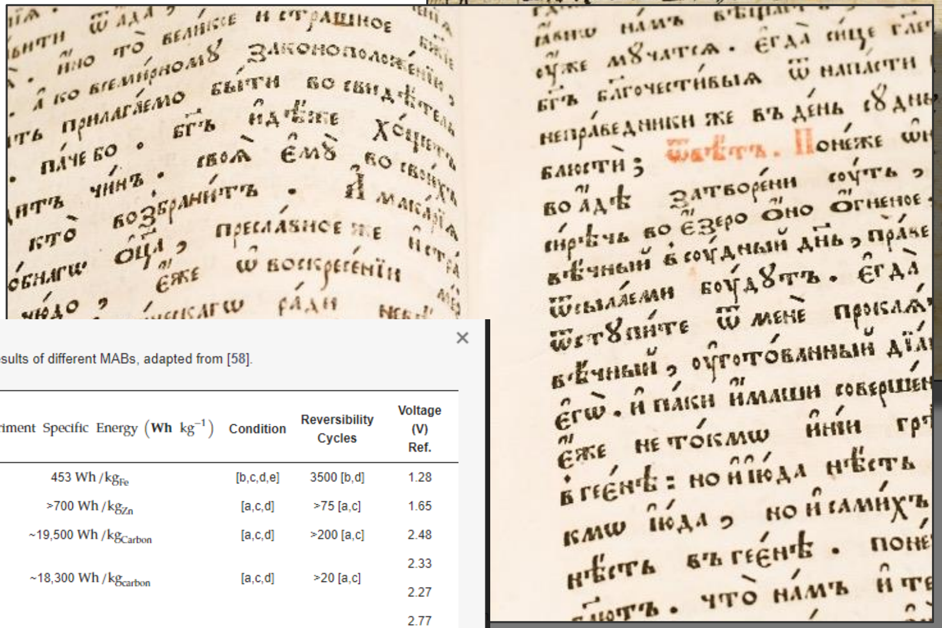
Net Ninja, Git and GitHub tutorial for beginners



https://www.youtube.com/watch?v=3RjQznt-8kE&list=PL4cUxeGkcC9goXbgTDQ0n_4TBzOO0ocPR

Get data – from publications

- Printed matter
- Pdf files with figures and tables
- Deposited data files
- (Rarely) workflows



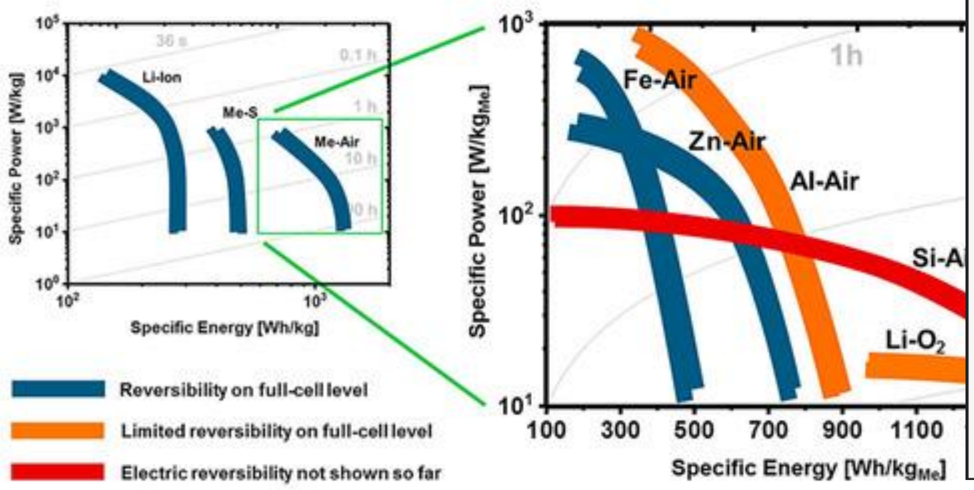
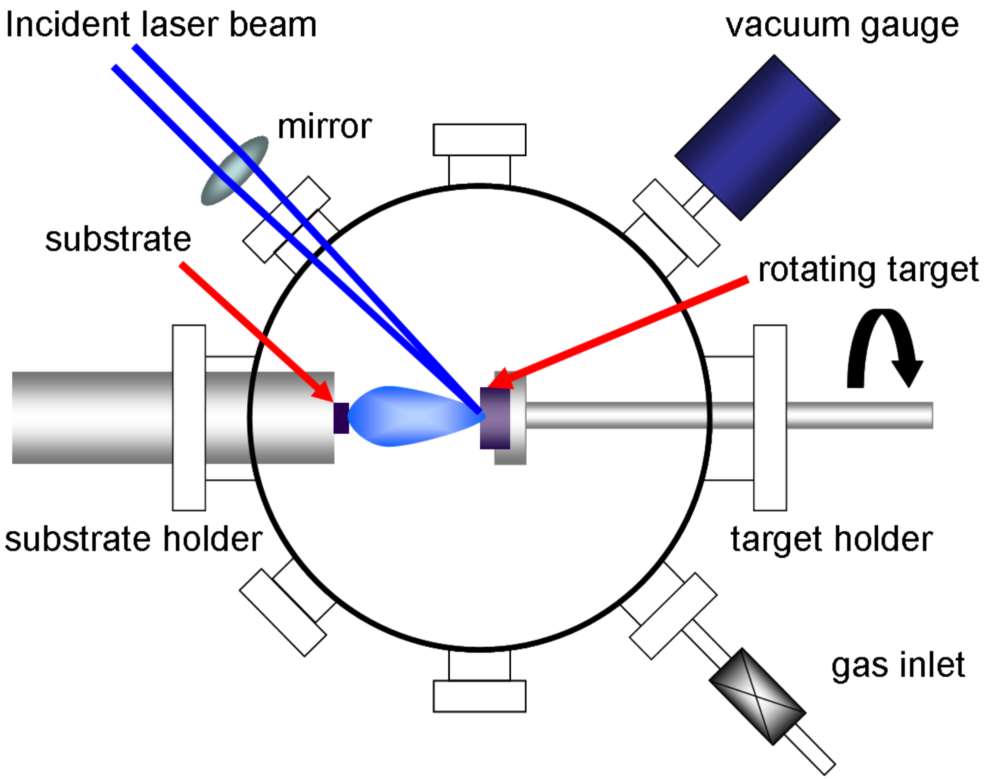


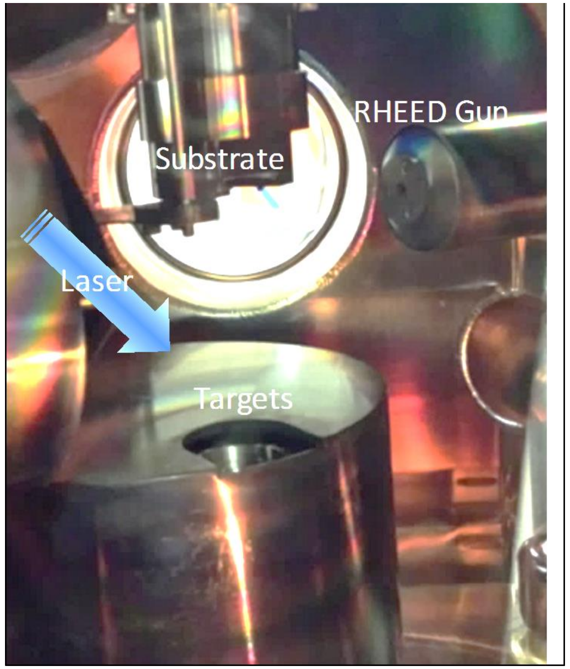
Table 5. The recent experimental results of different MABs, adapted from [58].

MAB	Discharge Product	Experiment Specific Energy (Wh kg^{-1})	Condition	Reversibility Cycles	Voltage (V) Ref.
Fe/O_2	$Fe(OH)_2$	453 Wh /kg_{Fe}	[b,c,d,e]	3500 [b,d]	1.28
Zn/O_2	ZnO	>700 Wh /kg_{Zn}	[a,c,d]	>75 [a,c]	1.65
K/O_2	KO_2	~19,500 Wh /kg_{Carbon}	[a,c,d]	>200 [a,c]	2.48
Na/O_2	Na_2O_2 NaO_2	~18,300 Wh /kg_{carbon}	[a,c,d]	>20 [a,c]	2.33 2.27
Mg/O_2	$Mg(OH)_2$ MgO	~2750 Wh /$kg_{cathode}$	[a,c,d,f]	<10 [a,c,d]	2.77 2.95
Si/O_2	$Si(OH)_4$ SiO_2	~1600 Wh /kg_{Si}	[a,c,d]	Not yet	2.09 2.21
Al/O_2	$Al(OH)_3$ Al_2O_3	~2300 Wh /kg_{Al}	[a,c,d]	Limited	2.71 2.1
Li/O_2	Li_2O_2 Li_2O	>11,050 Wh /kg_{carbon}	[a,c,d]	>250 [a,c]	2.96 2.91

Conditions: a is anode sheet/foil, b is porous/particulate anode, c is full-cell measurements, d is 100% deep discharge, e is repeated charge/discharge, and f is elevated temperature.

