

Data Provenance, Reproducibility



TECHNISCHE
UNIVERSITÄT
DARMSTADT

Marco Bonneschky, Verena Sieburger

Kontakt:
marco.bonneschky@gmx.de
verena@sieburger.de



Experts call this "hyperparameter tuning". xkcd.com/1838/

„Provenance information describes the origins and the history of data in its life cycle.“₃

Identifies the input-output dependencies and/or records the operation history

Reproducibility in empirical AI research is the ability of an independent research team to produce the same results using the same AI method based on the documentation made by the original research team.⁶

Replication Crisis

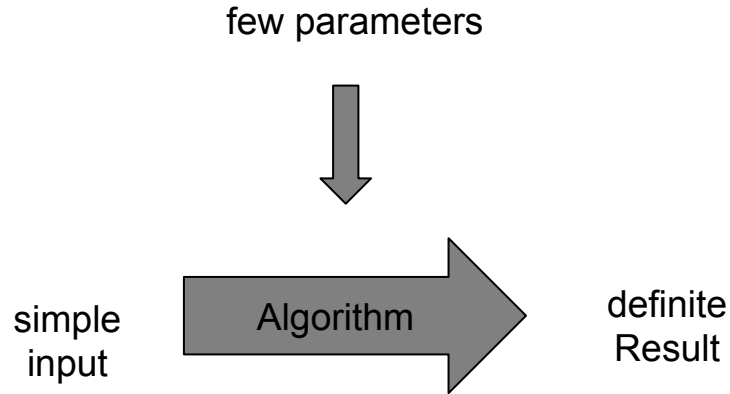
methodological crisis - scientific studies are difficult or impossible to replicate or reproduce

Levels of Reproducibility

Repeatable	• same result can be re-generated within same computational environment, no changes in data/code • verify if experiment is deterministic
Re-runnable	• varied input data but still same result • sign for robust system • original data was representative for the domain
Portable	• re-executable on different platform/environment/libraries
Extendable	• use dataflow/structure and add pre-/postprocessing
Modifiable	• use implementation for reuse • verify correctness through modifiability

Why are AI systems different?

