SoftwareX 25 (2024) 101627

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Geospatial Analytics Extension for KNIME

Lingbo Liu [a], Xiaokang Fu [a], Tobias Kötter [b], Kevin Sturm [b], Carsten Haubold [b], Weihe Wendy Guan [a], Shuming Bao [c], Fahui Wang [d,*]

[a] Center for Geographic Analysis, Harvard University, MA 02138, USA
[b] KNIME GmbH, Körtestr. 10, 10967 Berlin, Germany
[c] China Data Institute, Ann Arbor, MI 48108, USA
[d] Department of Geography & Anthropology, Louisiana State University, LA 70803, USA

ARTICLE INFO

Keywords:
Geospatial analytics
KNIME analytics platform
GIS
Visual programming
Replicability and reproducibility

ABSTRACT

The Geospatial Analytics Extension for KNIME (GAEK) is an innovative tool designed to integrate visual programming with geospatial analytics, streamlining GIS education and research in social sciences. GAEK simplifies access for users with an intuitive, visual interface for complex spatial analysis tasks and contributes to the organization of the GIS Knowledge Tree through its geospatial analytics nodes. This paper discusses GAEK's architecture, functionalities, and its transformative impact on GIS applications. While GAEK significantly enhances user experience and research reproducibility, future updates aim to expand its functionality and optimize its bundled environment.

Metadata

Nr	Code metadata description	*Please fill in this column*
C1	Current code version	*1.2*
C2	Permanent link to code/ repository used for this code version	https://github.com/spatial-data-lab/knime-geospatial-extension
C3	Permanent link to reproducible capsule	
C4	Legal code license	*MIT License*
C5	Code versioning system used	
C6	Software code languages, tools and services used	*Python*
C7	Compilation requirements, operating environments and dependencies	*KNIME Analytics Platform 5.1.2*
C8	If available, link to developer documentation/manual	https://github.com/spatial-data-lab/knime-geospatial-extension/tree/main/docs
C9	Support email for questions	spatialdatalab@lists.fas.harvard.edu

1. Motivation and significance

Geospatial data has become an integral part of data science, essential in a wide range of applications from urban planning to health research [1,2]. The rapid advancement in geospatial analysis and the surge of GIS software packages have introduced steep learning curves and fragmented geographical knowledge, presenting significant challenges in integrating with the evolving landscape of data science and consequently in GIS education [3,4].

The learning curve for tools requiring proficiency in programming languages like R or Python can present challenges for novices, potentially impacting the wider adoption of GIS tools. In contrast, popular GIS software such as ArcGIS Pro or QGIS, which are map-interface based, primarily rely on clicking actions for execution. While user-friendly, this approach may not provide clear visibility of the underlying workflows. [5]. Furthermore, the scattered nature of geospatial knowledge necessitates a comprehensive approach to integrate and understand its interconnected aspects, emphasizing the need for a cohesive GIS knowledge framework [6]. Integrating GIS with the broader developments in data science is crucial, given that geospatial data forms a significant subset of the field [7].

Visual programming emerges as a promising direction for GIS education [8]. Tools like Scratch offer an engaging programming environment with a user-friendly GUI, including a puzzle-like editor that simplifies error messages and supports incremental program execution [9]. Such tools significantly enhance students' grasp of programming concepts and computational thinking [10], boosting their motivation and engagement [11]. Visual data science platforms like Orange,

* Corresponding author.
E-mail address: fwang@lsu.edu (F. Wang).

https://doi.org/10.1016/j.softx.2023.101627
Received 7 November 2023; Received in revised form 7 December 2023; Accepted 19 December 2023
Available online 28 December 2023

RapidMiner, and KNIME lower the entry barrier by providing a visual programming framework and a mind map-like interface for organizing and understanding data flows [12]. KNIME, with its Python node development interface, has been our choice for developing GIS function plugins, leveraging its data science platform [13].

