



INDE lab

INtelligent Data Engineering

indelab.org | @INDE_LAB_AMS

Intro - Dutch-Belgian Database Day December 7, 2021

The INDE lab Team



Prof. Paul Groth



Dr. Frank Nack



Dr. Jacobijn Sandberg



Thiviyan
Thanapalasingam



Daniel Daza



Madelon Hulsebos

Olivier Sprangers



Dr. Sebastian Schelter



Dr. Sara Magliacane



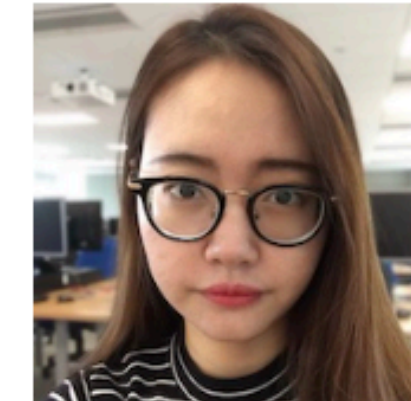
Fan Feng



Corey Harper



Melika Ayoughi



Effy Xue Li

Shubha Guha



Stian Soiland-Reyes



James Nevin



Stefan Grafberger



Dr. Peter Bloem

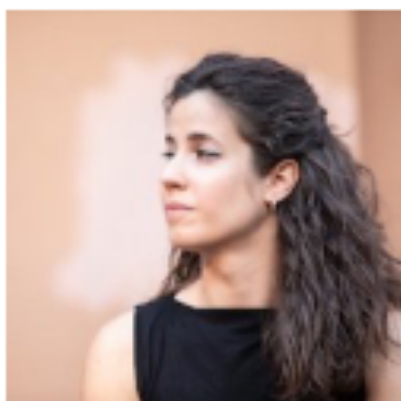


Dr. Hartmut Koenitz



Dr. Stefan Schlobach

Barrie Kersbergen



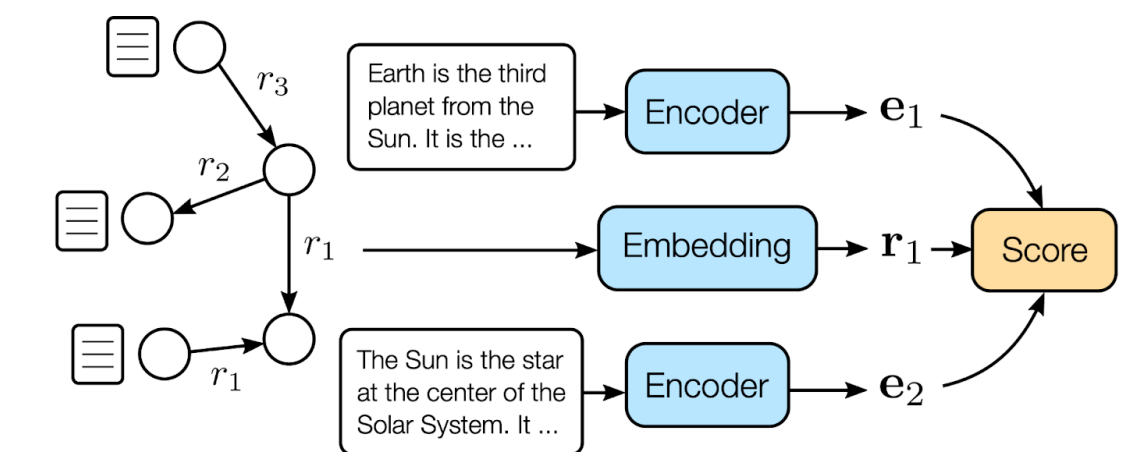
Valentina Carriero

INDELab started Nov. 5, 2018

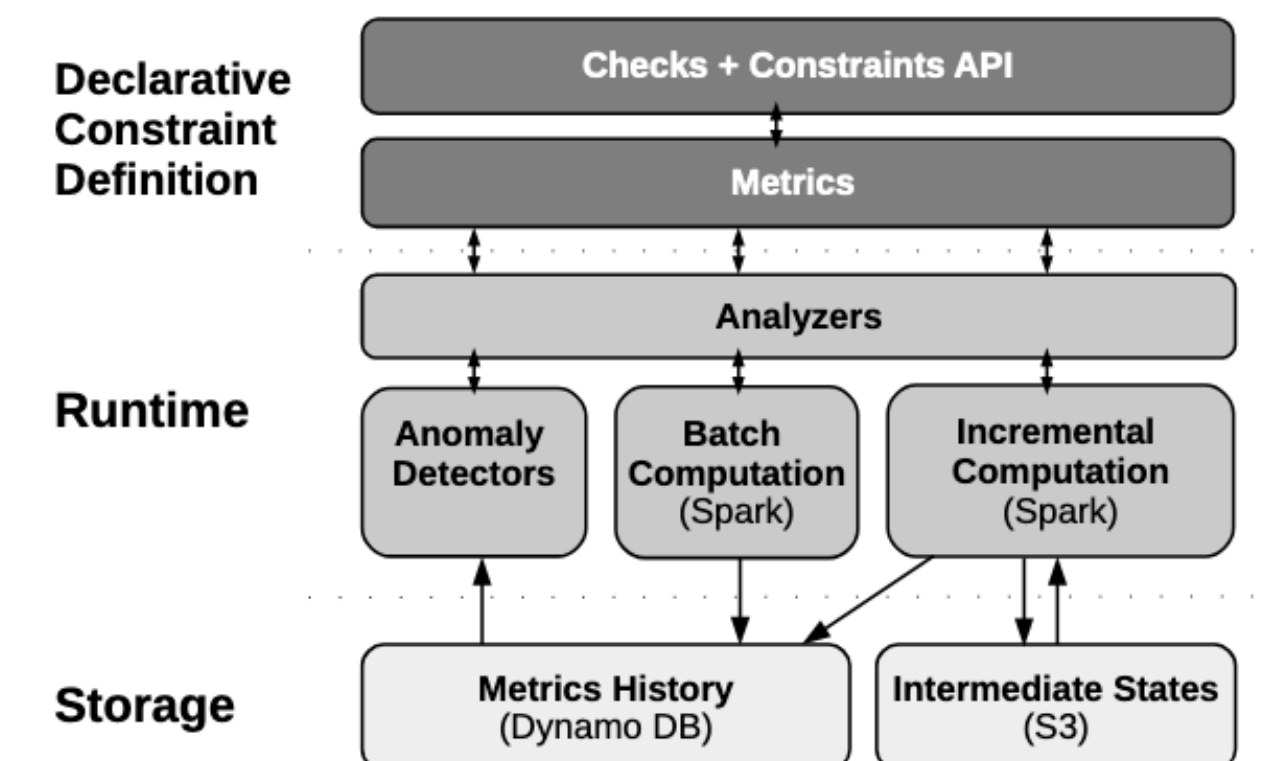
Research Topics at INDE lab

- Design **systems to support people in working with data** from diverse sources
- Address problems related to the **preparation, management, and integration of data**

- **Automated Knowledge Graph Construction**
(e.g., predicting and adding new links in datasets such as Wikidata based on text)
- **Data Search & Reuse**
(e.g., studies on GitHub hosted data; algorithms for making data FAIR)
- **Data Management for Machine Learning**
(e.g., scalable concept drift detection for ML training data, integrated in AWS SageMaker Model Monitor; using data provenance for ML debugging)
- **Causality-Inspired Machine Learning** (e.g., using ideas from causal inference to improve the robustness and generalization of ML algorithms, especially in cases of distribution shift; domain adaptation)



Link prediction



Highlights for the DBDBD audience

About GitTables gittables.github.io

GitTables is a dataset of currently 1.7M relational tables extracted from CSV files in GitHub. Our continuing curation aims at growing the dataset to at least 20M tables. Table columns in GitTables have been annotated with more than 2K different semantic types from Schema.org and DBpedia. Our column annotations consist of semantic types, hierarchical relations, range types and descriptions.