common systems



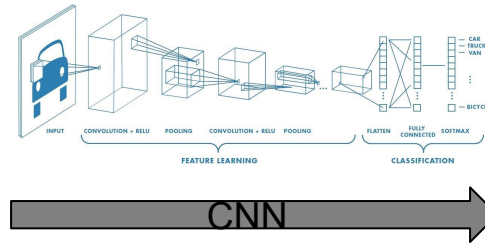
AI systems

AI Image Annotation/Classification

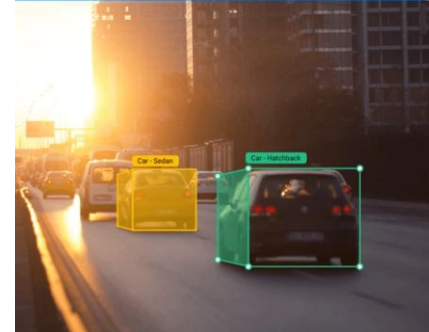
Input



Algorithm



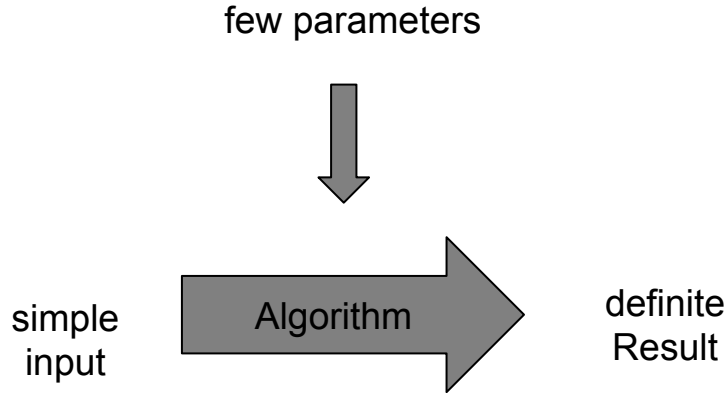
Result



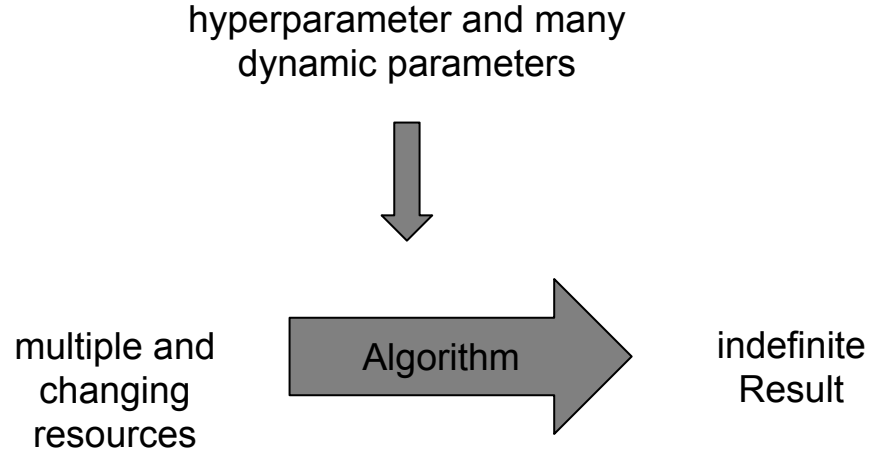
[7],[8],[9]

Why are AI systems different?

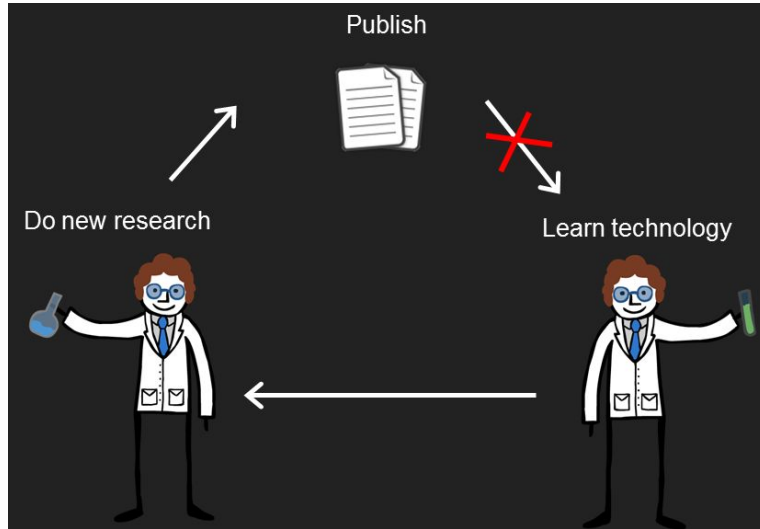
common systems



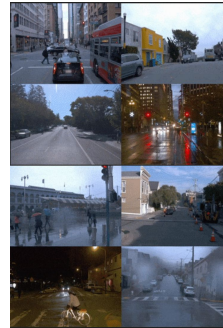
AI systems



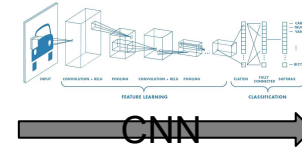
Reproducibility in our example



Input



Algorithm



Result

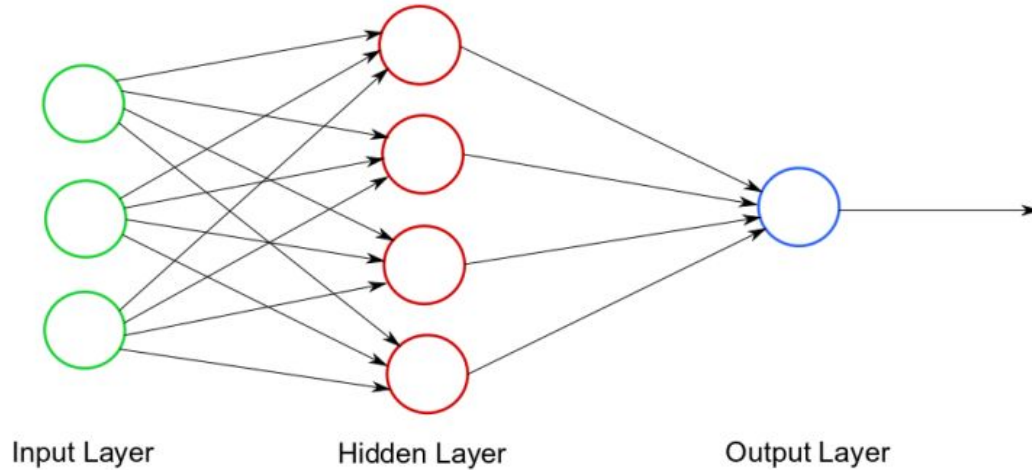


[7],[8],[9],[10]