The Geospatial Analytics Extension for KNIME, developed specifically for GIS applications, employs visual programming, knowledge tree support, and integration with the data science platform. This extension, built on the Python extension for KNIME, aims to make GIS tools more intuitive and user-friendly, akin to constructing with LEGO or using mind maps [14–17]. It facilitates a systematic approach to node development based on a geospatial knowledge structure, ensuring clarity and adaptability to the latest developments.

Our software aims to bridge the gap between complex GIS operations and users with diverse expertise. It incorporates visual and no-code features to simplify GIS tasks, making them more accessible and intuitive. This not only mitigates the steep learning curve but also supports the structured development of the GIS knowledge tree. Moreover, the visual workflow interface, akin to mind mapping, allows users to modularly visualize and assemble GIS components. This method aligns with constructivist learning theories, where knowledge is built incrementally. It also embodies Bruner's "spiral curriculum" theory, enabling students to revisit data science concepts at various learning stages, each time gaining a deeper understanding. Our software's innovative approach could significantly alter the teaching and application of GIS, making it more approachable for a broader audience of researchers and professionals.

The software's user-friendly interface, consistent with other KNIME components but simplified, allows users to interact with GIS components effortlessly. The visual nature of the software reduces the reliance on extensive coding, though it still accommodates advanced users with the option to integrate custom Python scripts for specialized tasks. Accompanying the software is a KNIME experimental manual that aligns with the latest GIS textbooks, guiding users through standard GIS operations.

2. Software description

2.1. Software architecture

Fig. 1 illustrates the development framework of the Geospatial Analytics Extension for KNIME (GAEK), primarily deployed on a GitHub repository. This repository houses all the essential components for the KNIME Python Extension development framework [18] and KNIME Python API [19], including GIS functional modules, auxiliary files such as icons and descriptions, and a YML file to set up the necessary Python environment. Users can set up their development and testing environment by installing the Python environment from the provided YML file and the Python Extension Development (Labs) framework. Once GAEK is released as an official plugin within KNIME, users can install the extension directly along with the bundled Python environment [20]. The repository on GitHub is also the platform for ongoing updates and improvements, which are informed by feedback from the KNIME community and other users.

The development files on GitHub mainly consist of three files and three folders (Fig. 2). The "knime_extension" folder is the core node development directory, the "tests" folder contains test data and workflows, and the "docs" folder includes some basic instructional manuals. The config.yml contains the local path for Python environment, while LICENSE.txt and README.md provide the corresponding MIT license and a general introduction.

Within the folder "knime_extension", there is the geospatial python environment yml file, as well as the knime.yml file for extension information. The "icons" folder contains icons for each node. The folder "src" contains the source code for each core GIS function. Some common function Python files are located in the "util" folder, while "geospatial_ext" is the root file that KNIME calls for all GIS functions. The development of all core node categories will be compiled in the nodes folder.

The structure of each Python file is methodically organized into three primary sections: dependencies, node category definition, and node definition, as illustrated in Fig. 3a. The node definition itself is further segmented into five distinct components: node information, input data, output data, node description, and the node's core functionality.

The core functionality of a node is comprised of several key elements. Firstly, the node input parameters are defined and subsequently rendered within the node's User Interface (UI) for user interaction. Secondly, the format of the output table data is specified to ensure that the results are structured correctly. Lastly, the core GIS code that the node executes is encapsulated within this section. Fig. 3b provides a snapshot of the 'Create Random Point' node's interface, serving as a practical example. This interface primarily consists of fields for the input geometry attributes and the desired number of points to be generated. The functionality of this node is to create a specified number of random points within the boundaries of the input polygon, effectively converting the original polygon data into point data.

2.2. Software functionalities

The Geospatial Analytics Extension for KNIME offers a comprehensive suite of tools designed to enhance the capabilities of geospatial data analysis within the KNIME environment. The functionalities in 1.2 release version are categorized to cover the basic spectrum of geospatial data processing, from initial data input to advanced spatial modelling and visualization. Table 1 lists all nodes and descriptions for the 12 categories.