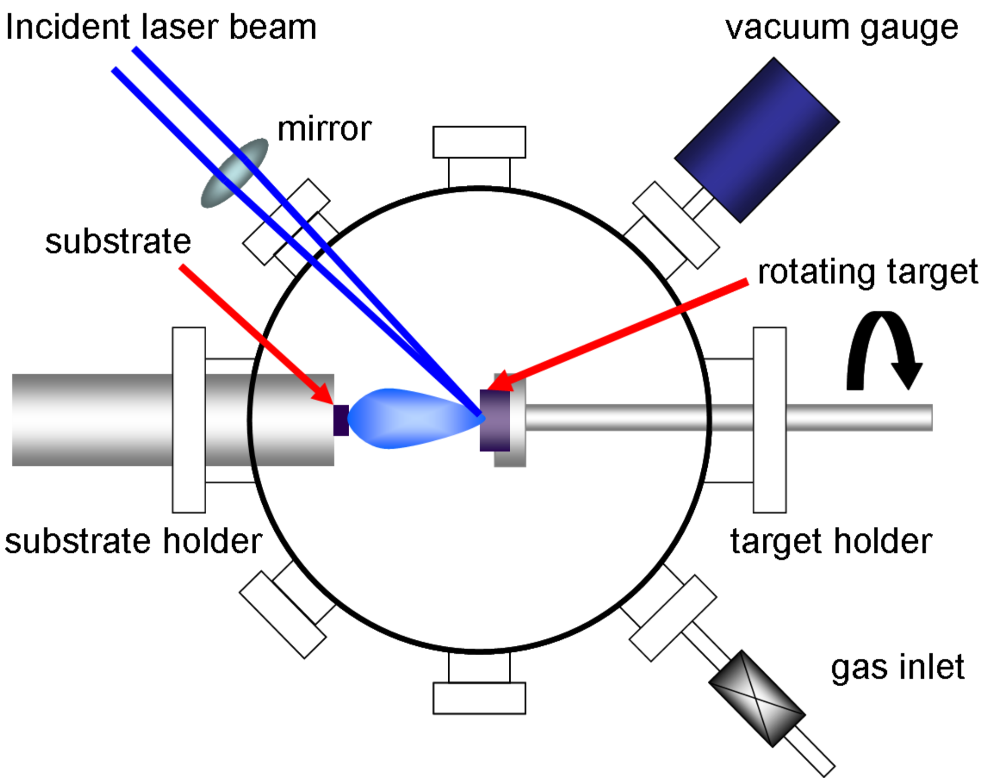
Pulsed Laser Deposition





- PLD benefits: great films, ease of setup, wide variety, unit-cell precision in growth
- Downside: difficult to correlate growth parameter with film properties, largely trial and error approach

Pulsed Laser Deposition



Parameters to vary:

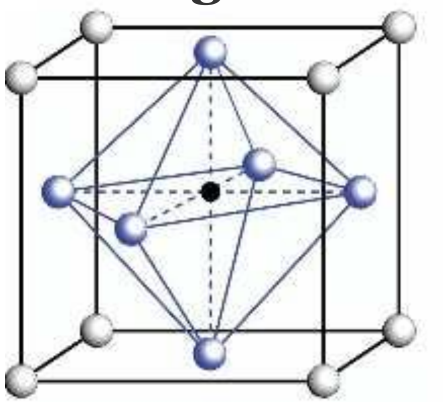
1. Substrate (compound, orientation)
2. Film (compound)
3. Growth temperature
4. Growth environment (e.g., pO_2)
5. Laser Fluence and repetition rate
6. Target-substrate Distance
7. Heterostructures (film electrodes, buffer layers, etc.)
8. Post-annealing
9. Cooling Rate

- Researchers will grow films by varying parameters, achieving different **functional property** results. These could include film roughness, stoichiometry, and specific properties, such as capacitance, remnant polarization, Curie temperature, etc.

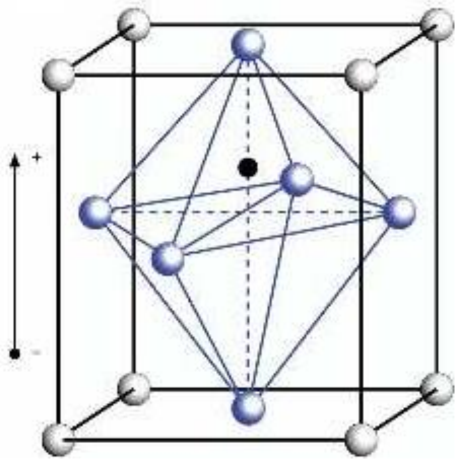
- Iterative procedure, since PLD conditions are generally not transferable.

Ferroelectric Materials

High T

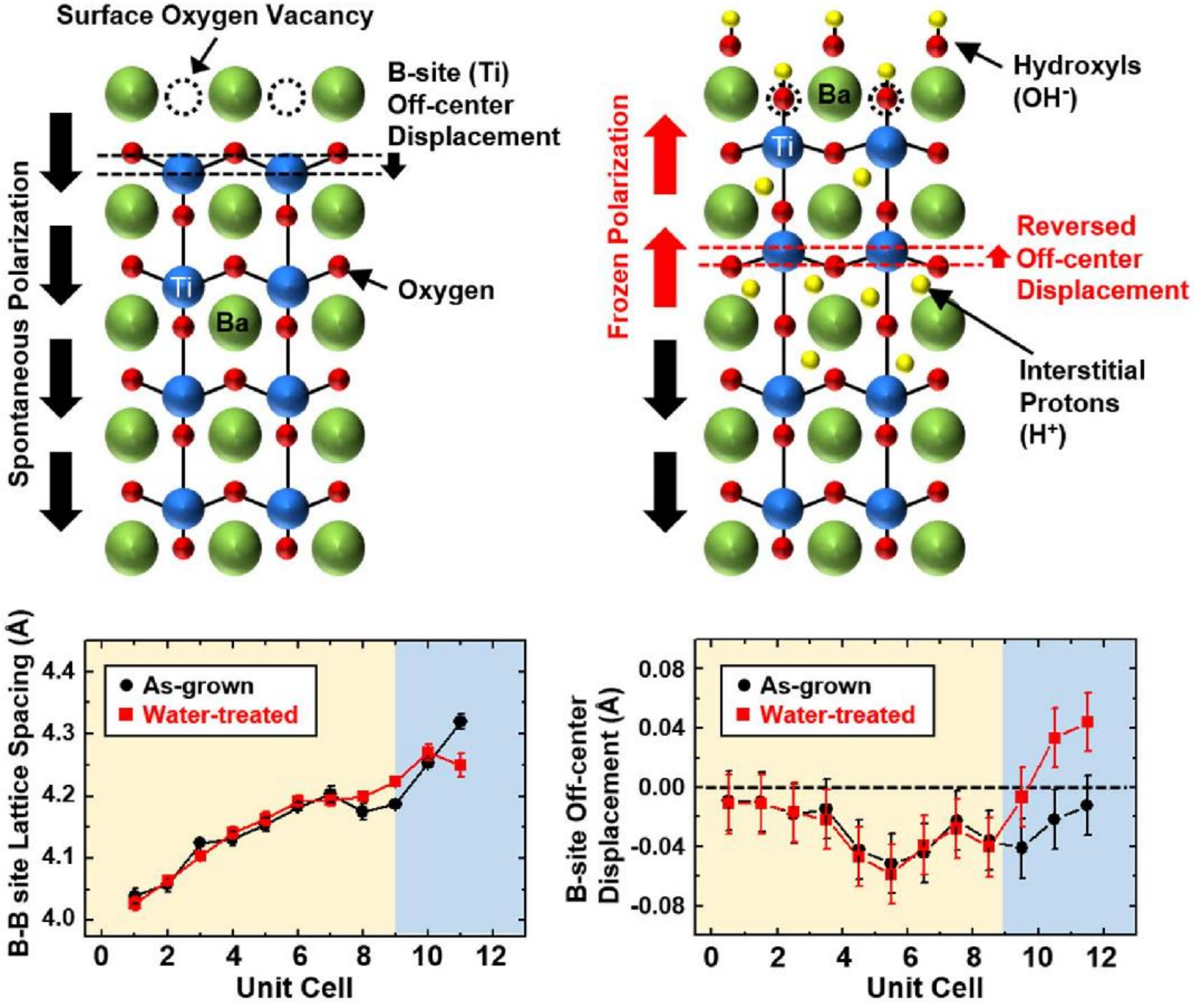


Low T



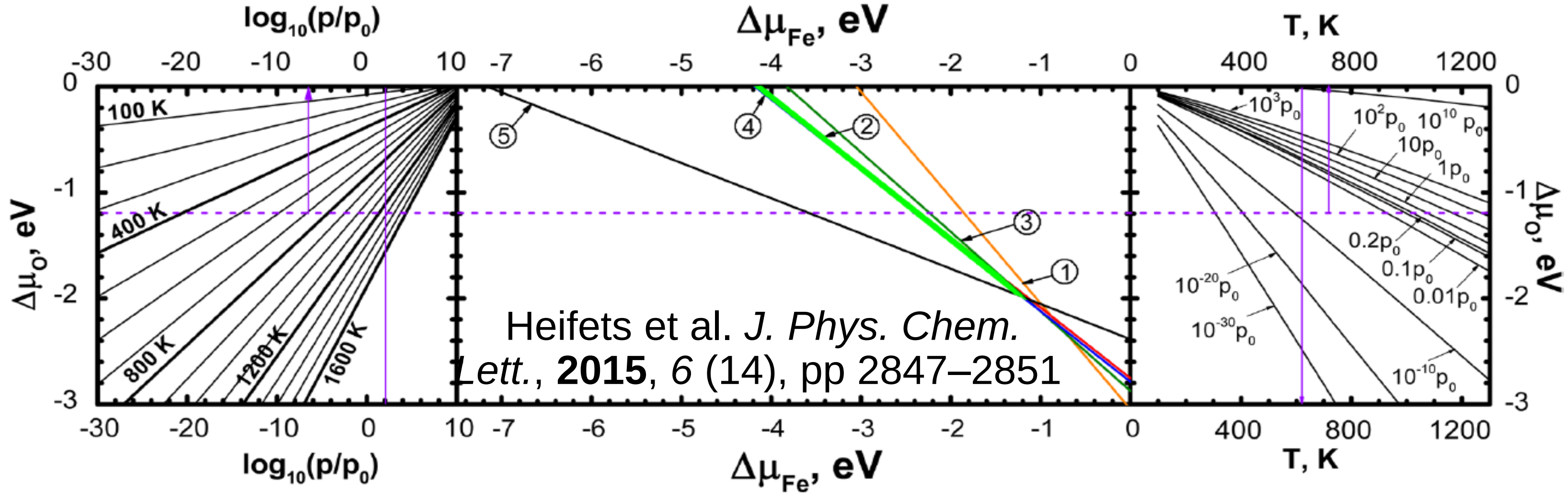
- Can store information
- Sensitive to charge
- Sensitive to strain

Surfaces and interfaces matter!



