Supplement of Earth Syst. Sci. Data, 18, 3671–3696, 2026
https://doi.org/10.5194/essd-18-3671-2026-supplement

Open Access Earth System Science Data

Supplement of

OneDZ: a global detrital zircon database and implications for constructing giant geoscience database

Keran Li et al.

Correspondence to: Xiumian Hu (huxm@nju.edu.cn)

Outline

S1. AI tools in constructing database

The web address for DataExpo is http://dataexpo.deep-time.org/. The main website is shown in Figure S1. By using "detrital zircon" as the keyword, the system can find and rank relevant items on open-access websites to identify potential online databases (Figure S2).

Regarding the use of GPT as an agent to find papers, it's important to note that this involves a workflow for processing data. Large language models (LLMs) currently face limitations in directly searching specific websites like Google Scholar, Geoscienceworld, and others. Therefore, an essential step is to collect summaries of potential titles from these websites. In this research, we tested the GPT-Agent workflow on Google Scholar. Initially, we adapted a collector from a GitHub repository (https://github.com/JessyTsu1/google_scholar_spider). We then applied ChatGPT using the API key locally. After configuring the keys, GPT was used to adjust title information. Following this, GPT worked with the adjusted title tables, and through several prompt engineering cycles and interactive checks, it was able to predict the sources of zircon data with fair accuracy.

DeepShovel (https://deepshovel.deep-time.org/) is an online platform integrated with artificial intelligence technologies, specifically designed to assist researchers in Earth sciences with data extraction from scientific literature in PDF format (Figure S3). The platform leverages advanced neural network models and user interaction to efficiently identify and extract data from tables, figures, maps, and text within the documents.

To utilize DeepShovel, researchers initially upload literature files containing the desired data. The platform then automatically parses different sections of the documents, including metadata, text, tables, and maps. For metadata extraction, DeepShovel employs tools such as Grobid and Science Parse, integrating extraction results from various tools through a voting mechanism to obtain key information like the title, authors, abstract, publication year, etc., of the literature.

For extracting academic entities from the text, the platform uses weak supervision learning models and rules to assist users in identifying and extracting specific information, such as geological era names. Table data extraction is another critical function of DeepShovel. The platform employs object detection models like Detectron2, combined with rules and user interaction, to help users locate and recognize tabular data within the literature. Users can adjust the structure of the table with system assistance and utilize optical character recognition technology such as Tesseract to extract cell content.

Map recognition and geographical location extraction are also key modules of DeepShovel (Zhang et al., 2022a). The platform can recognize maps within the literature and assist users in determining the latitude and longitude of points on the map through drawing and marking operations. Finally, all extracted data can be integrated into a unified table during the data integration phase, facilitating the compilation of a harmonised dataset. DeepShovel supports project-level data integration, allowing users to set the header of the master table on the project page and automatically match and integrate data from various parts of the document.

The introduction of the online platform is detailed in Zhang et al. (2023), Zhang et al. (2022a), and Zhang et al. (2022b).

S2. Python and MySQL code snippets for data cleaning

In this research, we developed Python scripts primarily aimed at automatically checking and removing erroneous rows in the dataset. These scripts include DuplicateRemove.py for removing duplicate rows, DuplicateCheck.py for identifying duplicates, and DuplicateLog.py for generating logs. All scripts utilize the Pandas package to operate on CSV files. However, we do not recommend using XLSX files for data cleaning, as Excel tables with over 1,000,000 rows require at least 128GB of running memory. Additionally, we recommend a minimum of 64GB of RAM on personal computers when using these Python scripts to process the full dataset.

To address the inefficiencies of Python scripts on gigabyte-scale files, we employed SQL queries as an intermediate step for fast deduplication, format validation, and cross-table consistency checks during the cleaning workflow. While MySQL offers efficient batch processing for these temporary operations, all final outputs were exported back to flat CSV files with a uniform schema. The lack of a visual interface in command-line SQL can be a barrier for Earth science researchers; therefore, we occasionally used Navicat software to visually inspect intermediate query results. It must be emphasized that Navicat and MySQL were used solely as internal cleaning utilities, not as the final data distribution format. All processed data are released as plain CSV files on Zenodo, requiring no database connectivity.