3. Illustrative examples

In the wake of Hurricane Ian's landfall in Florida on 2022, a comprehensive impact analysis based on Geotagged Tweets sentimental analysis was conducted using the Geospatial Analytics Extension for KNIME. This case study exemplifies the integration of diverse datasets and the application of GAEK's robust functionalities to assess the aftermath of the natural disaster.

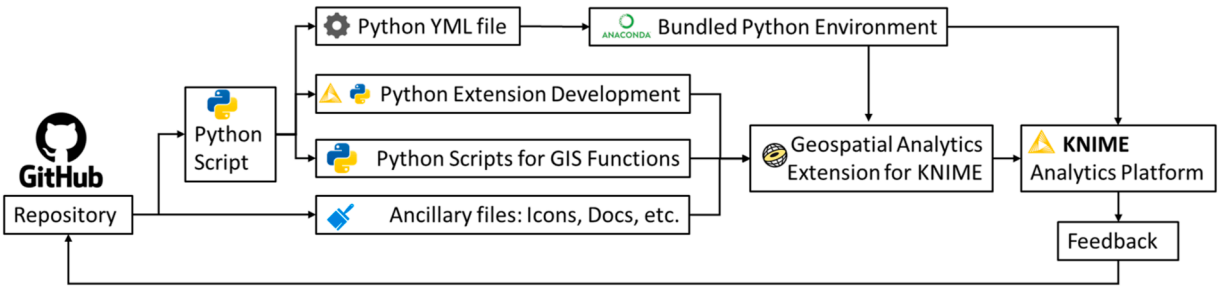

Fig. 1. The development framework of Geospatial Analytics Extension.

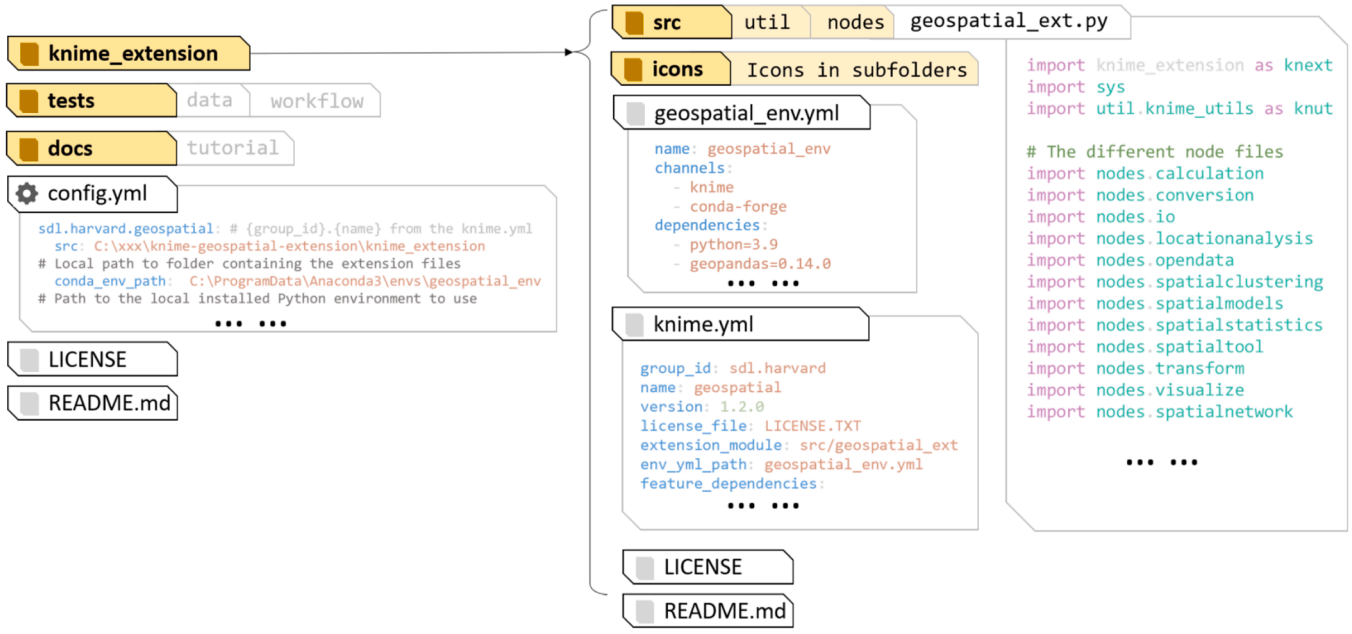

Fig. 2. File structure for geospatial analytics extension for KNIME on GitHub.