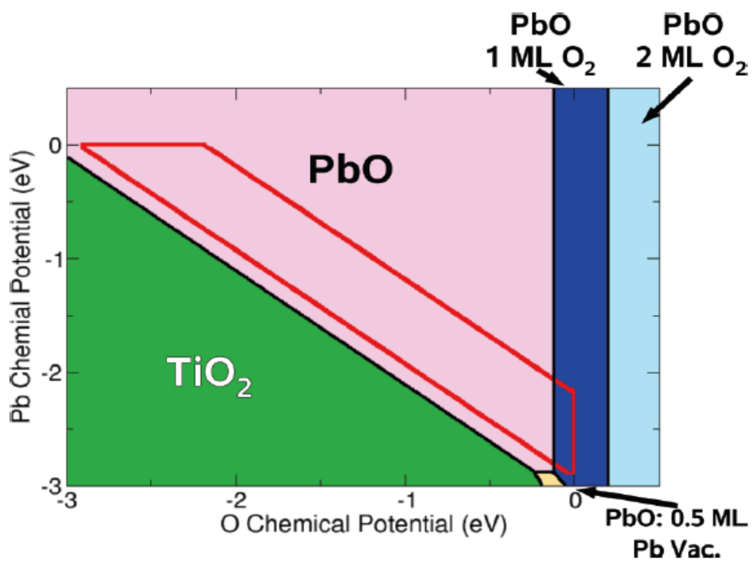
https://pubs.acs.org/doi/10.1021/acs.nanolett.5b05188

Can theory help?

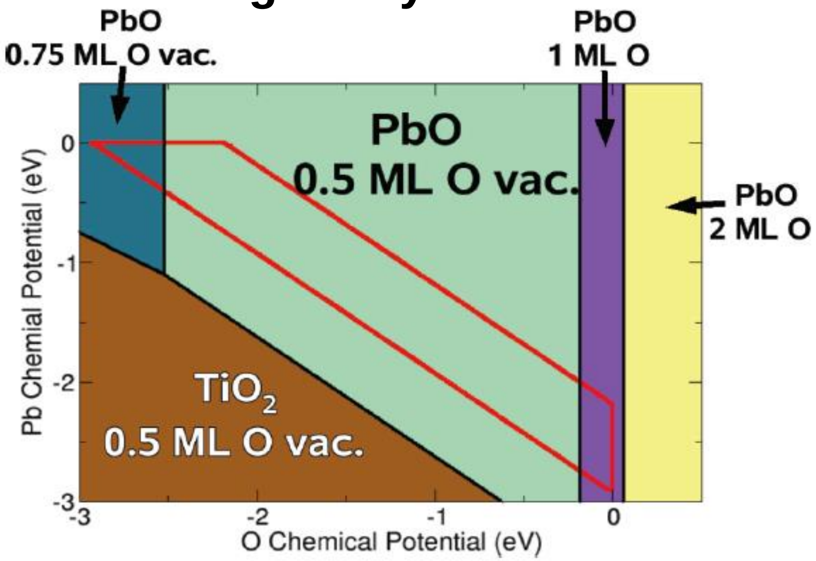


Heifets et al. *J. Phys. Chem. Lett.*, **2015**, *6* (14), pp 2847–2851

Positively Poled



Negatively Poled



Garrity et al. PRB 88, 045401 (2013)

Practical Considerations

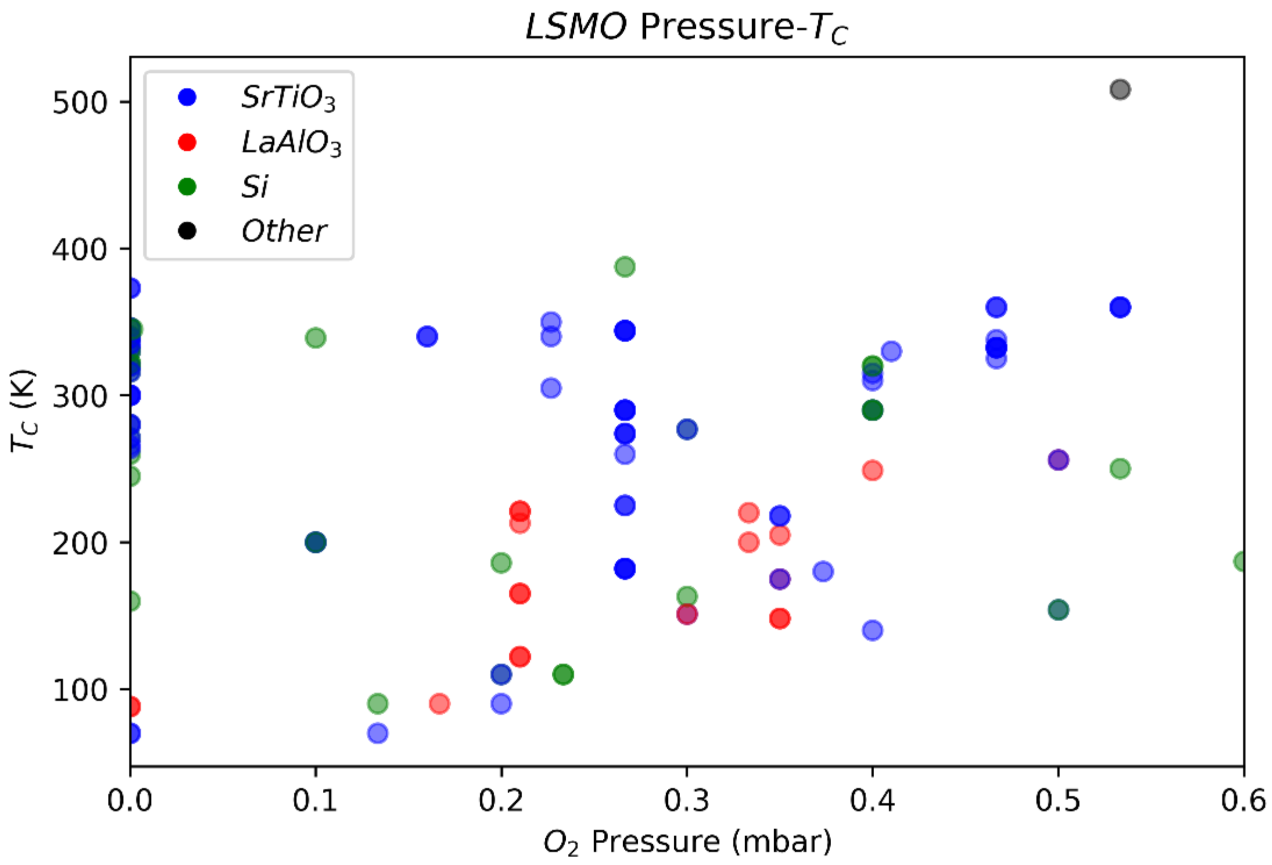
1. DFT is not always accurate, often doesn't account for defects.
2. Kinetic factors during growth.
3. Ideal stoichiometry not always correlated to best properties.

So, we need to know what conditions to use practically…

Slide courtesy of R. Vasudevan

We can mine publications for data. But…

- We get coercive temperature dependent on growth temperature and partial oxygen pressure.
- What’s next?



RESEARCH ARTICLE | MARCH 20 2018

Data mining for better material synthesis: The case of pulsed laser deposition of complex oxides

Steven R. Young; Artem Maksov; Maxim Ziatdinov; Ye Cao; Matthew Burch; Janakiraman Balachandran; Linglong Li; Suhas Somnath; Robert M. Patton; Sergei V. Kalinin; Rama K. Vasudevan