Data Provenance in our example



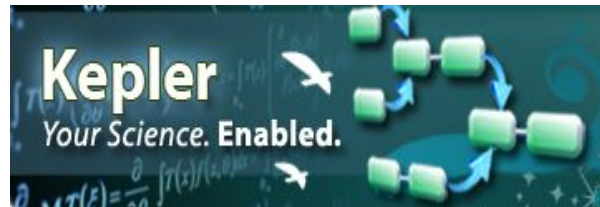
[12],[13]



Logging

- Data
 - Input
 - Output
 - Intermediate
- Features
- Structure
- Hyperparameter

Automatic Provenance Capture

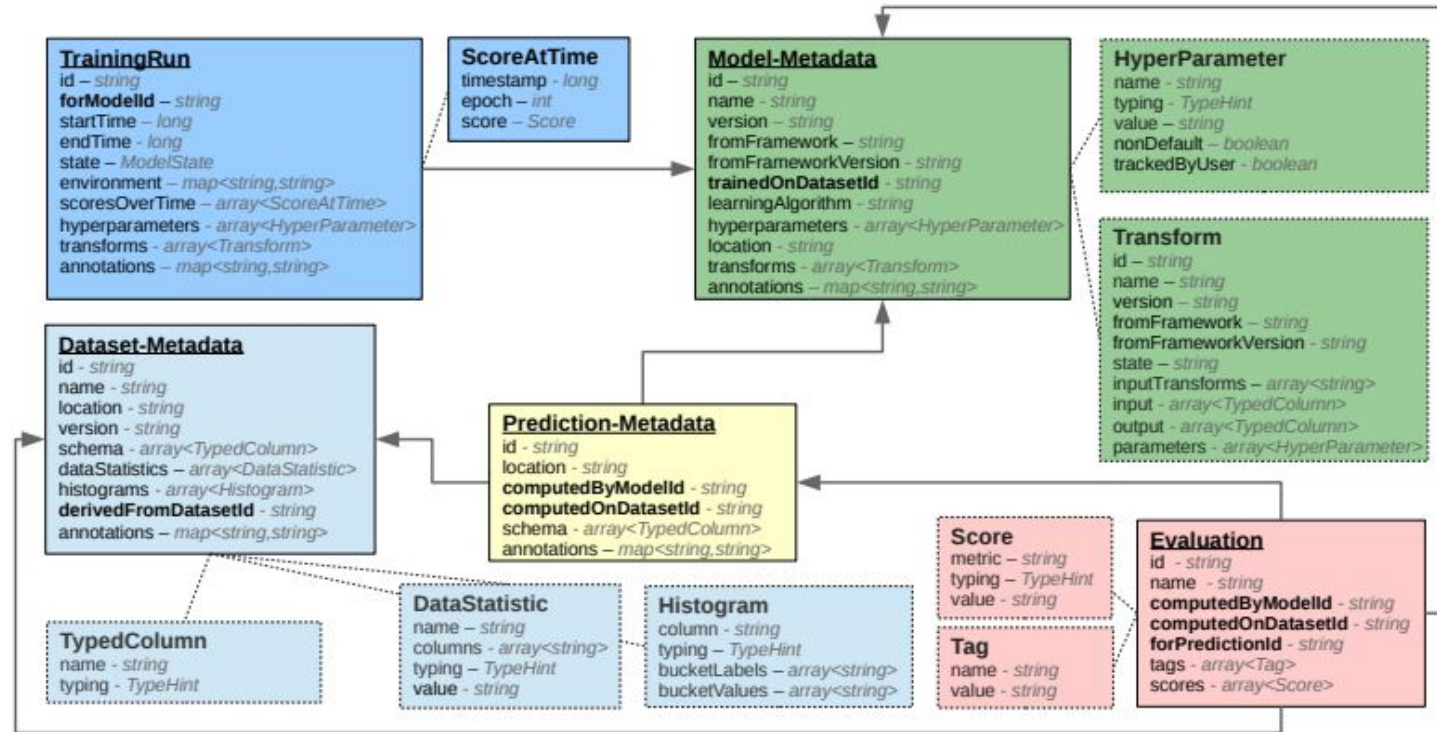


[15],[16],[17],[18],[19]

