Classification of Machine Learning Reproducibility Factors

Bachelor Thesis Presentation

Martin Johannes Loos

University of Passau
Faculty of Computer Science and Mathematics
Chair of Data Science

13.06.2022

My thesis in a nutshell.

UNIVERSITÄT PASSAU

- Reproducibility is important.
- Manual tasks are resource-intensive, non-standard, and non-scalable.
- Solution: A tool that checks an ML repository for compliance with the reproducibility guidelines.

1. Motivation
2. Related Work
3. Research Questions & Hypothesis
4. Identification & Classification
5. Implementation
6. Evaluation
7. Discussion
8. Conclusion

UNIVERSITÄT PASSAU

Motivation

Reproducibility Crisis?

A 2016 Nature's survey where 1,576 researchers participated found that:

"*More than 70% of researchers have tried and failed to reproduce another scientist's experiments, and more than half have failed to reproduce their own experiments.*"[1]

[1] Monya Baker. "1,500 scientists lift the lid on reproducibility." In: *Nature* 533.7604 (2016).

Motivation

In a report from the NeurIPS 2019 reproducibility program the authors noted that the *"Standardization of such tools would [...] improve ease of reproducibility."*[2]

[2] Joelle Pineau et al. "Improving reproducibility in machine learning research (a report from the neurips 2019 reproducibility program)." In: *arXiv preprint arXiv:2003.12206* (2020).

UNIVERSITÄT PASSAU

Related Work

Related Work

UNIVERSITÄT PASSAU

- Solid scientific basis from various disciplines on the identification of reproducibility factors.
- Various papers on the conceptual delimitation of reproducibility but none for classification.
- Estimating the likelihood of replicability *of a paper* using an ML model.[3]
- State-of-the-Art: Manual reviewing process

[3] https://www.pnas.org/doi/10.1073/pnas.1909046117 accessed: 07.06.2022

UNIVERSITÄT PASSAU

Research Questions & Hypothesis

Research Questions

UNIVERSITÄT PASSAU

- **RQ1:** Which factors influence the reproducibility of ML experiments?
- **RQ2:** Which influencing factors can be detected with the help of a software tool and which can not?
- **RQ3:** How can the detectable factors be analyzed using a software tool?

Hypothesis

UNIVERSITÄT PASSAU

A repository belonging to an ML experiment published by academic publishers can be expected to adhere to the reproducibility guidelines significantly better than an ML repository on GitHub (on average).

UNIVERSITÄT PASSAU

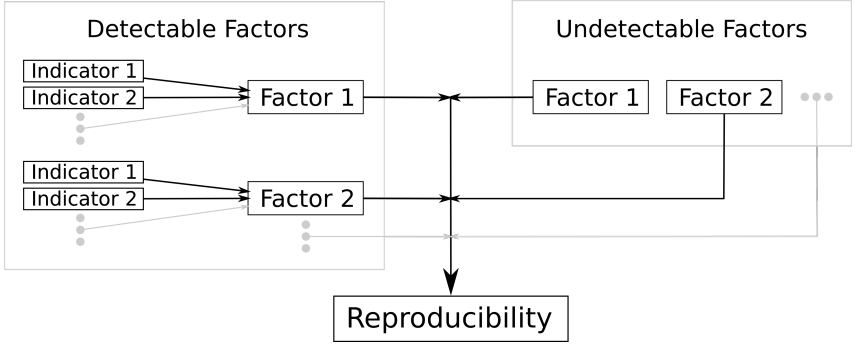
Identification & Classification

Identification of ML Reproducibility Factors

UNIVERSITÄT PASSAU

- Literature review
- 14 different influencing factors

Classification of ML Reproducibility Factors



- **Factor:** Direct correlation with reproducibility.
- **Indicator:** Software-based detectable properties of a factor.