Check for updates

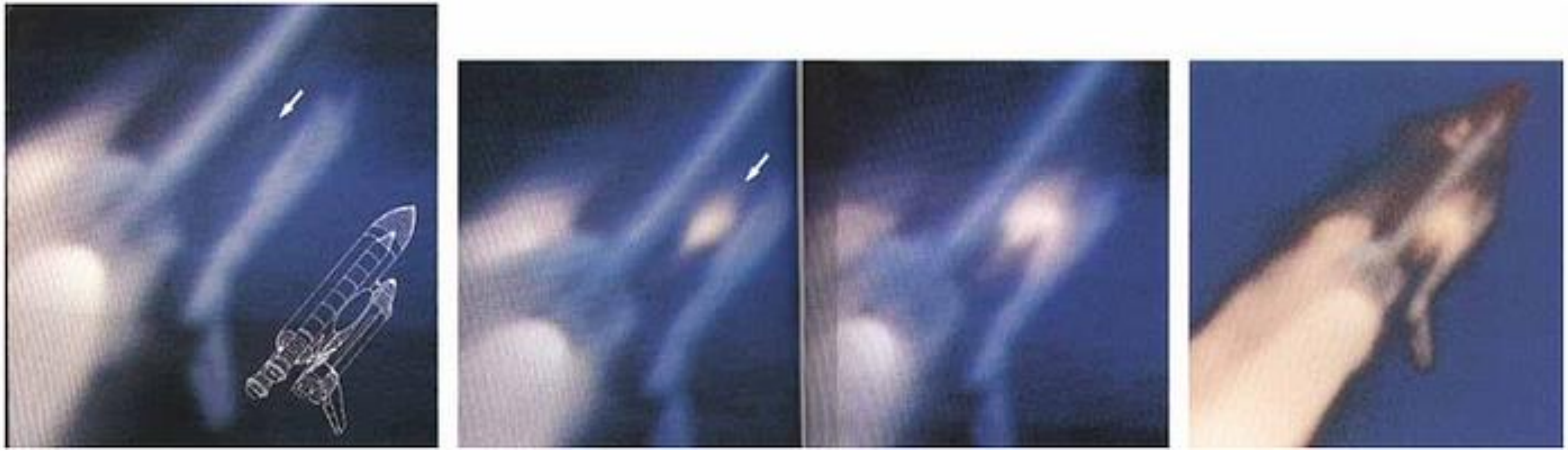
+ Author & Article Information

J. Appl. Phys. 123, 115303 (2018)
https://doi.org/10.1063/1.5009942 Article history

https://pubs.aip.org/aip/jap/article/123/11/115303/155202/Data-mining-for-better-material-synthesis-The-case

Slide courtesy of R. Vasudevan

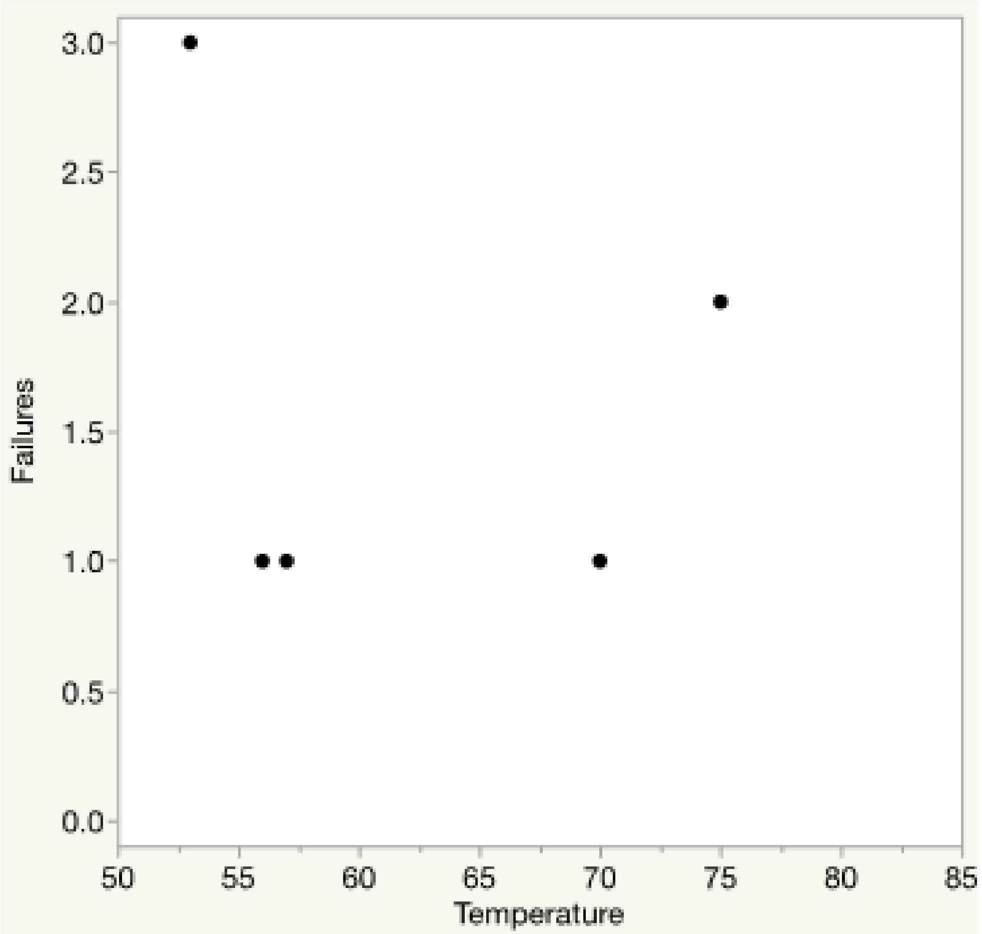
Challenger disaster



https://medium.com/habits-for-success/the-challenger-disaster-a-lesson-in-the-power-of-data-analysis-and-visualization-398d2ac8f59b

https://www.youtube.com/watch?v=hOEt1MOuYX4

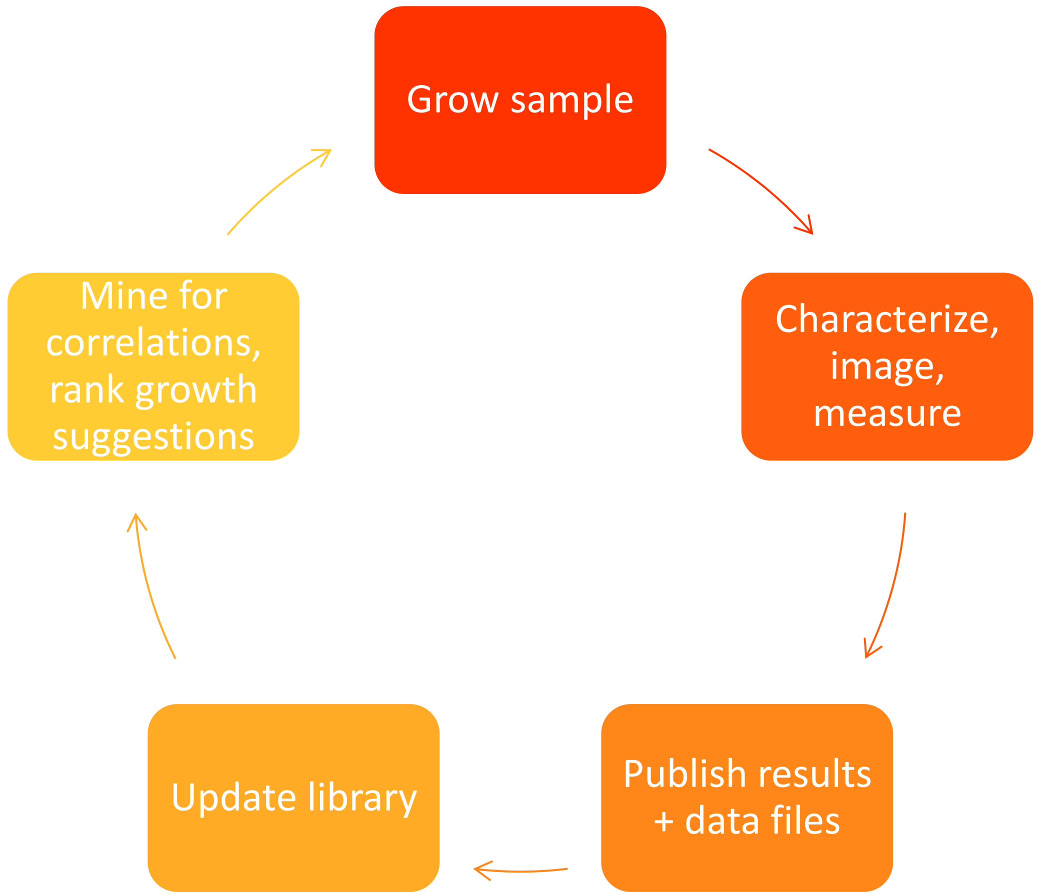
Challenger disaster

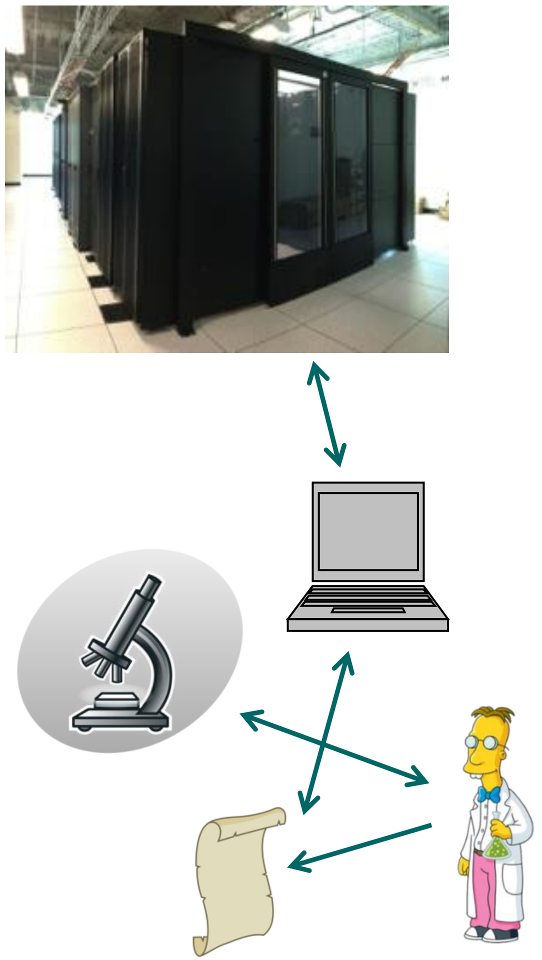


https://www.researchgate.net/figure/Challenger-Data-Number-of-O-Ring-Failures-Launch-Versus-TemperatureLeft-Panel-Five_fig2_344257416

Conclusion: requires community involvement

- Relevant sample preparation conditions
- Unique sample identifier
- Measured properties in appropriate format





Slide courtesy of R. Vasudevan

Get data – data bases!