Capture mode

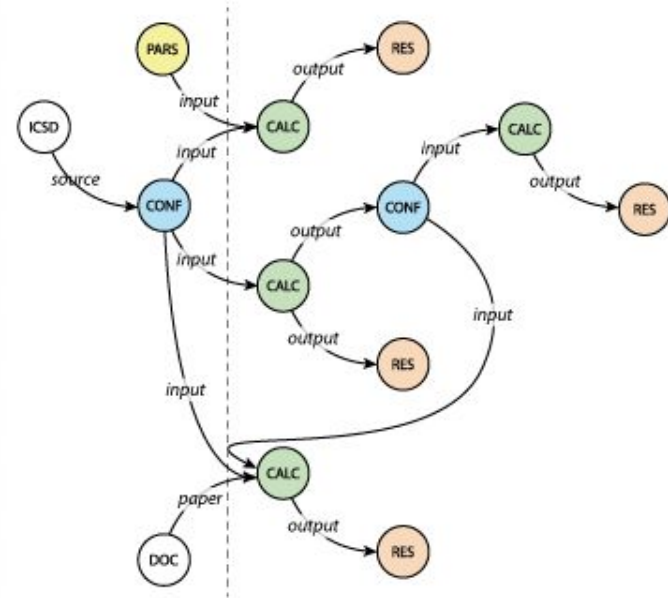
- data oriented
 - goods
- process oriented
 - lipstick
 - kepler

Storing provenance data



Access provenance data

- Graph
 - easy & fast overview
- Query
 - `SELECT image WHERE car.color=red`
- API
 - customizable interfaces



[5], [21], [22]

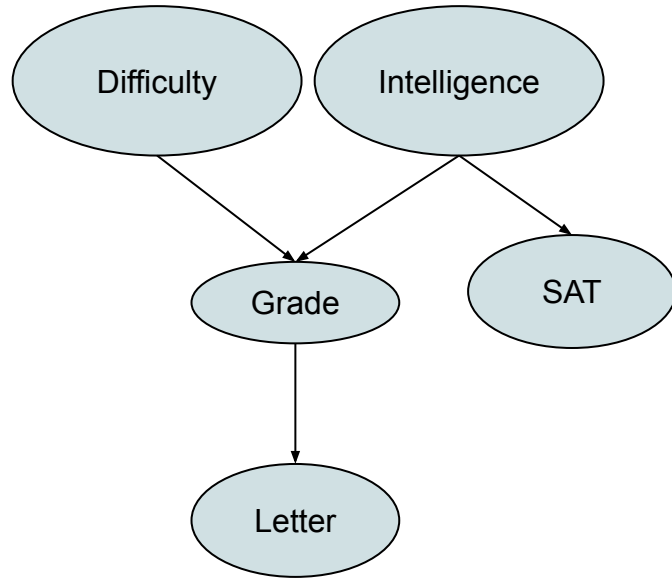
Analyse Data Provenance

- Mitigating Poisoning Attacks
- Crash recovery mechanisms
- Debugging support



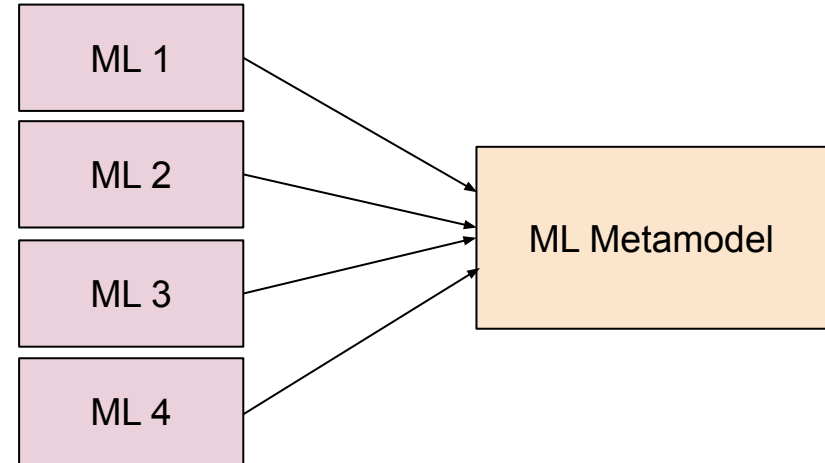
[1],[2],[5],[24]