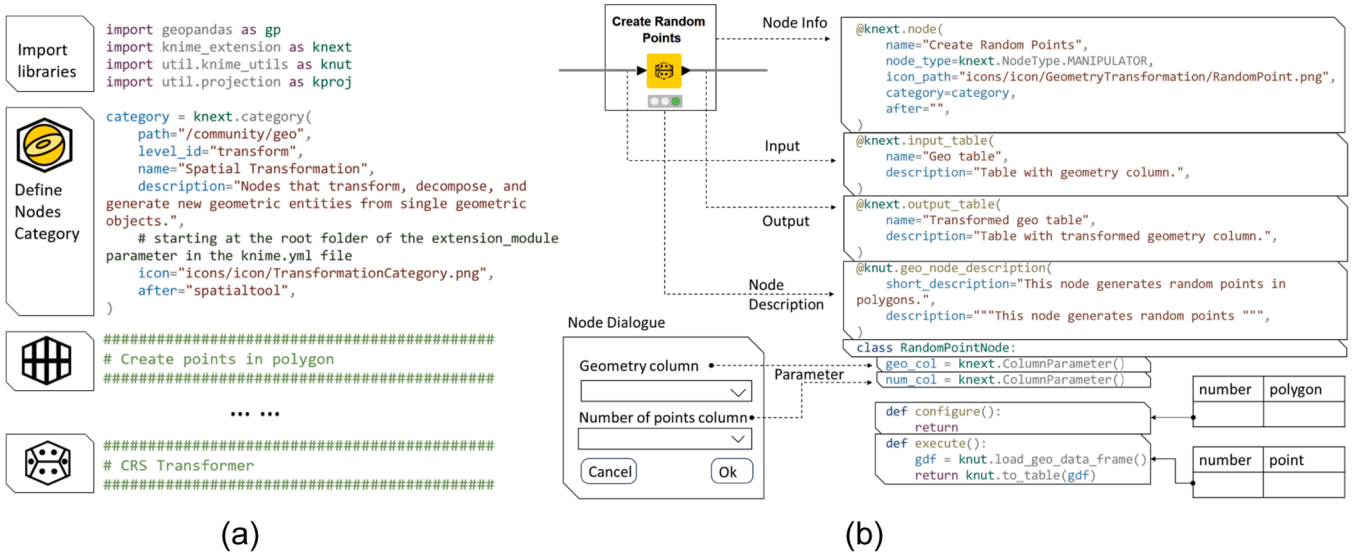

Fig. 3. Scripting structure for (a) Nodes category and (b) Node of Create Random Point.

The goal is to utilize GAEK to integrate various datasets, including the hurricane's trajectory, geotagged tweets, U.S. Census data, and US 2020 TIGER maps, to analyse the population affected and the public sentiment regarding the hurricane. Fig. 4 illustrates the key spatial tasks in the case study, and Fig. 4 (b) the whole workflow for this case study.

Data Integration: The GeoFile Reader node is employed to import the hurricane's track data.

U.S. Census data and TIGER maps are used as base maps for population data. A dataset of over 100,000 geotagged tweets with the keyword "hurricane" is preprocesses to be assigned with sentiment scores.

Spatial Analysis: The Multiple Ring Buffer node creates buffers at different distances from Hurricane Ian's track to delineate affected zones. The Overlay node assesses the population within these buffers. The Dissolve node aggregates the total population in the affected areas.

Spatial Visualization: The Geospatial View node visualizes these tweets, allowing for color-coding based on sentiment scores.