- **On-line data bases**
  - Materials Project, https://next-gen.materialsproject.org/
  - Materials Data Facility: https://materialsdatafacility.org/
  - Papers with code: https://paperswithcode.com/datasets
  - Standard data sets for molecular properties, e.g. http://quantum-machine.org/datasets/
  - Integrators, e.g.: https://github.com/sedaoturak/data-resources-for-materials-science

- **Enterprise data bases**

Get data – LLM Analysis!

- **Use ChatGPT to:**
  - Analyze collection of abstracts
  - Mine the paper for data
  - Summarize paper
  - Reproduce code in the paper

Homework assignment 2 (part 2)

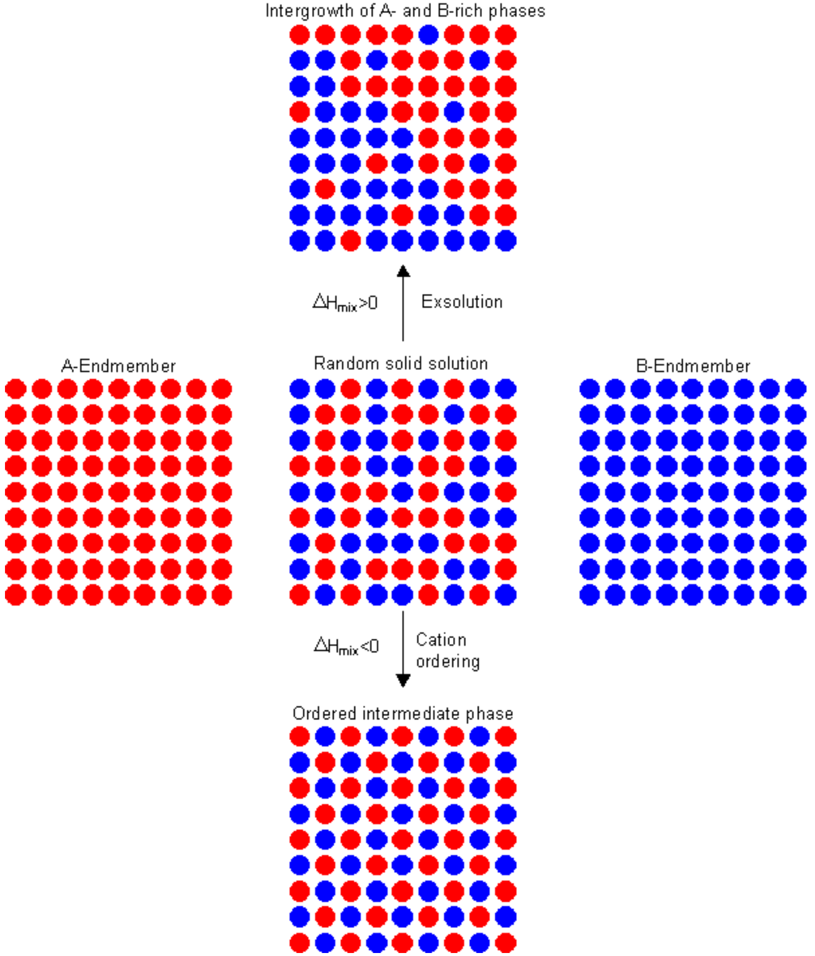
- **(advanced) Use LangChain to train ChatGPT on:**
  - Multiple papers
  - Finetune on arxiv, chemrxiv, etc

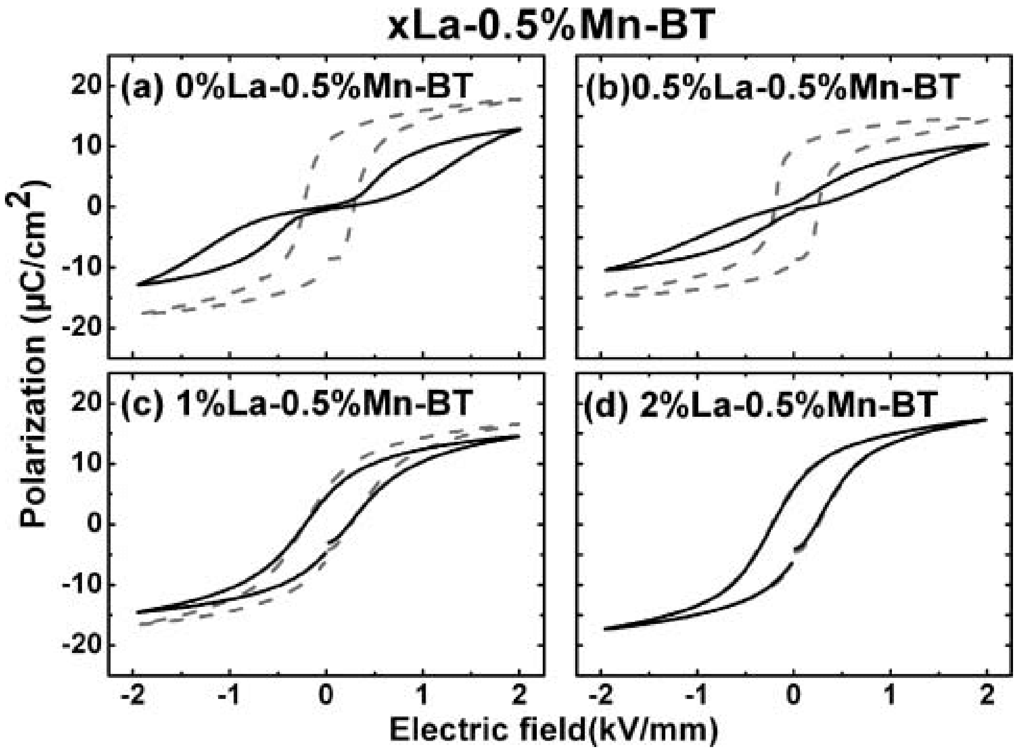
- **Expect more changes in the near future!**

ChatGPT Example

1. "Let's analyze scientific papers. I will upload three papers, and you will extract information about materials. Ok?“
2. "This is paper 1“
3. "This is paper 2“
4. "This is paper 3“
5. "Make a table of the compounds studied in these three papers, including name, formula, and lattice parameters.“
6. "What parameters are known for all studies compounds?“
7. "Has these compounds been studied by XRay scattering?“
8. "Can you extract positions of XRay peaks?“
9. "Make a list of prompts that I made"