Data Provenance for Graph Based ML



- LAMP
- using mathematical structure
 - partial derivative
- input-output dependencies
 - quantitative input importance

- data provenance information
 - includes ML1, ML2, ML3, ML4



Challenges of Data Provenance Capture

- number and size of datasets
- variety of data formats
- change of data(sets)
- provenance collection overhead $\longleftrightarrow$ interpretable

```

graph TD
    TrainingRun --> ScoredTime
    TrainingRun --> Model-Metadata
    TrainingRun --> HyperParameter
    TrainingRun --> Transform
    TrainingRun --> Dataset-Metadata
    TrainingRun --> Prediction-Metadata
    TrainingRun --> DataStatistic
    TrainingRun --> Histogram
    TrainingRun --> TypedColumn
    TrainingRun --> Metric
    TrainingRun --> Tag
    TrainingRun --> Evaluation
    ScoredTime --> Model-Metadata
    ScoredTime --> Prediction-Metadata
    Model-Metadata --> HyperParameter
    Model-Metadata --> Transform
    Model-Metadata --> Dataset-Metadata
    Model-Metadata --> Prediction-Metadata
    Model-Metadata --> DataStatistic
    Model-Metadata --> Histogram
    Model-Metadata --> TypedColumn
    Model-Metadata --> Metric
    Model-Metadata --> Tag
    Model-Metadata --> Evaluation
    HyperParameter --> Transform
    HyperParameter --> Dataset-Metadata
    HyperParameter --> Prediction-Metadata
    HyperParameter --> DataStatistic
    HyperParameter --> Histogram
    HyperParameter --> TypedColumn
    HyperParameter --> Metric
    HyperParameter --> Tag
    HyperParameter --> Evaluation
    Transform --> Dataset-Metadata
    Transform --> Prediction-Metadata
    Transform --> DataStatistic
    Transform --> Histogram
    Transform --> TypedColumn
    Transform --> Metric
    Transform --> Tag
    Transform --> Evaluation
    Dataset-Metadata --> Prediction-Metadata
    Dataset-Metadata --> DataStatistic
    Dataset-Metadata --> Histogram
    Dataset-Metadata --> TypedColumn
    Dataset-Metadata --> Metric
    Dataset-Metadata --> Tag
    Dataset-Metadata --> Evaluation
    Prediction-Metadata --> DataStatistic
    Prediction-Metadata --> Histogram
    Prediction-Metadata --> TypedColumn
    Prediction-Metadata --> Metric
    Prediction-Metadata --> Tag
    Prediction-Metadata --> Evaluation
    DataStatistic --> Histogram
    DataStatistic --> TypedColumn
    DataStatistic --> Metric
    DataStatistic --> Tag
    DataStatistic --> Evaluation
    Histogram --> TypedColumn
    Histogram --> Metric
    Histogram --> Tag
    Histogram --> Evaluation
    TypedColumn --> Metric
    TypedColumn --> Tag
    TypedColumn --> Evaluation
    Metric --> Tag
    Metric --> Evaluation
    Tag --> Evaluation
  
```

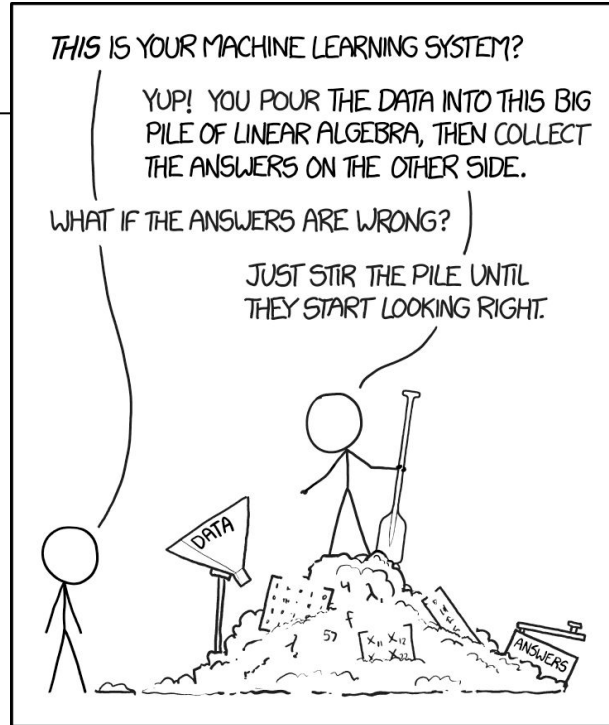
The diagram illustrates the relationships between various MLflow metadata objects. The objects and their fields are as follows:

