Reproducibility review of: GeoXTag: Relative Spatial Information Extraction and Tagging of Unstructured Text

Jakub Krukar

2022-06-09

This report is part of the reproducibility review at the AGILE conference. For more information see https://reproducible-agile.github.io/. This document is published on OSF at https://doi.org/10.17605/OSF.IO/3G9S8. To cite the report use

Krukar, J. (2022, May 5). Reproducibility review of: GeoXTag: Relative Spatial Information Extraction and Tagging of Unstructured Text. https://doi.org/10.17605/OSF.IO/3G9S8

Reviewed paper

Syed, M. A., Arsevska, E., Roche, M., and Teisseire, M.: GeoXTag: Relative Spatial Information Extraction and Tagging of Unstructured Text, AGILE GIScience Ser., 3, 16, https://doi.org/10.5194/agile-giss-3-16-2022

Summary

The main contribution of the paper are two web applications on the streamlit.io platform. The paper includes the DASA section and links to live online instances of the applications, as well as to GitHub repositories with the code and data. The repositories are well documented. I was able to clone the repositories and run the code, reproducing examples demonstrated on figures in the paper. One of the reproduced figures differed from the one presented in the paper. The tables demonstrating results of the evaluation of the app were partially reproducible but returned slightly different values. The paper has been partially reproduced.

Reproducibility reviewer notes

I have attempted to reproduce 4 elements of the paper:

- Results of the RSI Extraction app (RSI-Parser repository) presented in Figure 2 Step 2.
- Results of the RSI Tagging app (RSI-Tagger repository) presented in Figure 2 Step 3.
- Table 1 and
- Table 2 with the results of the evaluation

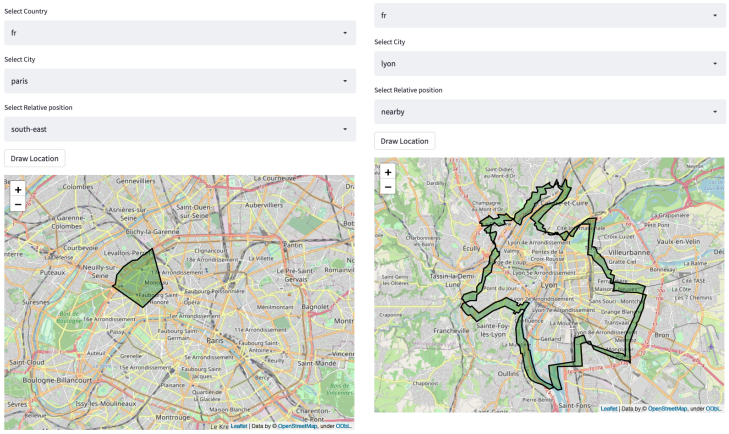
RSI Extraction app

The link to the RSI Extraction Demo provided in Table 3 did not work (error message: "Whoops — something went wrong! An error has been logged."). I have cloned the corresponding GitHub repository (https://github.com/mehtab-alam/RSI-Parser). The repository does not specify a license.

I have run the app locally following the instruction in the README file. After pressing the "Extract" button, a message "Running addPatterns()." appeared for a few minutes. Then, the message disappeared and it seemed that the app was finished without providing any results. After leaving the window open for another few minutes, the results appeared. I have obtained the same result as those demonstrated in Figure 2 Step 2 in the manuscript.

RSI Tagging app

The link to the RSI Tagging Demo listed in Table 3 worked. I have cloned the provided GitHub repository (https://github.com/mehtab-alam/RSI-Tagger). The README file did not contain explicit information on how to run the streamlit app locally so I used the command provided in the README of the RSI-Parser repository. After locally running the RSI Tagging app and selecting the corresponding cities from the drop-down menus I obtained results for "South-East Paris" (left) and for "Nearby Lyon" (right):



These figures are very similar to the original Figure 2 Step 3 provided in the manuscript:

The entries for the remaining two boxes in Figure 2 step 3 ("West London" and "East Florence") were not available by default in the RSI Tagging app provided in the repository.

Following the instructions in the README file I was able to add these inputs to the app. The description in the README specifies where to add new cities in the code (however the description does not name the specific file, neither where one can find the list of accepted spelling of the possible cities/countries).