All the Python and SQL code snippets mentioned in this research are available for download at https://github.com/KeranLi/Global-Detrital-Zircon. This resource aims to provide researchers with the tools necessary to efficiently clean and harmonise large-scale geoscience datasets.

S3. Python code snippets in coordinates

Firstly, the automatic conversion process must account for the multiple formats of DMS (Degrees, Minutes, Seconds) coordinates. After careful re-evaluation, we identified that the primary DMS formats use either symbols like “° ′ ″” or the terms "degree/minutes/seconds" to represent coordinates. Other more complex cases include combinations of symbols, letters, spaces, slashes, dashes, and varying capitalization. To handle this, we implemented a fuzzy retrieval method that first automatically detects the separator symbols. This allows for quick extraction of values from different time scales based on the identified separators, followed by rapid calculations.

Moreover, the construction of the database involves importing multiple DMS format data in batches for conversion. Given the variety of data storage formats generated during the crowdfunding process, we have also developed fast automated parsing of multivariate files within Python scripts, further enhancing the efficiency of data conversion.

In addition to coordinate conversion, we also implemented a method for estimating latitude and longitude coordinates based on geometric relationships in images, using Python. This method considers the distortion effects caused by projection and manual selection errors. Projection distortion is closely linked to the choice of projection mode during map drawing. In sedimentology, it is often challenging to trace the original author's settings for projection modes in spatial maps. Therefore, we determined the distortion parameters experimentally. First, known spatial points were projected using different

projection modes at various spatial scales (e.g., 1 km by 1 km). Then, the estimated coordinates of the target point in the image were repeatedly measured, and the distortion coefficient was iteratively calculated until the estimated coordinates closely matched the actual coordinates.

All the Python code snippets used in this research can be downloaded from https://github.com/KeranLi/Global-Detrital-Zircon. This repository provides the tools necessary for efficient data conversion and coordinate estimation in the context of global detrital zircon studies.

S4. Paleo spatial reconstruction method

The calculation of Paleo globality is further enhanced using the PyGplate package. Initially, the data with spatial coordinates is linked to a Plate ID. This bundled data can then be restored to any desired time using PyGplate's encapsulated code. The data points reconstructed based on rigid blocks are evaluated for globality using grid partitioning methods.

To improve code execution efficiency, we introduced GC packet dynamic memory management in this study. The computational processes were carried out on the Intel Xeon E5-2680 v3 (12 Cores, 30MB Cache, 2.50 GHz, 9.6 GT/s QPI) processor at the Supercomputing Center of Nanjing University, running on a Linux operating system.

All the Python code snippets used in this research can be accessed and downloaded from https://github.com/KeranLi/Global-Detrital-Zircon.

S5. LLM-driven agents for detrital zircon database construction

The database construction pipeline employs three sequentially integrated LLM-driven modules to automate the extraction, cleaning, and standardization of detrital zircon U-Pb and Lu-Hf data from published literature. First, ZirconDector_LLM agent performs multimodal data extraction from PDF articles and supplementary files. The extractor dynamically switches between text-based parsing (for standard PDFs) and vision-based image recognition (for garbled or nested-header tables) using commercial LLM APIs (DeepSeek, Doubao, and Kimi). To minimize API costs and extraction errors, the system follows an attachment-first strategy, where Excel and Word supplementary files are parsed locally whenever available, requiring zero API calls, while the main PDF is reserved for metadata retrieval and table extraction only when necessary. For both text and vision modes, the LLM outputs data as raw two-dimensional arrays (row 0 as standardized headers, subsequent rows as raw values), which are then mapped to geological fields by the Python backend through multi-layer fault tolerance, including header keyword matching, positional fallback for common geological table layouts, and weighted geological consistency checks that auto-correct ratio–age misplacements.

Second, the Multi-Agent Data Cleaning System corrects domain-specific errors through specialized agents operating under a coordinator architecture. After initial field classification, independent agents handle literature metadata (author formatting, journal standardization, DOI validation), geographic information (coordinate format conversion, latitude/longitude swap detection using country-specific bounding boxes), U-Pb isotope data (ratio validation, age calculation, discordance assessment), Lu-Hf

isotope data (εHf and TDM age computation), and geological descriptors (rock type and stratigraphic name normalization). A rule-based DataQualityFixer module operates entirely offline to correct manual entry errors-such as unit conversions (e.g., years to Ma), spelling standardization (e.g., “Sinkiang” to "Xinjiang"), and coordinate formatting, like a dynamic engine selector (pandas, Polars, or Dask) ensures scalable processing across file sizes ranging from megabytes to gigabytes. Each processed file receives a quality score (0-100), with scores below 60 flagged for mandatory manual review.