Cause and effect in doped ferroelectrics

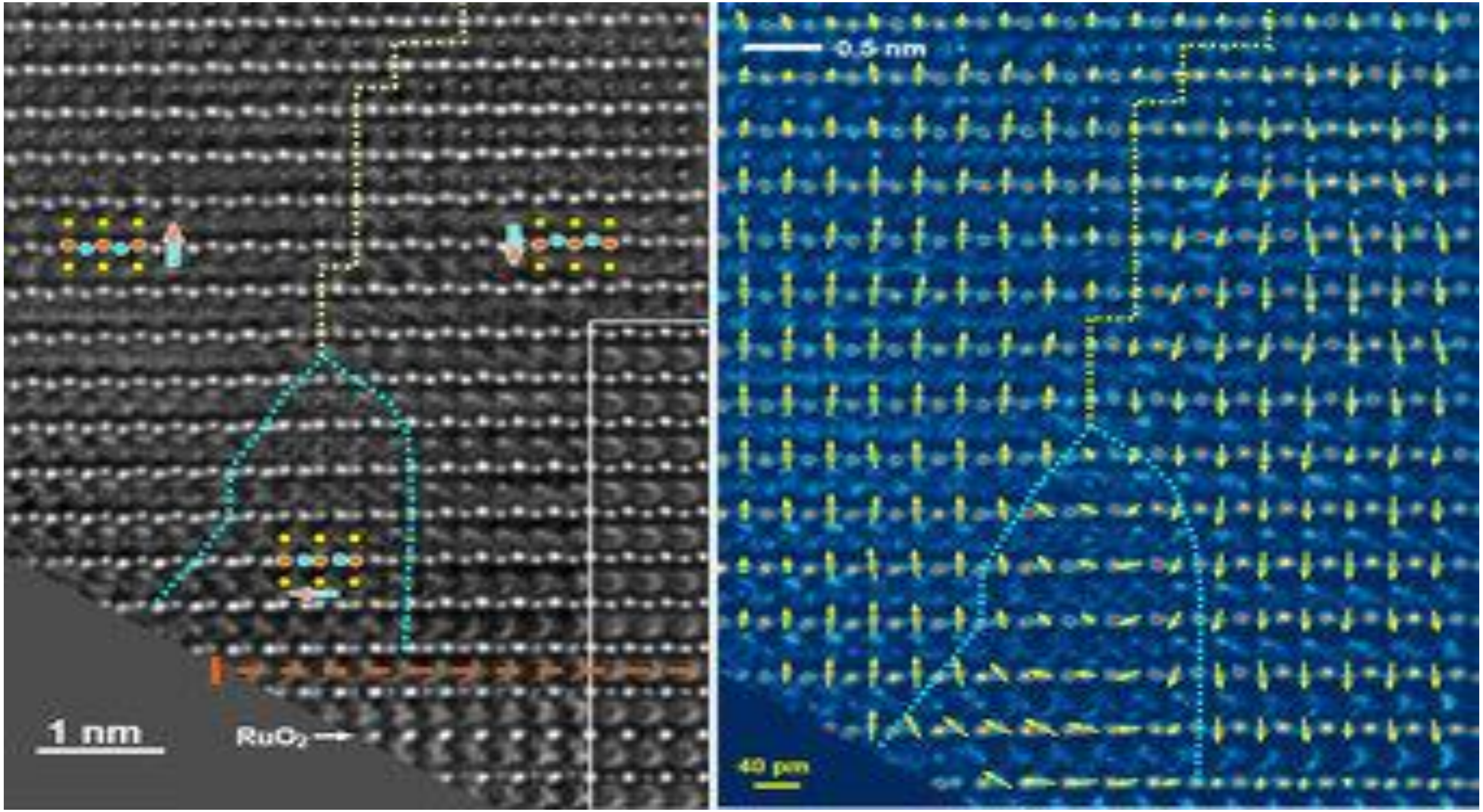




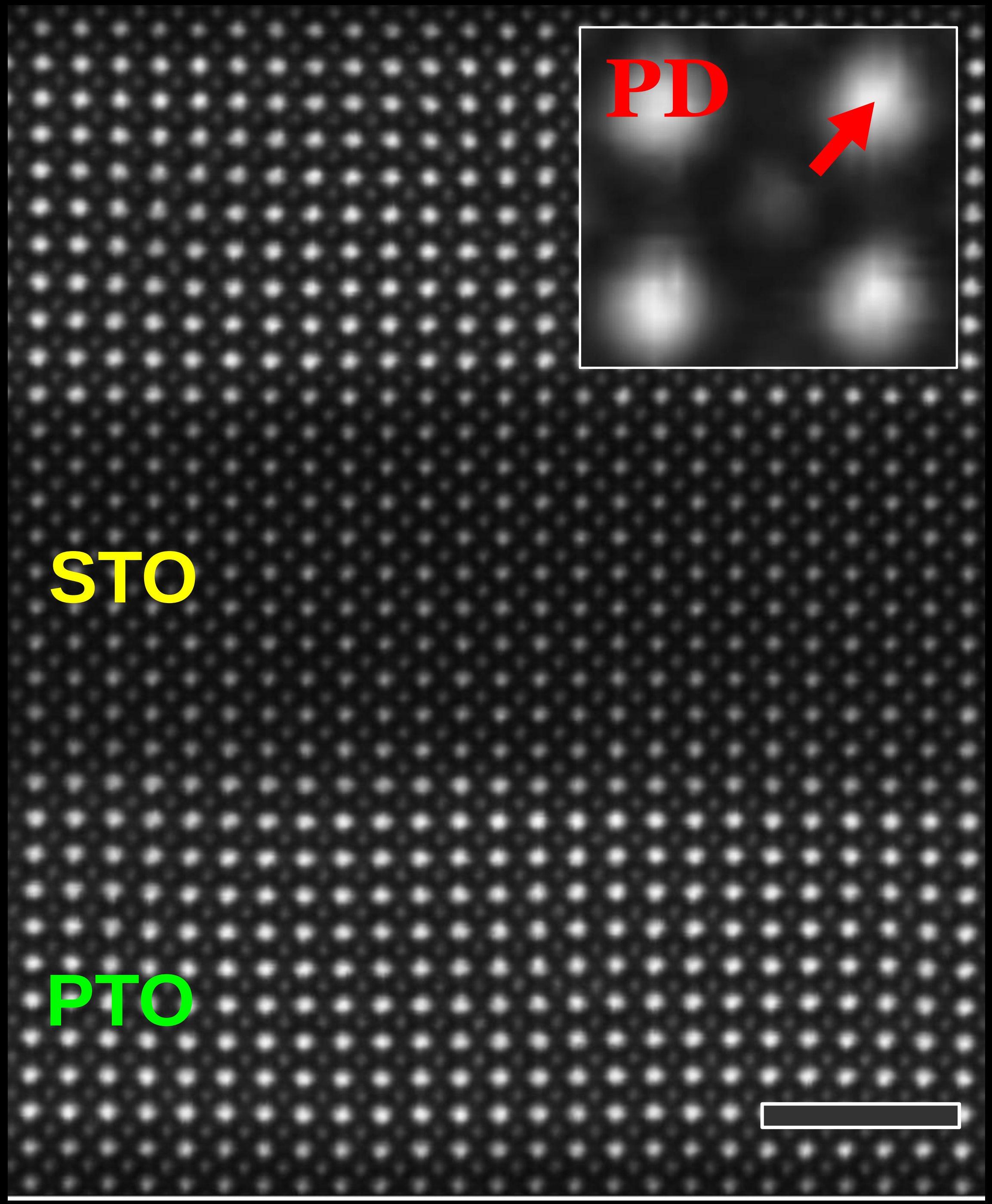
- We generally assume that cationic order is frozen at the state of material formation, and then polarization field evolves to accommodate charged dopants.
- However, ions can move to compensate polarization – segregation at the domain walls, memory effects, etc.

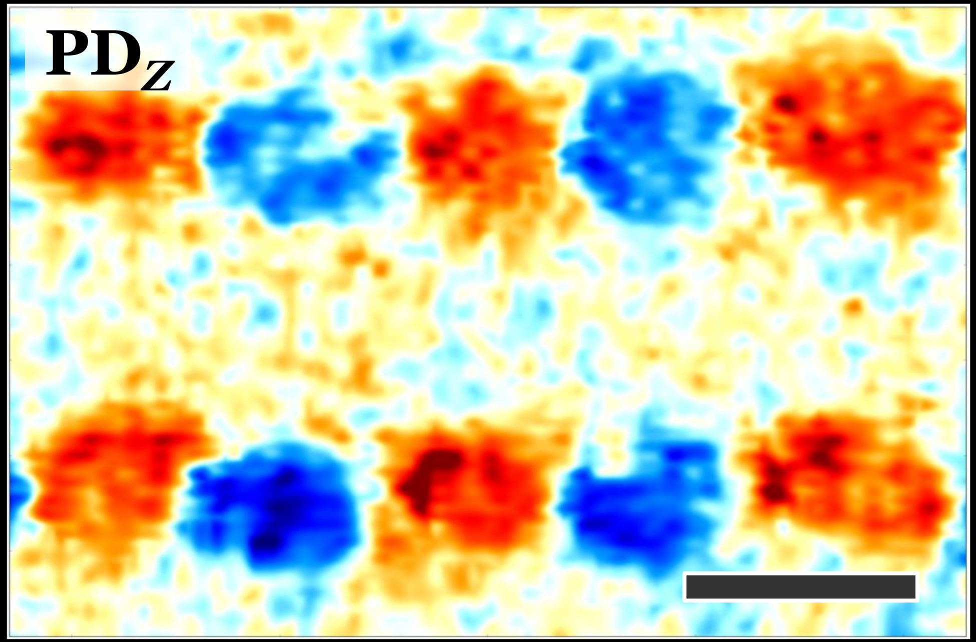
https://www.doitpoms.ac.uk/tlplib/solid-solutions/printall.php