After modifying the code, adding new cities can be done with the command

```
python GeojsonUtil.py
```

that initially did not work on my machine. I needed to refer to python3 explicitly:

```
python3 GeojsonUtil.py
```

The reproduced figure for "West London" (left) looked very similar to the one in the manuscript. The polygon on the figure reproduced for "East Florence" (right) has a notably different shape, especially in its top-left and top-right:

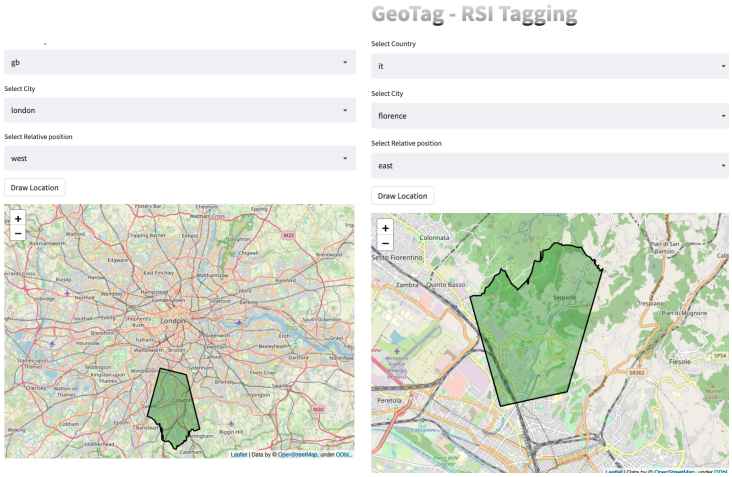


Table 1. Evaluation of RSI Extraction in Diseases News Dataset

I was able to access the GitHub repository containing the manually annotated dataset used for creating Table 1 (https://github.com/mehtab-alam/RSI_Disease_Dataset). ~~The repository does not contain the code used to aggregate data from the .csv files, nor to calculate Precision, Recall, or F-Score.~~ After contacting the authors, they have provided code with which I was able to reproduce results reported in Table 1. The provided streamlit application worked required me to manually remove the files located in the 'logs' directory, open the app, load .csv files with the data, and click on the "Evaluate" button. The obtained results were similar but not identical with those reported in the manuscript. In some cases the value of the first column differed, suggesting that the dataset present in the repository may be slightly different from the one analysed for the manuscript.

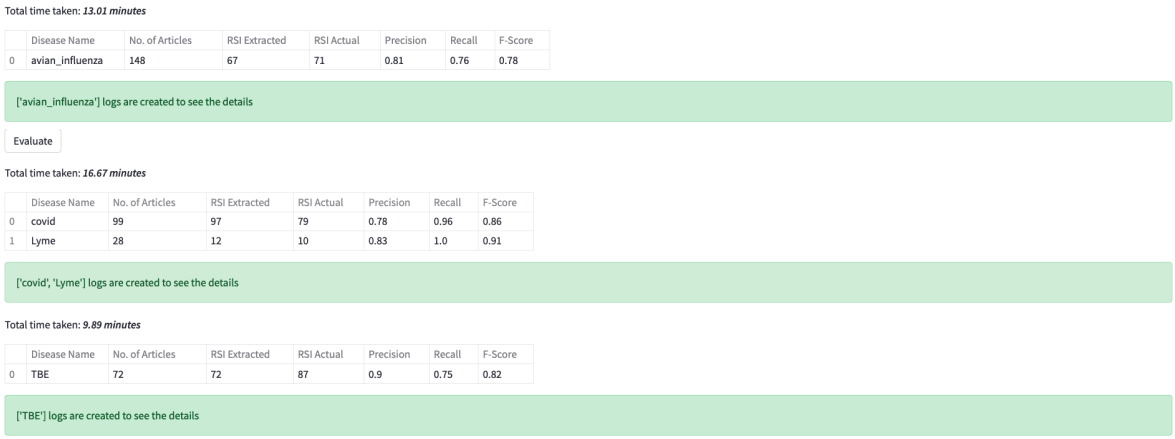


Table 2. Evaluation of RSI Geotagging

I was able to access a Google Doc document used for evaluating the Tagging app (https://drive.google.com/file/d/1y-RMUuy3UbFg0wVZAEgJJ1_pgM0GlvQs/view). The document contains manually assigned Scores but no code used to aggregate this data into Table 2. After contacting the authors, they have included additional information in the document that details how the data present in the document was aggregated to that presented in the manuscript.

Recommendations

- Specify a license in each repository.
- In the README file of the RSI-Tagger repository add an explicit reference to the file 'GeojsonUtil.py' in the 'Configuration' section.
- In the above, specify that the python3 is required.
- In the above, add information that the list of the available countries and cities with their supported spelling is available in the corresponding .csv files.
- In the README file of the RSI-Tagger repository add information on how to run the streamlit app locally.
- In the above, add information about the expected run time and about the fact that the web app might seem unresponsive.