ML Reproducibility Factors

UNIVERSITÄT PASSAU

- Source Code Availability & Documentation
- Software Environment
- Out-of-the-Box Buildability
- Data Set Availability
- Random Seed Control
- Hyperparameter-Logging
- Model-Serialization
- Hardware Environment (1)
- Data Set Preprocessing (1)
- Research Practices & Experimental Design (1)
- Underfitting & Overfitting (1)
- Knowledge Gap (2)
- Probability Hacking (2)
- Bias (2)

Not implemented because: (1): Complexity; (2): Undetectable

UNIVERSITÄT PASSAU

Implementation

Implementation Challenge

UNIVERSITÄT PASSAU

- Tool measures indicator values
- Goal: Score for each factor

→ Approach to connect them needed

Factor-Indicator-Connection

UNIVERSITÄT PASSAU

***If** a factor score is calculated from **different indicators**:*

- Different value ranges → Min-Max Normalization
- Measured indicator value good or bad? → Comparative Values
- Indicators have varying degrees of influence on a factor. → Weights
- Is a factor's score good or bad? → Thresholds

Data Sets for Factor-Indicator-Connection

UNIVERSITÄT PASSAU

We have manually gathered:

- 20 reproducible ML repositories.
- 20 non-reproducible ML repositories.

→ Analysed these sets with our tool.
→ Statistical evaluation of the results to determine comparative values, weights and thresholds.

Tool Output

- The progress is displayed on the command line.
- All measured indicator values are saved in a .csv file.
- **Feedback file** (.md) is generated based on the results.

Exemplary Feedback File

REPRODUCIBILITY FACTOR SCORING (from 0 to 1):

Reproducibility factor	Score	Feedback	T*	A*	L*
Source-code availability and documentation	0.93	Score higher than top threshold (T). Very good.	0.8	0.54	0.28
Software environment	0.93	Score higher than top threshold (T). Very good.	0.61	0.46	0.31
Dataset availability and preprocessing	1	Score equal to top threshold (T). Very good.	1	-	0
Random seed control	1.0	Score higher than top threshold (T). Very good.	0.94	0.73	0.51
Model serialization	1	Score equal to top threshold (T). Very good.	1	-	0
Hyperparameter logging	0	Score equal to lower threshold (L). Major improvements should be made.	1	-	0
Out-of-the-box buildability	0	Score equal to lower threshold (L). Major improvements should be made.	1	-	0

*- Thresholds computed from respective reproducible and non-reproducible sets of repositories. More information on this in the associated thesis in chapter 6.

SOURCE CODE AVAILABILITY AND DOCUMENTATION FEEDBACK

1. License scoring (score: 1.0 out of 1.0): All found licenses are open-source.
2. README overall scoring (score: 0.93 out of 1.0): See sub-ratings for more information.
   - Average length scoring (score: 1.0 out of 1.0): We determined that 82 or more lines are best. We found 222.0 lines (in average) in the README file(s)
   - Average accessible links scoring (score: 0.65 out of 1.0): We determined that 4 or more links are best. We found 3.0 accessible links (in average) in the README file(s)
3. Source-code pylint scoring (score: 0.63 out of 1.0): Not normalized pylint rating is 3.61 where 10.0 is best. We found that readable code is best for reproducibility and consider ratings over 5.71 as desirable.
4. Source-code-comment-ratio scoring (score: 1.0 out of 1.0): We measured a average ratio of 6.44. We consider a ratio below 8.73 as best. But more comments are fine too (which would make the ratio smaller). Well documented code is important.

Source-code availability and documentation is calculated from the above identifiers. Since these have a different influence on the overall result, they are weighted as follows:

- License weight: 0.3
- Readme weight: 0.5
- Sub-readme: Length weight: 0.8
- Sub-readme: Average accessible links weight: 0.2
- Pylint rating weight: 0.1
- Code-comment-ratio weight: 0.1

SOFTWARE ENVIRONMENT FEEDBACK

1. Libraries used in the source-code and mentioned in config file scoring (score: 0.53 out of 0.6): We found that 87.5% of the used libraries are defined in the