- TrainingRun**
 - id - string
 - artifactUri - string
 - startTime - long
 - endTime - long
 - state - ModelState
 - transform - map-string-string
 - scoresOverTime - array-ScoredTime
 - hyperParameters - array-HyperParameter
 - transforms - array-Transform
 - annotations - map-string-string
- ScoredTime**
 - timeStep - long
 - epoch - int
 - score - Score
- Model-Metadata**
 - id - string
 - name - string
 - framework - string
 - frameworkVersion - string
 - dataSet - string
 - learningAlgorithm - string
 - hyperParameters - array-HyperParameter
 - transforms - array-Transform
 - annotations - map-string-string
- HyperParameter**
 - name - string
 - type - string
 - value - string
 - required - boolean
 - trackedUser - boolean
- Transform**
 - id - string
 - name - string
 - version - string
 - framework - string
 - frameworkVersion - string
 - inputTargets - array-string
 - input - array-TypeColumn
 - output - array-TypeColumn
 - parameters - array-HyperParameter
- Dataset-Metadata**
 - id - string
 - name - string
 - location - string
 - version - string
 - dataStatistic - array-DataStatistic
 - histograms - array-Histogram
 - typedColumns - array-TypedColumn
 - annotations - map-string-string
- Prediction-Metadata**
 - id - string
 - location - string
 - computedModelId - string
 - computedDataSetId - string
 - schema - array-TypeColumn
 - annotations - map-string-string
- DataStatistic**
 - name - string
 - columns - array-string-string
 - type - TypeRef
 - value - string
- Histogram**
 - columns - string
 - type - TypeRef
 - bucketLabels - array-string
 - bucketValue - array-string
- TypedColumn**
 - name - string
 - type - TypeRef
- Metric**
 - score - string
 - type - TypeRef
 - value - string
- Tag**
 - name - string
 - value - string
- Evaluation**
 - id - string
 - name - string
 - computedModelId - string
 - computedDataSetId - string
 - forPrediction - string
 - tags - array-Tag
 - score - array-Score

Arrows indicate dependencies or relationships between the fields of these objects. For example, TrainingRun depends on many other objects, while Evaluation depends on several others.

02.07.20 | Fachbereich 20 | Software Engineering for Artificial Intelligence | 19