Comparative Analysis: Spatial joins are applied to incorporate the spatial information of the impact zone into the tweets dataset. This allowed us to categorize the tweets into two phases: before and after Hurricane Ian's landfall, using September 28th as the key date. Sentiment scores are compared between the central 50-kilometer buffer zone and the 100 to 150-kilometer buffer zones.

Results: The analysis reveals that the buffer zone from Tampa to Jacksonville was the most populous and faced the greatest challenges. Before the hurricane made landfall, the central zone had the lowest sentiment scores, indicating concern and anxiety. After landfall, the sentiment in the outer buffer zones became negative, possibly due to inadequate preparation for the hurricane's extensive impact area. The sentiment scores' changes highlight the relief felt by those in the hurricane's direct path and the distress of those in the south side of Florida.

The GAEK facilitated a comprehensive impact analysis of Hurricane Ian by integrating and analysing spatial data and sentiment scores from social media. The software's capabilities in handling spatial IO,

Table 1
Nodes in geospatial analytics extension for KNIME 1.2

Category	Nodes	Description
Spatial IO	GeoFile Reader, GeoFile Writer, GeoPackage Reader, and GeoPackage Writer	Enable users to read and write spatial data across a variety of formats, ensuring compatibility and ease of data exchange.
Spatial calculation	Area, Length, Coordinates XYZ, Bounds, Total Bounds, Bounding Box, Bounding Circle, Convex Hull, and Unary Union	Calculate spatial properties, create geometric envelopes, and perform complex unions of spatial entities.
Spatial manipulation	Spatial Join, Nearest Join, Overlay, Dissolve, Clip, Simplify, Buffer, Multiple Ring Buffer, Euclidean Distance, Haversine Distance, Create Grid, and Voronoi (Thiessen) polygons	Merge, modify, and construct new spatial features based on spatial relationships, proximity, and geometric characteristics
Spatial transformation	Projection, Geometry To Point, Polygon To Line, Points To Line, Line To MultiPoint, Multipart To Singlepart, Create Random Points, and Line Endpoints	Allow for the transformation of spatial data, enabling the conversion of single geometries into different forms and the generation of new geometric shapes
Spatial conversion	Lat/Lon to Geometry, GeoJSON to Geometry, WKT to Geometry, Geometry to Lat/Long, Geometry to GeoJSON, Geometry to WKT, Geocoding, Reverse Geocoding, and Geometry to Metadata are functionalities used for	Convert between various geometric and textual representations, or apply geocoding operations, and metadata extracting from geometries.
Spatial visualization	Geospatial View, Geospatial View Static, Kepler.gl Geoview, and Spatial Heatmap are functionalities	Facilitate the visualization and exploration of geospatial data.
Exploratory spatial data analysis	Spatial Weights, Global Geary's C, Global Getis-Ord G, Global Moran's I, Local Getis-Ord G, and Local Moran's.	Provide various measures and methods to analyze spatial autocorrelation.
Spatial modelling	OLS with Spatial Test, 2SLS with Spatial Test, Spatial Error Model, Spatial Lag Model, Spatial Lag Panel Model, Spatial Error Panel Model, GWR Model, GWR Predictor, and MGWR Model	Conduct spatial regression, panel modelling, and geographically weighted regression analyses.
Location analysis	P-median Solver, LSCP Solver, MCLP Solver, P-center Solver, and MAEP Solver	Solve various location optimization problems
Spatial clustering	SKATER, SCHC, REDCAP, MaxP-Greedy, AZP-Greedy, Peano Curve, MSSC Initialization, MSSC Refiner, Isolation Tackler, Mean Center, and Standard Deviation Ellipse	Support regionalization operations, including clustering and measuring spatial pattern
Spatial network	Google Distance Matrix, OSRM Distance Matrix, Road Network Distance Matrix, and Road Network Isochrone Map	creating distance matrices and isochrone maps using various routing engines
Open datasets	OSM Boundary Map, OSM Road Network, OSM POIs, US2020 Census Data, US2020 TIGER Map, and US ACS 5-Year Estimates tools	Provide access to various public datasets, such as OpenStreetMap data and US census data,

calculations, manipulation, transformation, visualization, and exploratory analysis were crucial in providing insights into the hurricane's impact on population and public sentiment. This case study serves as an illustrative example of how GAEK can be applied to real-world scenarios, demonstrating its utility in spatial data analysis and its impact on disaster management and response strategies. More case studies can be openly downloaded at the KNIME Hub [21].