UNIVERSITÄT PASSAU

Evaluation

Evaluation of the Indicator Weights

UNIVERSITÄT PASSAU

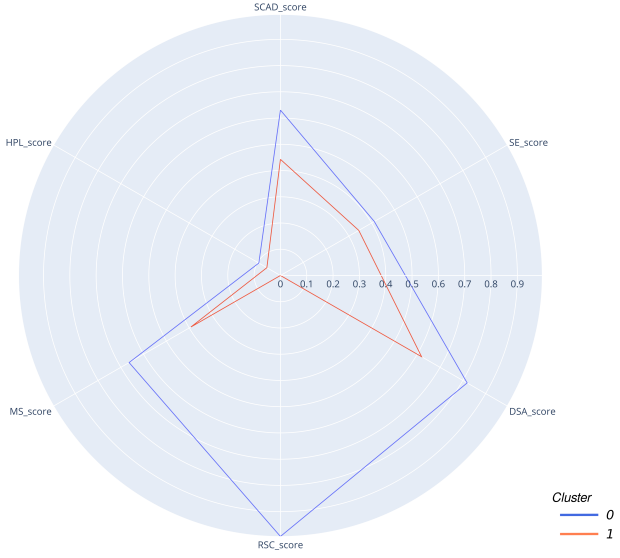
- Performed feature importance ranking to evaluate our chosen weights (same data sets).
- In general, the findings coincide with our argumentation.

K-Means Cluster Analysis

UNIVERSITÄT PASSAU

- Input set: Measured factor scores of 100 randomly gathered ML repositories from GitHub and 100 ones randomly gathered from PapersWithCode (labeled accordingly).

Result of the Cluster Analysis



Result of the Cluster Analysis II

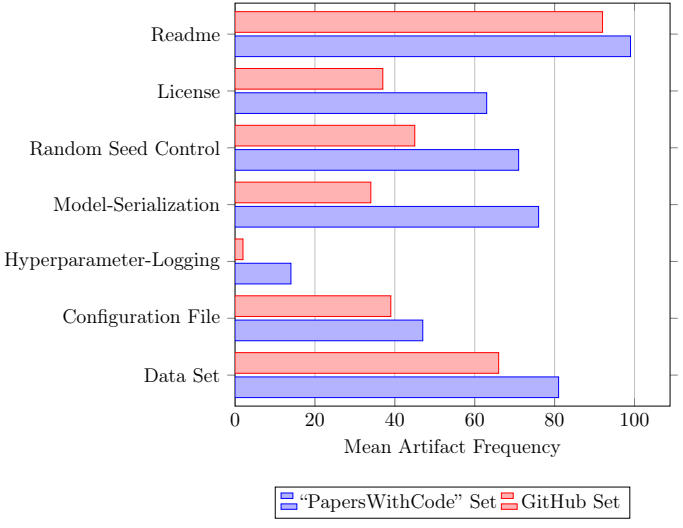
UNIVERSITÄT PASSAU

- **Cluster 0**: About two thirds of these are from the PapersWithCode set. → Comparable scores as the reproducible set.
- **Cluster 1**: About two thirds of these are from the GitHub set.
- Modifications to the input set did not affect these observations.

Statistical Evaluation

- Cluster analysis to determine key characteristics
- Statistical evaluation to determine key differences of these sets

Statistical Evaluation II



UNIVERSITÄT PASSAU

Discussion

Discussion

UNIVERSITÄT PASSAU

- Identification: 14 influencing factors
- Classification: Indicator-based reasoning
- Implementation: Factor-Indicator-Connection
- Evaluation: Findings support our hypothesis

Limitations

UNIVERSITÄT PASSAU

- Small data set sizes
- Limited tool functionality
- Supports Python language only
- Requires GitHub hosted repository

Future Work

UNIVERSITÄT PASSAU

- Enhancing the tool
- Refining the factor-indicator-connection approach by collecting larger data sets
- Evaluation on larger data sets
- Combination of the functionality of this tool and the approach that estimates the likelihood of replicability of a paper using an ML model

UNIVERSITÄT PASSAU

Conclusion

Conclusion

UNIVERSITÄT PASSAU

This tool enables:

- researchers to check their repository prior to publication.
- automated feedback of how well a repository to be examined complies with the reproducibility guidelines.
- the comparison of these metrics between different repositories.

UNIVERSITÄT PASSAU

We believe that this tool can have a positive contribution to reducing the reproducibility crisis.

Time for Questions.

UNIVERSITÄT PASSAU

Thanks for your attention!

Thesis repository: https://github.com/martinloos/ml_repository_reproducibility_analysis_tool