Summary



Experts call this "hyperparameter tuning". xkcd.com/1838/

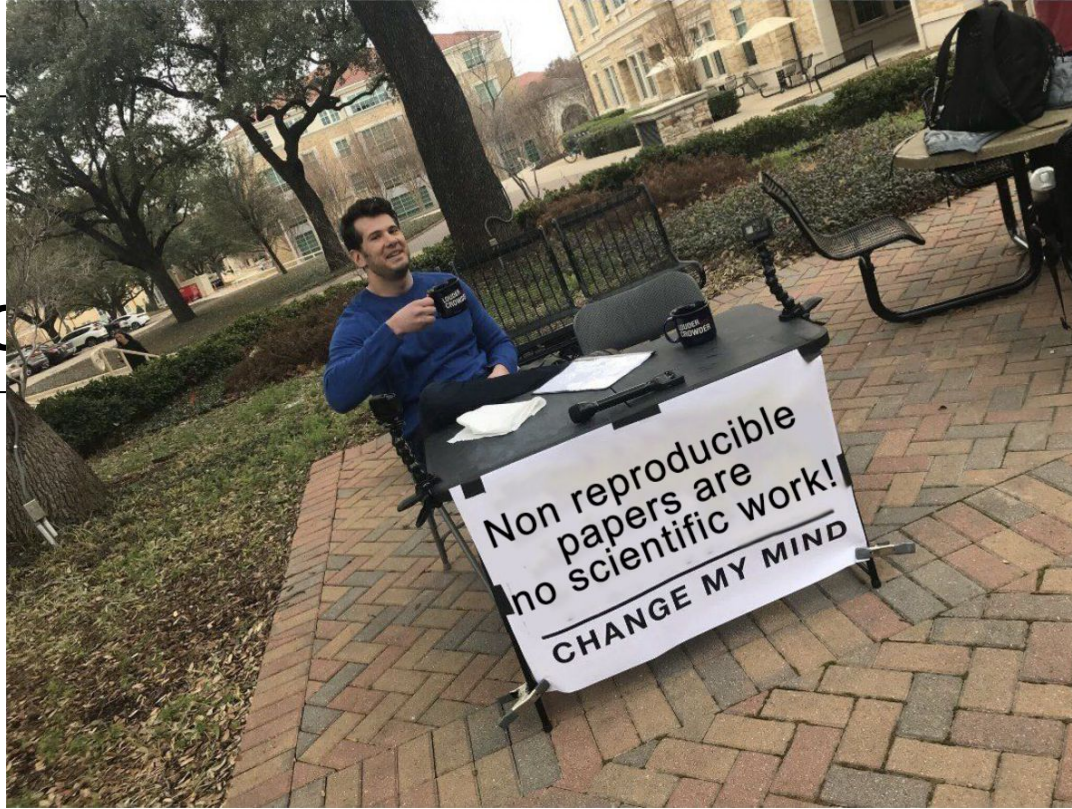
Sources

1. <https://dl.acm.org/doi/pdf/10.1145/3106237.3106291>
2. <https://dl.acm.org/doi/abs/10.1145/3128572.3140450>
3. <http://homepages.inf.ed.ac.uk/jcheney/publications/provdbsurvey.pdf>
4. <https://ebookcentral.proquest.com/lib/ulbdarmstadt/reader.action?docID=5357977&ppg=137>
5. <http://static.googleusercontent.com/media/research.google.com/en//pubs/archive/45390.pdf>
6. <https://www.aaai.org/ocs/index.php/AAAI/AAAI18/paper/viewFile/17248/15864>
7. <https://techcrunch.com/2019/08/21/waymo-releases-a-self-driving-open-data-set-for-free-use-by-the-research-community>
8. <https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53>
9. <https://venturebeat.com/2018/11/16/hive-taps-a-workforce-of-700000-people-to-label-data-and-train-ai-models/>
10. <https://www.jove.com/blog/scientist-blog/data-vs-methods-why-science-articles-are-so-difficult-to-reproduce/>
11. <http://sites.computer.org/debull/A18mar/p15.pdf>
12. <https://de.mathworks.com/solutions/deep-learning/convolutional-neural-network.html>
13. <https://www.polygons.tech/image-annotation/annotation-for-self-driving-car-adas/>
14. <https://www.kdnuggets.com/2017/10/neural-network-foundations-explained-gradient-descent.html>
15. <https://valohai.com/>
16. <https://netflixtechblog.com/introducing-lipstick-on-a-pache-pig-f17e0a4e0c89>
17. <https://zeenea.com/google-goods-the-management-and-data-democratization-tool-of-google/>
18. https://www.vistrails.org/index.php/Main_Page
19. <https://kepler-project.org/users/features.html>
20. http://learningsys.org/nips17/assets/papers/paper_13.pdf
21. <https://sigmodrecord.org/publications/sigmodRecord/0509/p31-special-sw-section-5.pdf>
22. <https://arxiv.org/pdf/1910.04223.pdf>
23. <http://www.aiida.net/feature/data-provenance/>
24. https://dev.to/molly_struve/10-tips-for-debugging-in-production-ko1