4. Impact

The integration of the Geospatial Analytics Extension for KNIME (GAEK) with the educational resources has been provided in a KNIME workbook, "Computational Methods and GIS Application in Social Science- lab manual" [22]. It offers a unique opportunity to investigate how visual programming interfaces can enhance learning outcomes in social science education. The textbook, coupled with ArcGIS Pro tutorials, titled as ""Computational Methods and GIS Application in Social Science" [23], sets a traditional foundation, while the GAEK version workbook allows for exploration into how alternative GIS platforms can influence the development of computational methods in social sciences. Research could focus on the comparative effectiveness of these tools in teaching complex concepts and whether a visual programming approach facilitates a deeper understanding of spatial data analysis.

The GAEK extension enhances the pursuit of existing research questions by providing a more accessible and replicable framework for geospatial analysis. This combination allows for a seamless transition between learning GIS concepts and applying them to real-world social science research questions, potentially increasing the rate of discovery and the quality of insights gained from spatial data.

For educators and students in social sciences, GAEK has likely changed the daily practice of teaching and learning GIS. The software's visual programming capabilities, as outlined in the GAEK lab manual, offer a more interactive and engaging way to understand GIS applications, which may lead to a more dynamic classroom experience and a stronger grasp of computational methods in social science research.

The adoption of GAEK within social science education, as evidenced by its inclusion in a main textbook and lab manual, suggests that the software has found a niche within this academic community. The extent of its use beyond this group could be measured by tracking its incorporation into other disciplines' curricula, its mention in interdisciplinary research, and its download rates by users outside the social sciences.

The commercial implications of GAEK, particularly in the context of social sciences, could be explored by examining its role in data analytics firms, urban planning agencies, and other sectors where GIS is crucial. If the methodologies outlined in the "Computational Methods and GIS Application in Social Science" textbook have been adopted by industry professionals, this would indicate the software's practical value. Any spin-off companies that have emerged to specialize in GAEK-based solutions would further highlight its commercial viability and impact.

In summary, the "Computational Methods and GIS Application in Social Science" textbook and the accompanying GAEK lab manual not only serve as educational resources but also as catalysts for new research questions, improved methodologies, changes in educational practices, and potential commercial applications.

5. Conclusions

The Geospatial Analytics Extension for KNIME (GAEK) represents a significant stride forward in the realm of GIS education and application within social sciences. By harnessing the principles of visual programming, GAEK demystifies complex geospatial analysis, making it more accessible to a broader range of users, from students to seasoned researchers. Its integration into the educational framework through the "Computational Methods and GIS Application in Social Science" textbook and lab manual exemplifies its potential to enhance learning outcomes and foster a deeper understanding of spatial data analysis. The benefits of GAEK are manifold; it not only simplifies the learning curve associated with traditional GIS software but also encourages a more systematic and replicable approach to geospatial research, thereby