Third, ZirconRegular_LLM consolidates all cleaned records into a uniform 64-column schema encoded in UTF-8-BOM with comma delimiters and LF line endings. Large datasets are split into chunks of approximately 100,000–130,000 rows to maintain computational efficiency, and automated statistical scripts generate per-field coverage reports quantifying non-null ratios, numeric parseability, and unique value counts. This flat-file structure ensures direct usability in any standard data-analysis environment without requiring proprietary database infrastructure.

All the Python code snippets used in this research can be accessed and downloaded from https://github.com/KeranLi/onedz_llm_coding

S6. Figures

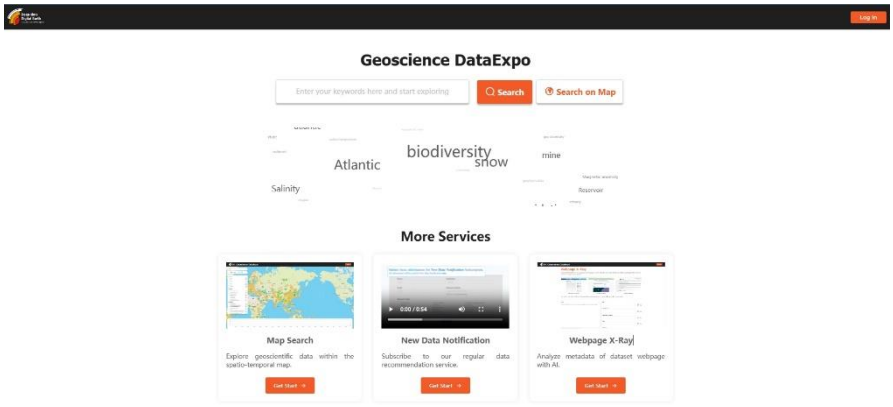

Figure S1: Screenshots of the DataExpo system (http://dataexpo.deep-time.org/), reproduced with permission from the Deep-time Digital Earth (DDE) program. The platform is developed and maintained by DDE.