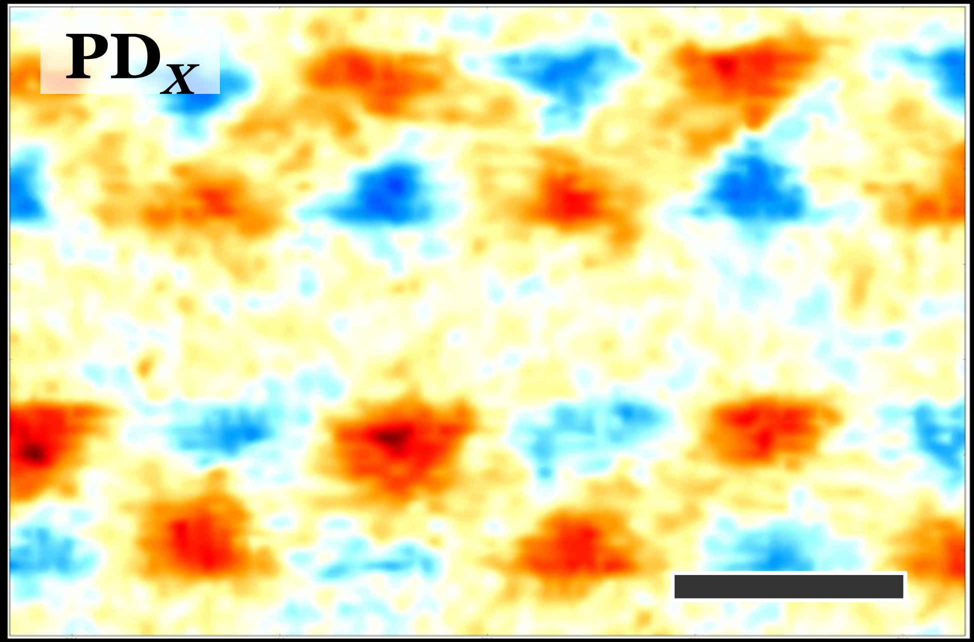
Direct Observation of Atomic Structure



https://www.mpg.de/1318023/electric-data-storage





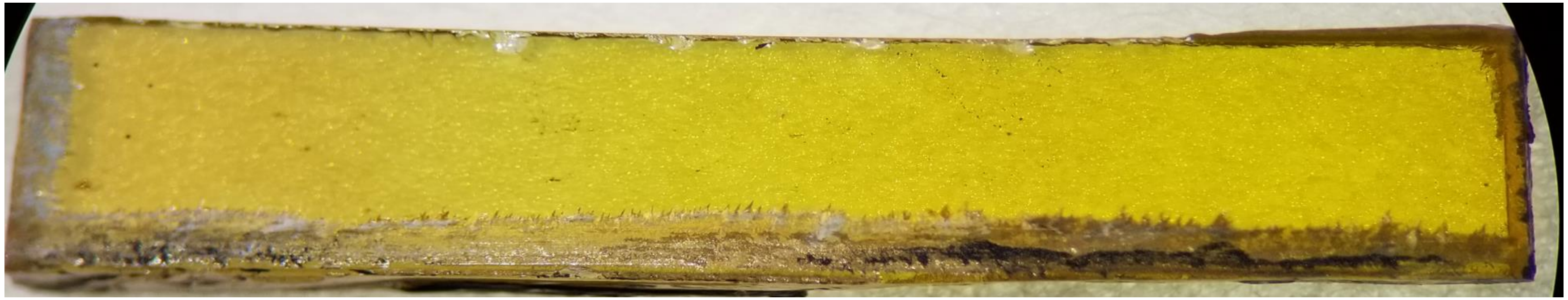


Li et al., Nat. Comm. 2017

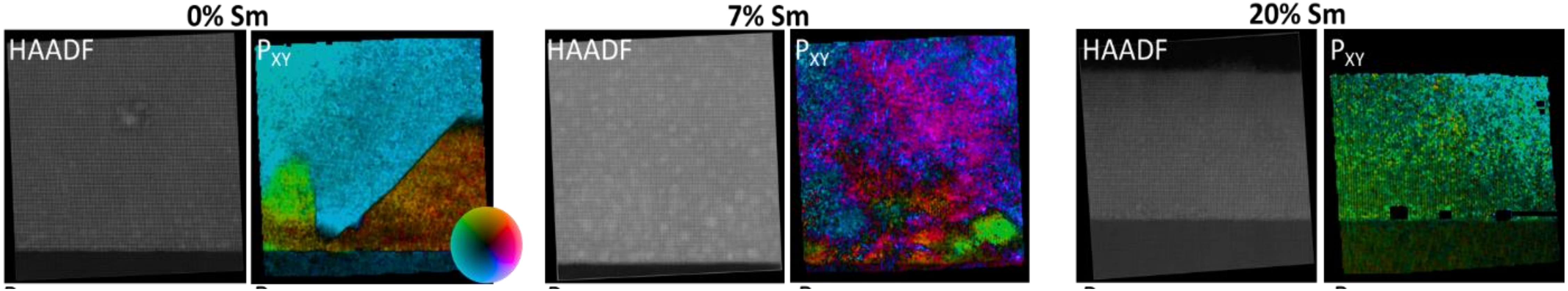
Cause and effect in doped ferroelectrics

For real material, can we establish what is the cause and what is the effect:

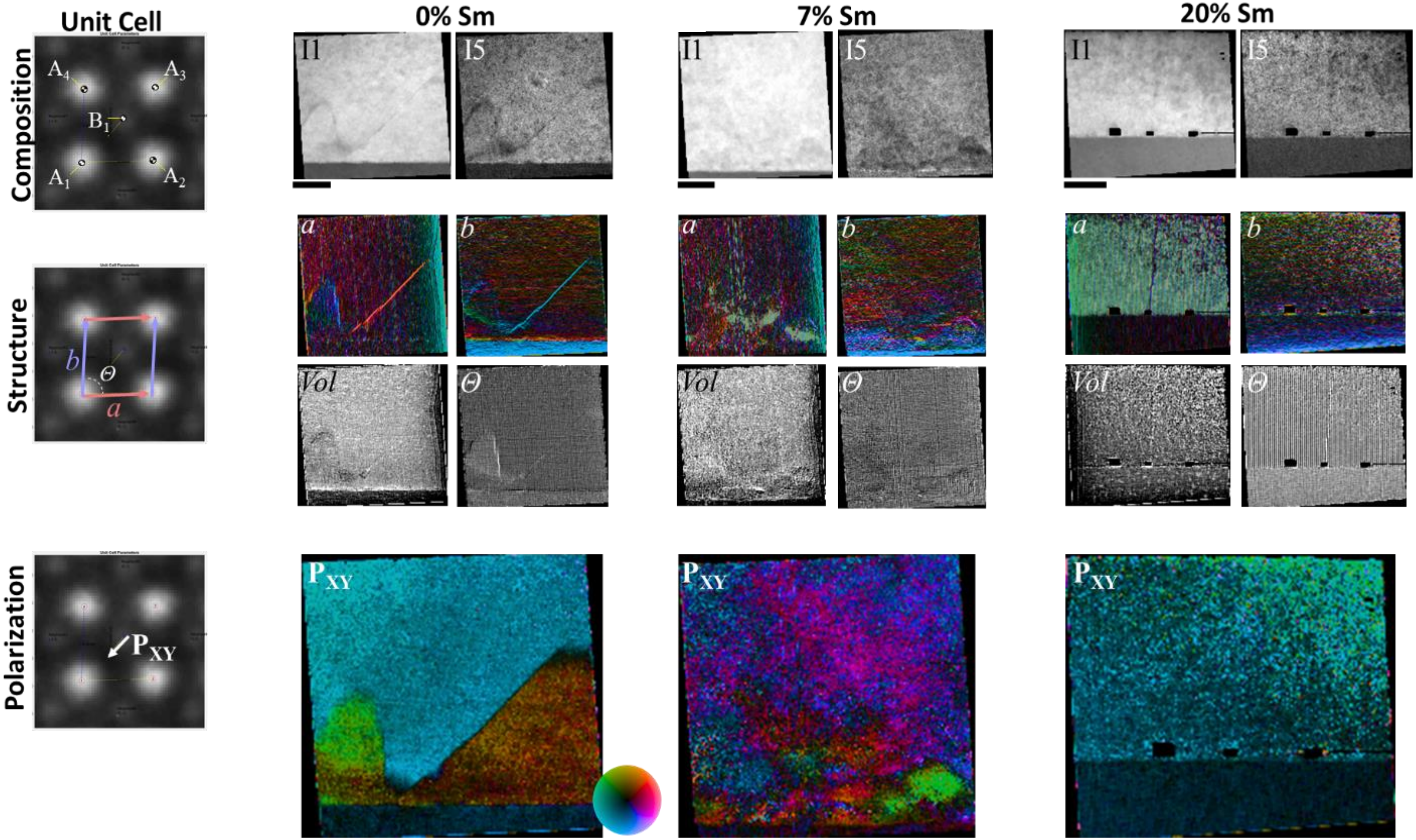
- Does polarization align to the cationic disorder
- Or does polarization instability drive cationic disorder?



20%Sm $BiFeO_3$ Linear est. 7%Sm $BiFeO_3$ $BiFeO_3$



Building descriptor banks



Data-based causal analysis

Step 1: Identify conditional independencies between variables.
Step 2: Gradually remove edges in a fully connected graph to create a skeleton that represents potential causal relationships.
Step 3: Orient edges based on these independencies to form a DAG

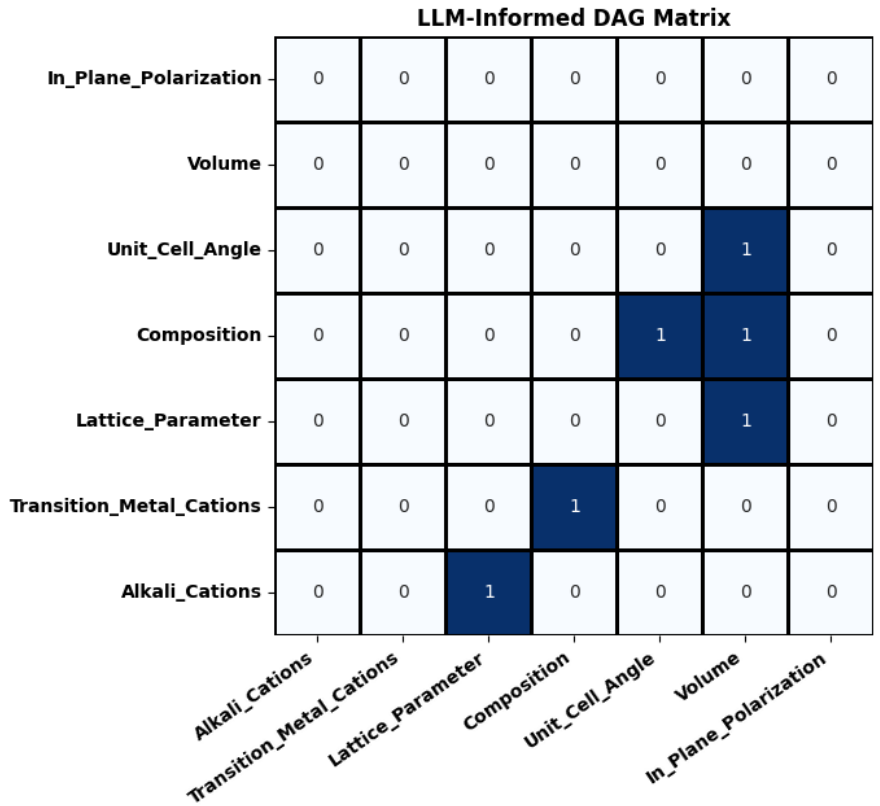
```
for entry in SBFOdata:
print(entry.keys())


def extract_physical_values(entry):
return {
'Alkali_Cations': entry['I1'].flatten(),
'Transition_Metal_Cations': entry['I5'].flatten(),
'Lattice_Parameter': entry['a'].flatten(),
'Composition': entry['Composition'],
'Unit_Cell_Angle': entry['alpha'].flatten(),
'Volume': entry['Vol'].flatten(),
'In_Plane_Polarization': entry['Pxy'][0].flatten(),
}
# PC discovery without LLM assist
pc = PC(variant='stable')
pc.learn(scaled_data)

# Create the inverse mapping from indices to variable names
inverse_var_map = {v: k for k, v in all_vars.items()}

# Replace the indices with variable names in the DAG matrix
labels = [inverse_var_map[i] for i in
range(pc.causal_matrix.shape[0])]
pc_dag_named = pd.DataFrame(pc.causal_matrix, index=labels,
columns=labels)
```

PC-Discovered DAG Matrix without LLM assist

	Alkali_Cations	Transition_Metal_Cations	Lattice_Parameter	Composition	Unit_Cell_Angle	Volume	In_Plane_Polarization
In_Plane_Polarization	1	0	1	1	0	1	0
Volume	1	0	1	1	0	0	0
Unit_Cell_Angle	1	0	1	1	0	1	0
Composition	1	0	0	0	0	0	0
Lattice_Parameter	1	0	0	1	0	0	0
Transition_Metal_Cations	1	0	0	1	0	1	1
Alkali_Cations	0	0	0	1	0	0	0

But what about domain knowledge?