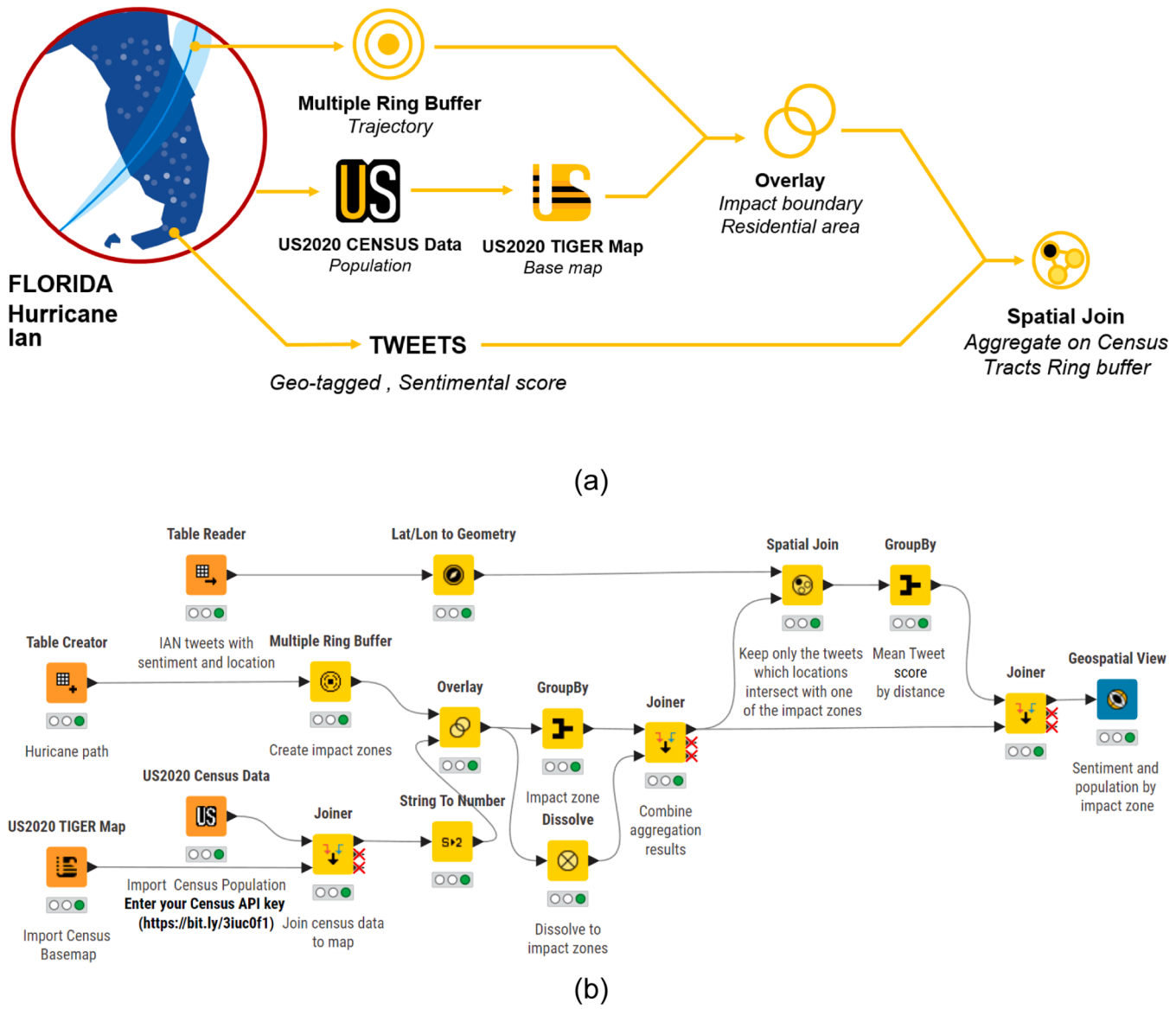

Fig. 4. The workflow for this case study: (a) Conceptual flow chart and (b) KNIME workflow.

expanding the horizons of computational social science.

However, the journey of GAEK is not without its challenges. The current limitations, such as the need for a broader array of GIS function nodes and the substantial size of the bundled Python environment, suggest areas for future development. Addressing these limitations will be crucial in ensuring that GAEK continues to evolve in step with the dynamic needs of geospatial analytics. Looking ahead, the roadmap for GAEK includes expanding its library of function nodes to cover a more comprehensive range of GIS operations and optimizing the Python environment to make it more lightweight and efficient. In doing so, GAEK will continue to serve as a pivotal tool in the ever-growing field of geospatial data science, pushing the boundaries of what can be achieved in both academic and commercial settings.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Lingbo Liu reports financial support was provided by KNIME. Co-authors, Tobias Kötter, Kevin Sturm, and Carsten Haubold are employed by KNIME. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

have shared the link to my data/workflow at the KNIME Hub space for Center for Geographic Analysis at Harvard University , See reference 21

Acknowledgements

This work is partially funded by NSF grant #1841403.

