Reproducibility Report for ACM SIGMOD 2020 Paper: "LISA: A Learned Index Structure for Spatial Data"

ILIA PETROV, Reutlingen University, Germany

Include a short summary of the reproducibility findings.

1 INTRODUCTION

In this report we describe the reproducibility effort for "*LISA: A Learned Index Structure for Spatial Data*" [2] which is a joint work of *Pengfei Li*, Zhejiang University, China, *Hua Lu*, Roskilde University, Denmark, *Qian Zheng*, NTU, Singapore, *Long Yang*, Zhejiang University, China, and *Gang Pan*, Zhejiang University, China.

The code [2] on which this reproducibility report is based, has been made available on a public GitHub repository [1]. It also contains external links to the datasets necessary for reproducibility.

2 SUBMISSION

The reproducibility submission comprises the LISA [2] codebase and a set of configuration files, together with three dataset files (approx. 1.2 GB each).

Codebase. The codebase of LISA [2] is available under https://github.com/pfl-cs/LISA as a public repository. The code package comprises a set of python scripts under the *src* directory the most relevant being *main.py*. Furthermore the code package, comprises a set of parameter files, the most relevant of which is *src/utils/core/config.py*.

Readme and Instructions. The LISA package [1] comes with a concise, but well-defined description in a *readme* file. The users are asked to make some preparatory steps prior to running the repeatability experiments, i.e. installing python packages (`numpy`, `scipy` and `scikit-learn`).

The expected directory structure is shown in Figure 1.a. To run the experiments the users need to consider `config.py` and run `python main.py` in the `src` directory. The individual experiments for reproducing most of the figures are coded in `main.py` and the users are expected to uncomment the respective parts as they see fit.

Datasources. The users also need to download the 2D and 3D experimental datasets. These are stored externally, but LISA repository [1] provides the respective links. The expected directory structure is displayed Figure 1.b. The authors provide *uniform* 2D and 3D datasets, while the the paper [2] reports numbers for *zipfian* datasets as well as higher number of dimensions.

3 HARDWARE AND SOFTWARE ENVIRONMENT

The paper [2] reports no numbers regarding the hardware setup, however hardware settings are available in [1]. An overview of the settings used in the submission and for the reproducibility is provided in Table 2.

4 REPRODUCIBILITY EVALUATION

4.1 Process

The initial submission contained a 4D dataset. Running LISA [2] out-of-the-box took more than 1.5 months on the experimental machine (without completing). Thankfully the authors provided assistance, a new codebase and new 2D and 3D uniform datasets. With those revised artifacts the the evaluation process took approx. 72 hours for each of the datasets. Due to the tools used to

(a) Source structure

```
src
|-- FLAGS_DEFINE.py
|-- __init__.py
|-- main.py
|-- solution
|   |-- LISA.py
|   |-- __init__.py
|   |-- lattice_regression.py
|   `-- piecewise_linear_curve_fit.py
`-- utils
    |-- FileViewer.py
    |-- __init__.py
    |-- core
    |   |-- __init__.py
    |   `-- config.py
    |-- data_gen.py
    |-- layout_utils.py
    |-- np_utils.py
    `-- string_util.py
```

(b) Data structure

```
workspace
`-- LISA
    |-- 2d_uniform
    |   |-- data
    |   |   |-- 2d_uniform_data_0.npy
    |   |   |-- 2d_uniform_data_2.npy
    |   |   |-- 2d_uniform_data_3.npy
    |   |   |-- 2d_uniform_query_ranges.qr
    `-- 3d_uniform
        `-- data
            |-- 3d_uniform_data_0.npy
            |-- 3d_uniform_data_2.npy
            |-- 3d_uniform_data_3.npy
            `-- 3d_uniform_query_ranges.qr
```

Table 1. Directory Structure

Table 2. Hardware & Software environments

	Repository [1]	Reproducibility Review
CPU	32 Intel Xeon E5-2620	4× Intel Xeon X7560 (4 sockets/NUMA nodes)
cores	8	4 (per socket)
GHz	2.10	2.27
RAM	128 GB	1024 GB
Storage	HDD (3.7 TB)	HDD (5 TB)
OS	–	Debian
Kernel	–	4, release 9 (4.9.0-4-amd64)
Python	–	2.7.13

the create the graphs in [2], the authors do not provide `gnuplot` scripts and therefore only output based validation is possible.

4.2 Results

The numbers for LISA in Figure 8 [2] and Figure 9 [2] for 2 and 3 uniform dimensions appear to be valid. The same holds for the numbers for LISA in Figures 9 and 13[2].

REFERENCES

[1] Pengfei Li and et al. [n.d.]. LISA GitHub Repository. https://github.com/pfl-cs/LISA.

[2] Pengfei Li, Hua Lu, Qian Zheng, Long Yang, and Gang Pan. 2020. LISA: A Learned Index Structure for Spatial Data. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data* (Portland, OR, USA) *(SIGMOD '20)*. Association for Computing Machinery, New York, NY, USA, 2119–2133. https://doi.org/10.1145/3318464.3389703