The high-level pipeline in Figure 1 illustrates how GitTables was created.

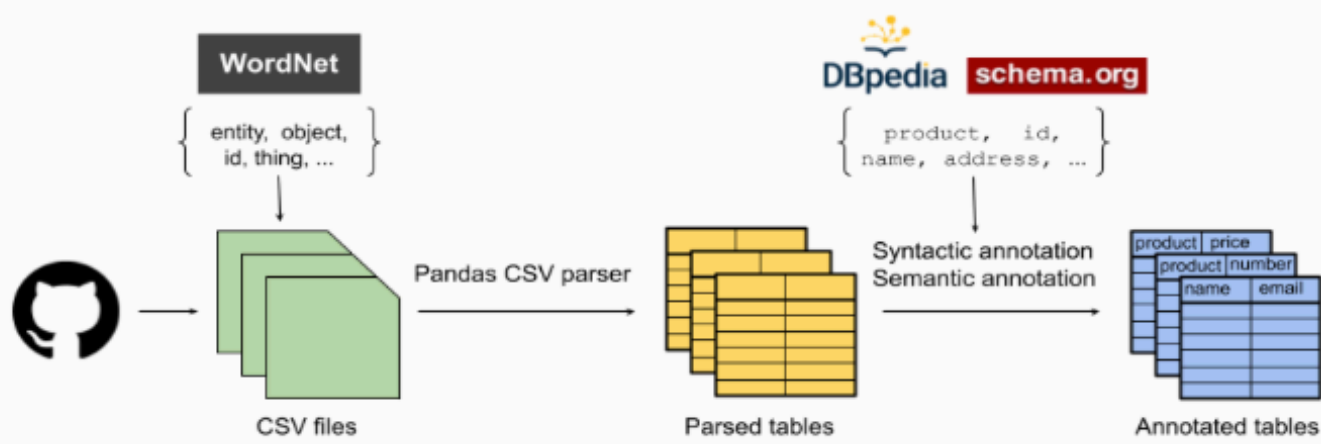


Figure 1: high-level pipeline of the process of constructing GitTables.

mlinspect <https://github.com/stefan-grafberger/mlinspect>

Potential issues in preprocessing pipeline:

- 1 Join might change proportions of groups in data
- 2 Column 'age_group' projected out, but required for fairness
- 3 Selection might change proportions of groups in data
- 4 Imputation might change proportions of groups in data
- 5 'race' as a feature might be illegal!
- 6 Embedding vectors may not be available for rare names!