References

[1] Kerski JJ, Demirci A, Milson AJ. The global landscape of GIS in secondary education. J Geogr 2013;112(6):232–47.
[2] Kim M, Bednarz R. Development of critical spatial thinking through GIS learning. J Geogr Higher Educ 2013;37(3):350–66.
[3] Schulze U, Kanwischer D, Reudenbach C. Essential competences for GIS learning in higher education: a synthesis of international curricular documents in the GIS&T domain. J Geogr Higher Educ 2013;37(2):257–75.

[4] Jo I, Hong JE. Effect of learning GIS on spatial concept understanding. J Geogr 2020;119(3):87–97.
[5] Marra WA, et al. Using GIS in an Earth Sciences field course for quantitative exploration, data management and digital mapping. J Geogr Higher Educ 2017;41(2):213–29.
[6] Mkhongi FA, Musakwa W. Perspectives of GIS education in high schools: an evaluation of uMgungundlovu district, KwaZulu-Natal, South Africa. Educ Sci 2020;10(5):131.
[7] Bowlick FJ, Goldberg DW, Bednarz SW. Computer science and programming courses in geography departments in the United States. Profession Geogr 2017;69(1):138–50.
[8] Karelkhan N, Kadirbek A, Schmidt P. Setting Up and implementing ArcGIS to work with maps and geospatial data with python for teaching Geoinformation systems in higher education. Int J Emerg Technol Learn (Online) 2023;18(14):271.
[9] Walshe N. Developing trainee teacher practice with geographical information systems (GIS). J Geogr Higher Educ 2017;41(4):608–28.
[10] Olsson M, Mozelius P, Collin J. Visualisation and gamification of e-Learning and programming education. Electron J e-Learn 2015;13(6):452-465.
[11] Banerjee G, Murthy S, Iyer S. Effect of active learning using program visualization in technology-constrained college classrooms. Res Pract Technol Enhance Learn 2015;10(1):1–25.
[12] Malarvizhi AS, et al. An open-source workflow for spatiotemporal studies with COVID-19 as an example. ISPRS Int J Geo-Inf 2021;11(1):13.
[13] Berthold M R, Cebron N, Dill F, et al. KNIME: the Konstanz information miner studies in classification. Data Anal Know 2007.
[14] Meerbaum-Salant O, Armoni M, Ben-Ari M. Learning computer science concepts with scratch. In: Proceedings of the Sixth international workshop on Computing education research; 2010.
[15] Robinson JA, Block D, Rees A. Community geography: addressing barriers in public participation GIS. Cartogr J 2017;54(1):5–13.
[16] Petrasova A, et al. Tangible modeling with open source GIS. Springer; 2018.
[17] Mathews AJ, DeChano-Cook LM, Bloom C. Enhancing middle school learning about geography and topographic maps using hands-on play and geospatial technologies. J Geogr 2023:1–11.
[18] *Create a new Python based KNIME extension*. KNIME analytics platform 5.1 2023 [cited 2023 11-3]; Available from: https://docs.knime.com/latest/pure_python_node_extensions_guide/index.html.
[19] *KNIME Python API*. 2023 [cited 2023 11-3]; Available from: https://knime-python.readthedocs.io.
[20] Geospatial analytics extension for KNIME [cited 2023 11-3]; Available from: https://hub.knime.com/center%20for%20geographic%20analysis%20at%20harvard%20university/extensions/sdl.harvard.features.geospatial/latest/.
[21] *KNIME* hub space for *center for geographic analysis at Harvard university* [cited 2023 11-3]; Available from: https://hub.knime.com/center%20for%20geographic%20analysis%20at%20harvard%20university.
[22] Liu L, Wang F. Computational methods and GIS applications in social science-lab manual. CRC Press; 2023.
[23] Wang F, Liu L. Computational methods and GIS applications in social science. CRC Press; 2023.