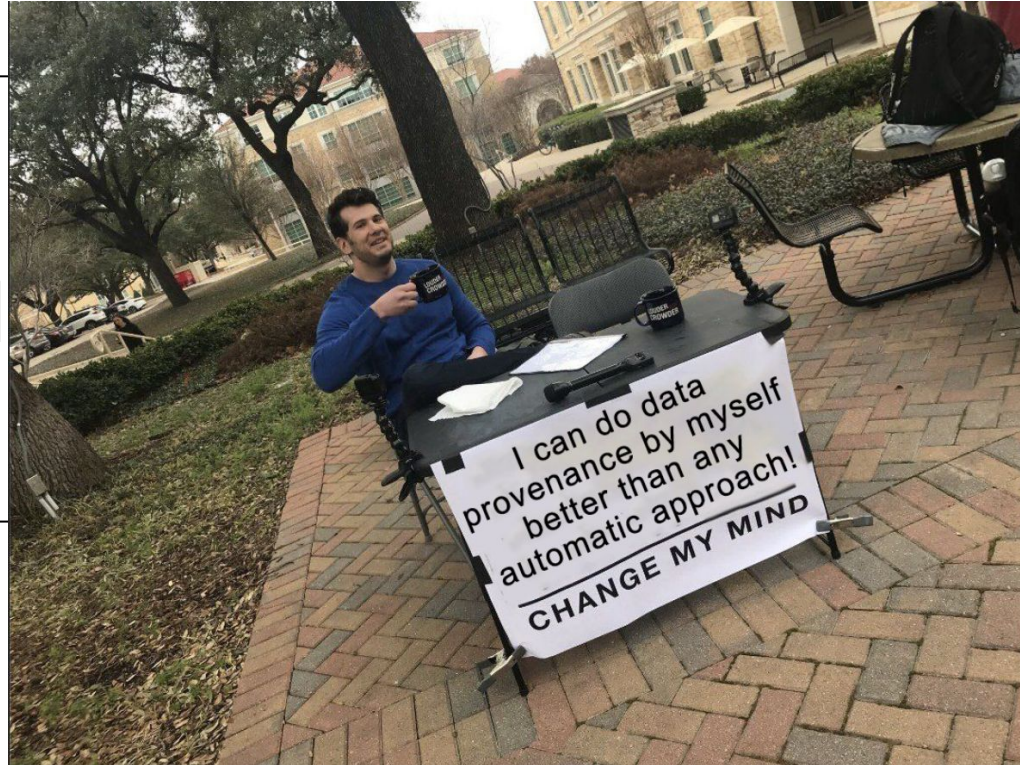
Questions

Non repro

tific work!



I can do d



better than

Backup-slides following

Provenance Why, How, and Where - DB notion

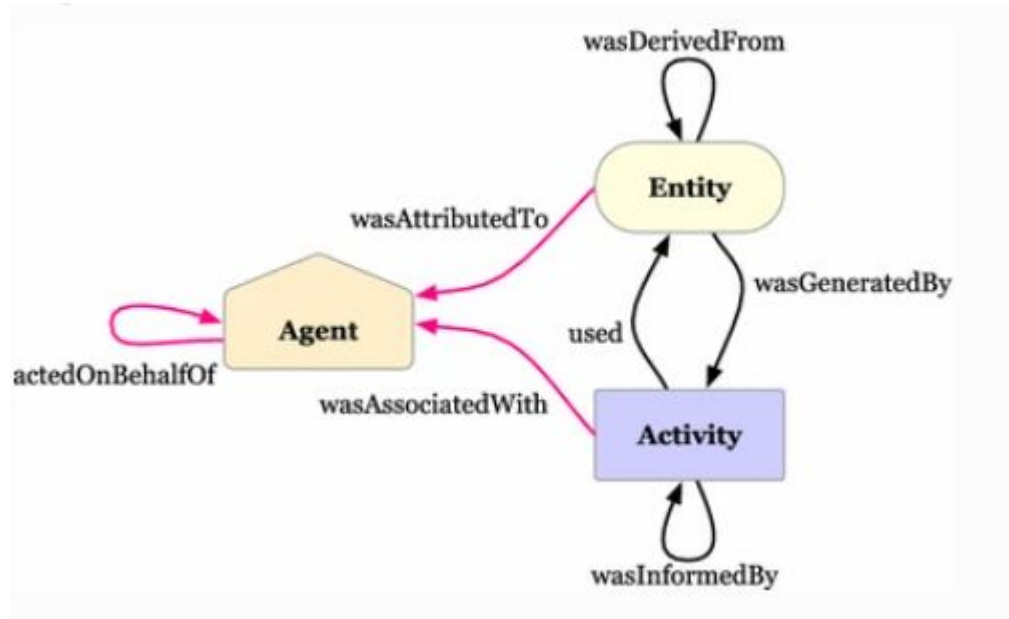
Why inputs that explain why an output record was produced

How describing in detail how an output was produced

Where output data came from the input

J. Cheney, L. Chiticariu, and W.-C. Tan. Provenance in databases: Why, how, and where. Found. Trends databases, 1(4):379–474, Apr. 2009

W3C provenance (PROV)



Acknowledgements & License

- Material Design Icons, by Google under [Apache-2.0](#)
- Other images are either by the authors of these slides or attributed where they are used
- These slides are made available by the authors (Verena Sieburger, Marco Bonneschky) under [CC BY 4.0](#)