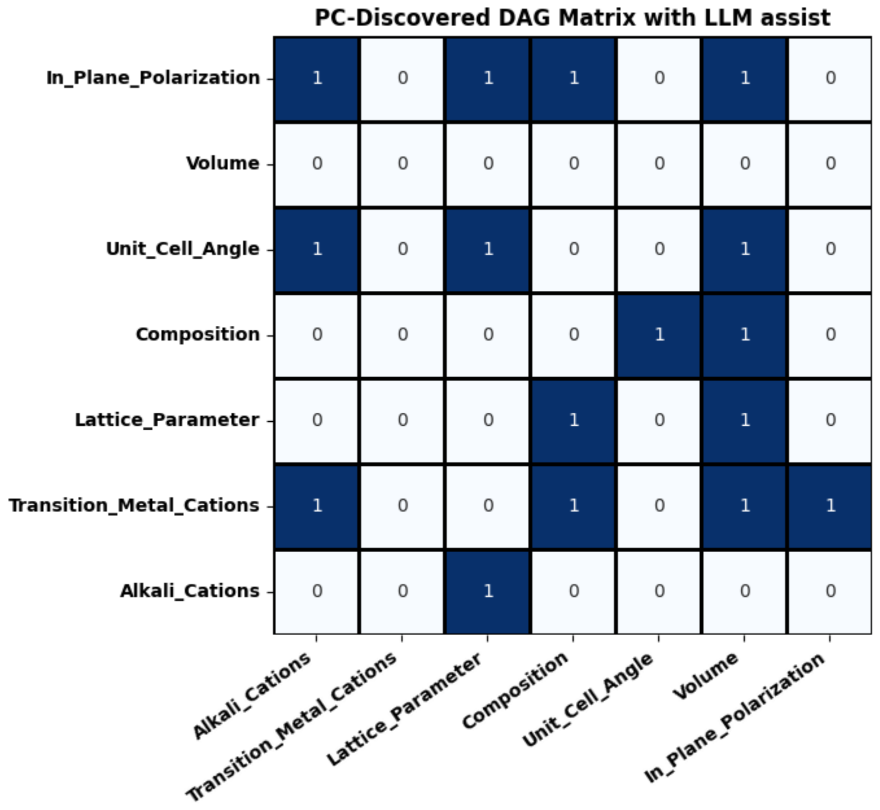
```python
# Instantiate the LLM
llm = ChatOpenAI(temperature=0, model='gpt-4')

# Load tools for LangChain
tools = load_tools(["arxiv"], llm=llm)

# Initialize the agent
agent = initialize_agent(tools, llm,
agent=AgentType.CHAT_ZERO_SHOT_REACT_DESCRIPTION,
handle_parsing_errors=True, verbose=False)

# Define a function to query the LLM for causal relationships
def get_llm_info(llm, agent, var_1, var_2):
out = agent(f"Does {var_1} cause {var_2} or the other way around?
We assume the following definition of causation: if we change A, B
will also change. The relationship does not have to be linear or
monotonic. We are interested in all types of causal relationships,
including partial and indirect relationships, given that our
definition holds.")

pred = llm.predict(f'We assume the following definition of
causation: if we change A, B will also change. Based on the
following information: {out["output"]}, print (0,1) if {var_1}
causes {var_2}, print (1, 0) if {var_2} causes {var_1}, print
(0,0) if there is no causal relationship between {var_1} and
{var_2}. Finally, print (-1, -1) if you don\'t know. Importantly,
don\'t try to make up an answer if you don\'t know.')
return pred

priori_knowledge = PrioriKnowledge(n_nodes=len(all_vars))
# Generate the LLM-informed DAG matrix
priori_dag = np.clip(priori_knowledge.matrix, 0, 1)
```



- Based on the A. Molak https://towardsdatascience.com/jane-the-discoverer-enhancing-causal-discovery-with-large-language-models-causal-python-564a63425c93

Put them all together!

```python
# Re-run PC with Priori Knowledge
pc_priori = PC(priori_knowledge=priori_knowledge,
variant='stable')
pc_priori.learn(scaled_data)

# Replace the indices with variable names in the
DAG matrix
labels_llm_informed = [inverse_var_map[i] for i in
range(pc_priori.causal_matrix.shape[0])]
pc_dag_named_llm_informed =
pd.DataFrame(pc_priori.causal_matrix,
index=labels_llm_informed,
columns=labels_llm_informed)
```



Key message:

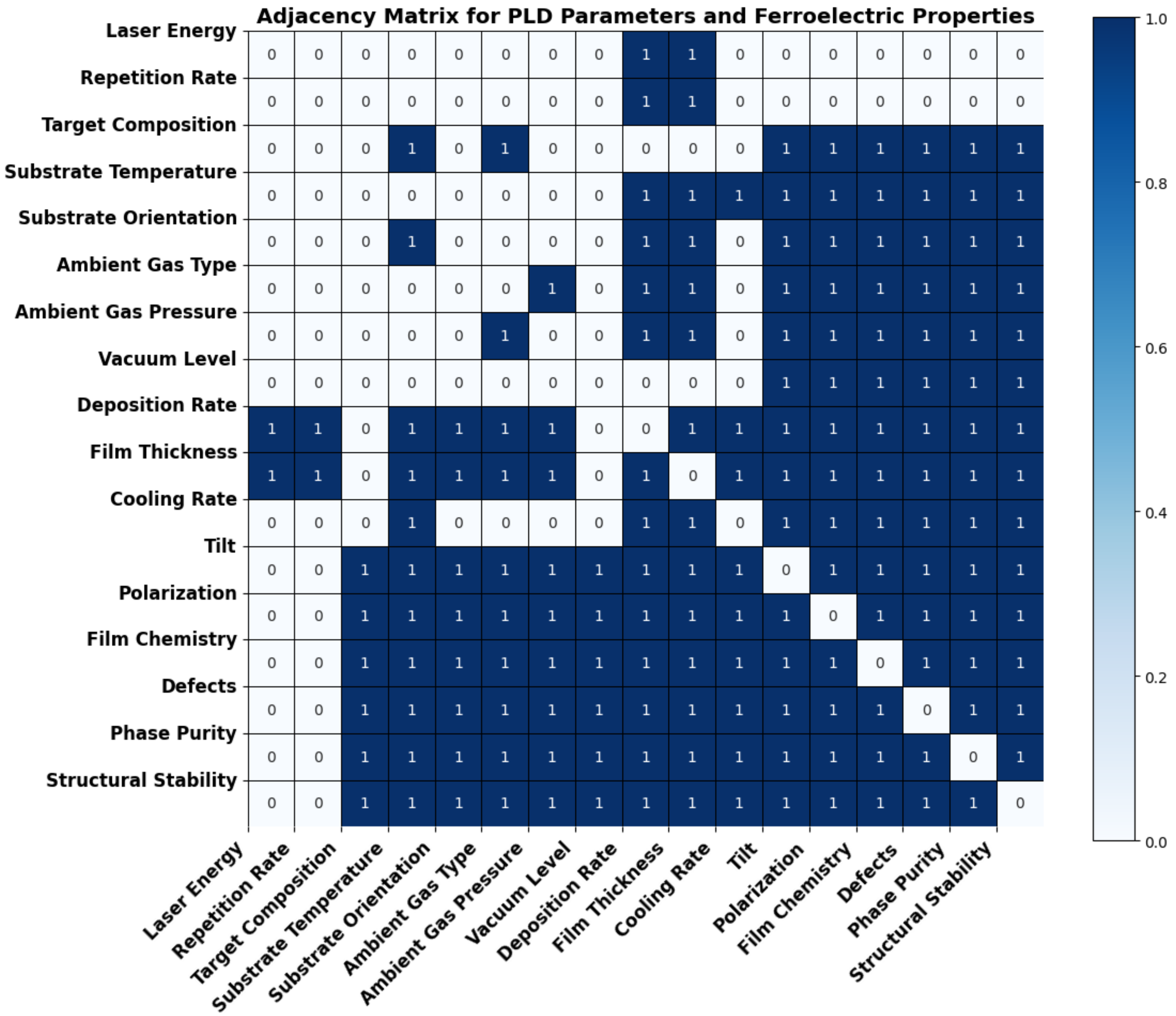
- We need frameworks that allow to constrain LLM outputs to the rigid format compatible with downstream applications. This can be DAGs, knowledge graphs, JSONs, etc.
- We still need to check – by human or another ML agent
- There is (past) knowledge in numbers

What’s next?

- We can perform causal analysis by mining data from sources like arXiv to establish links between film growth parameters and material properties. actionable insights

- It didn’t work too well – but its possible!

- Refining queries (human heuristics), adding physics, and so on!



- Homework 2: Use ChatGPT for paper analysis
- Also possible start for final project!