Python script for preprocessing, written exclusively with native pandas and sklearn constructs

```
# load input data sources, join to single table
patients = pandas.read_csv(...)
histories = pandas.read_csv(...)
data = pandas.merge([patients, histories], on=['ssn'])

# compute mean complications per age group, append as column
complications = data.groupby('age_group')
    .agg(mean_complications=('complications', 'mean'))
data = data.merge(complications, on=['age_group'])

# Target variable: people with frequent complications
data['label'] = data['complications'] > 1.2 * data['mean_complications']

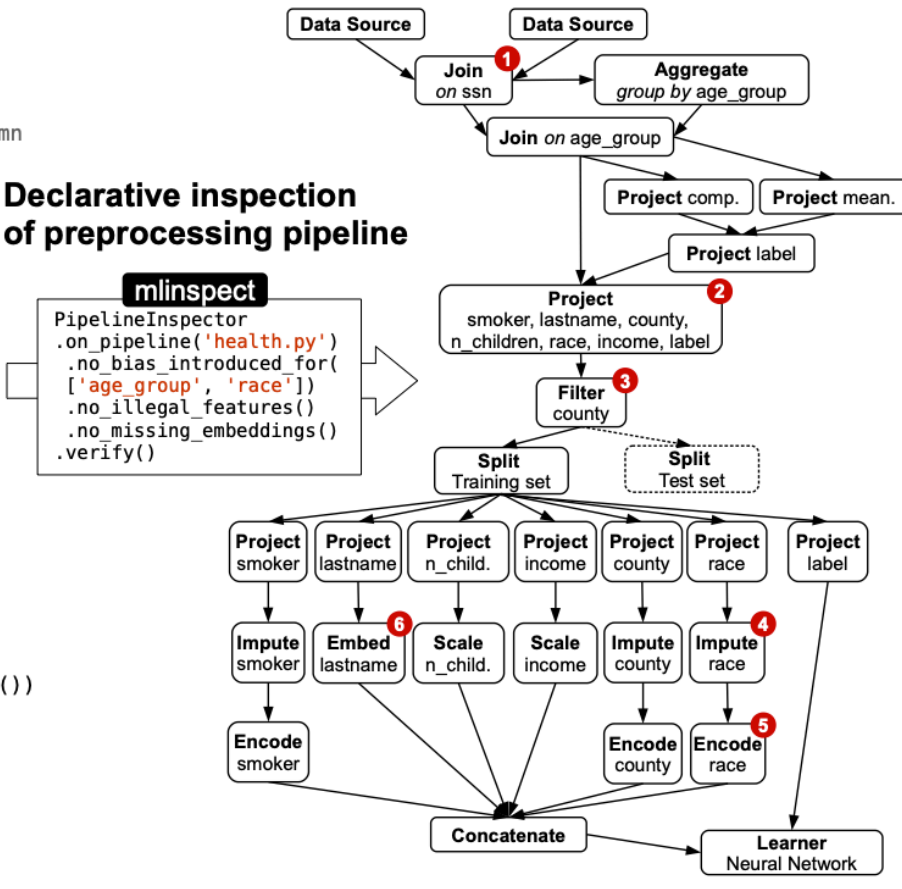
# Project data to subset of attributes, filter by counties
data = data[['smoker', 'last_name', 'county',
            'num_children', 'race', 'income', 'label']]
data = data[data['county'].isin(counties_of_interest)]

# Define a nested feature encoding pipeline for the data
impute_and_encode = sklearn.Pipeline([
    (sklearn.SimpleImputer(strategy='most_frequent')),
    (sklearn.OneHotEncoder())])
featurisation = sklearn.ColumnTransformer(transformers=[
    (impute_and_encode, ['smoker', 'county', 'race']),
    (Word2VecTransformer(), 'last_name')
    (sklearn.StandardScaler(), ['num_children', 'income'])])

# Define the training pipeline for the model
neural_net = sklearn.KerasClassifier(build_fn=create_model())
pipeline = sklearn.Pipeline([
    ('features', featurisation),
    ('learning_algorithm', neural_net)])

# Train-test split, model training and evaluation
train_data, test_data = train_test_split(data)
model = pipeline.fit(train_data, train_data.label)
print(model.score(test_data, test_data.label))
```

Corresponding dataflow DAG for instrumentation, extracted by mlinspect



Declarative inspection of preprocessing pipeline

```
mlinspect
PipelineInspector
.on_pipeline('health.py')
.no_bias_introduced_for(
    ['age_group', 'race'])
.no_illegal_features()
.no_missing_embeddings()
.verify()
```

Analysis of how data reuse and search in the wild

Harvard Data Science Review

Lost or Found? Discovering Data Needed for Research

Kathleen Gregory¹, Paul Groth², Andrea Scharnhorst¹, Sally Wyatt³

¹Data Archiving and Networked Services, Royal Netherlands Academy of Arts and Sciences,
²Informatics Institute, University of Amsterdam,
³Faculty of Arts and Social Sciences, Maastricht University

Published on: Apr 30, 2020

Updated on: May 07, 2020

DOI: 10.1162/99608f92.e38165eb

Patterns

Dataset Reuse: Toward Translating Principles to Practice

Laura Koesten¹, Pavlos Vougloukis², Elena Simperl¹, and Paul Groth^{1,3*}

¹King's College London, London WC2R 4BG, UK
²Huawei Technologies, Edinburgh EH9 3BF, UK
³University of Amsterdam, Amsterdam 1090 GH, the Netherlands

*Lead Contact
^{*}Correspondence: laura.koesten@kcl.ac.uk (L.K.), p.groth@uva.nl (P.G.)
<https://doi.org/10.1016/j.patter.2020.100138>

THE BIGGER PICTURE The web provides access to millions of datasets. These data can have additional impact when it is used beyond the context for which it was originally created. We have little empirical insight into what makes a dataset more reusable than others, and which of the existing guidelines and frameworks, if any, make a difference. In this paper, we explore potential reuse features through a literature review and present a case study on datasets on GitHub, a popular open platform for sharing code and data. We describe a corpus of more than 1.4 million data files, from over 65,000 repositories. Using GitHub's engagement metrics as proxies for dataset reuse, we relate them to reuse features from the literature and devise an initial model, using deep neural networks, to predict a dataset's reusability. This work demonstrates the practical gap between principles and actionable insights that allow data publishers and tools designers to implement functionalities that provably facilitate reuse.

1 2 3 4 6 Proof-of-Concept: Data science output has been formulated, implemented, and tested for one domain/problem

SUMMARY

The web provides access to millions of datasets that can have additional impact when used beyond their original context. We have little empirical insight into what makes a dataset more reusable than others and which of the existing guidelines and frameworks, if any, make a difference. In this paper, we explore potential reuse features through a literature review and present a case study on datasets on GitHub, a popular open platform for sharing code and data. We describe a corpus of more than 1.4 million data files, from over 65,000 repositories. Using GitHub's engagement metrics as proxies for dataset reuse, we relate them to reuse features from the literature and devise an initial model, using deep neural networks, to predict a dataset's reusability. This demonstrates the practical gap between principles and actionable insights that allow data publishers and tools designers to implement functionalities that provably facilitate reuse.

1 INTRODUCTION

There has been a gradual shift in the last years from viewing datasets as byproducts of digital work to critical assets, whose value increases the more they are used.¹ However, our understanding of how this value emerges, and of the factors that demonstrably affect the reusability of a dataset is still limited. Using a dataset beyond the context where it originated remains challenging for a variety of socio-technical reasons, which have been discussed in the literature.^{2,3} The bottom line is that simply making data available, even when complying with existing guidance and best practices, does not mean it can be easily used by others.⁴

At the same time, making data reusable to a diverse audience, in terms of domain, skill sets, and purposes, is an important way to realize its potential value and recover some of the, sometimes considerable, resources invested in policy and infrastructure support.⁵ This is one of the reasons why scientific journals and research-funding organizations are increasingly calling for further data sharing⁶ or why industry bodies, such as the International Data Spaces Association (IDSA) (<https://www.internationaldataspaces.org>) are investing in reference architectures to smooth data flows from one business to another. There is plenty of advice on how to make data easier to reuse, including technical standards, legal frameworks, and guidelines. Much work places focus on machine readability

harperco / MeasEval

<> Code Issues Pull requests Actions Projects Wiki Security Insights

main 2 branches 0 tags Go to file Add file Code

Corey Harper Adding text files for evaluation phase 98a0210 2 days ago 22 commits

annotationGuidelines	update to guidelines; adding IsMeanHasSD to mod validation	26 days ago
assets	Initial commit readme and image asset	6 months ago
data	Adding text files for evaluation phase	2 days ago
eval	Add validate only option -v to eval; fix eval to support empty submi...	3 days ago
README.md	Fixing typos, helping readability	last month

README.md

Welcome to MeasEval: Counts and Measurements!

Counts and measurements are an important part of scientific discourse. It is relatively easy to find measurements in text, but a bare measurement like 17 mg is not informative. However, relatively little attention has been given to parsing and extracting these important semantic relations. This is challenging because the way scientists write can be ambiguous and inconsistent, and the location of this information relative to the measurement can vary greatly.

MeasEval is a new entity and semantic relation extraction task focused on finding counts and measurements, attributes of these quantities, and additional information including measured entities, properties, and measurement contexts.

For more details and to participate, head over to our CodaLab pages:
<https://competitions.codalab.org/competitions/25770>

FYI:

- Hiring: assistant prof. in data management methodologies
- Special Issue JWS on Knowledge Engineering for large scale knowledge graphs