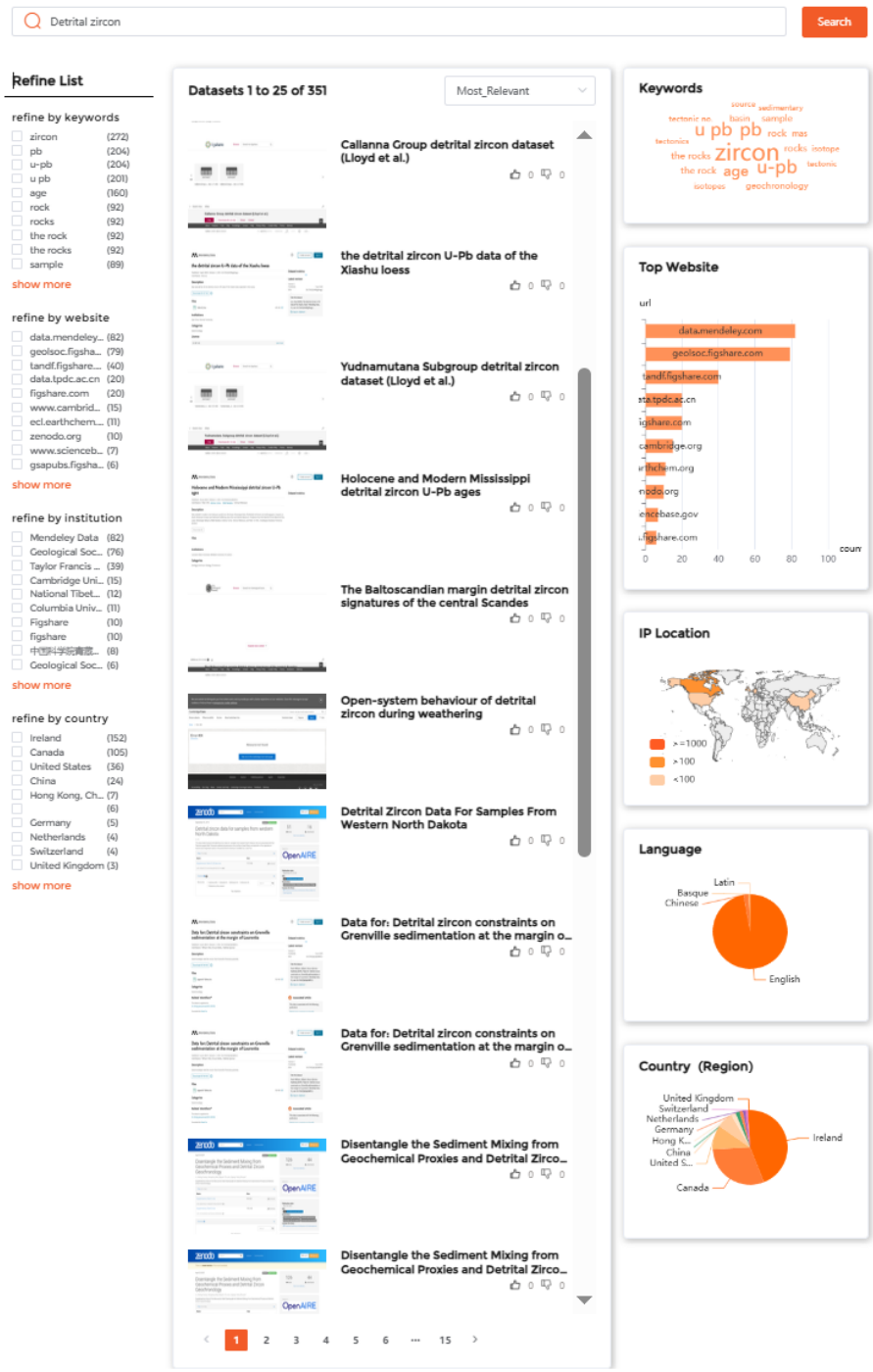

Figure S2: Screenshots of the search results in the DataExpo system (http://dataexpo.deep-time.org/), reproduced with permission from the Deep-time Digital Earth (DDE) program.

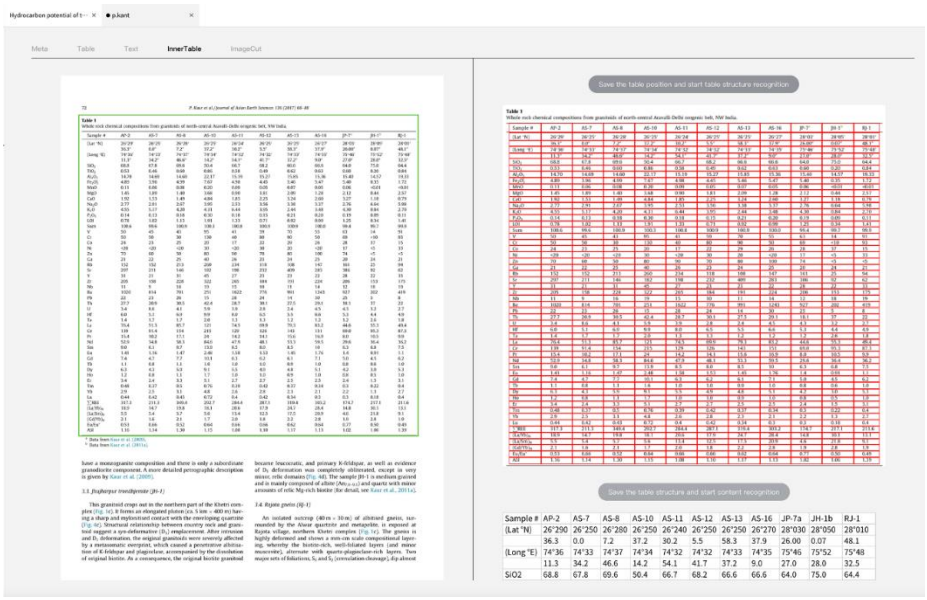

Figure S3: Screenshot of DeepShovel, reproduced with permission. DeepShovel is an AI-assisted data extraction platform developed by Zhang et al. (2022) for the Deep-time Digital Earth (DDE) program.

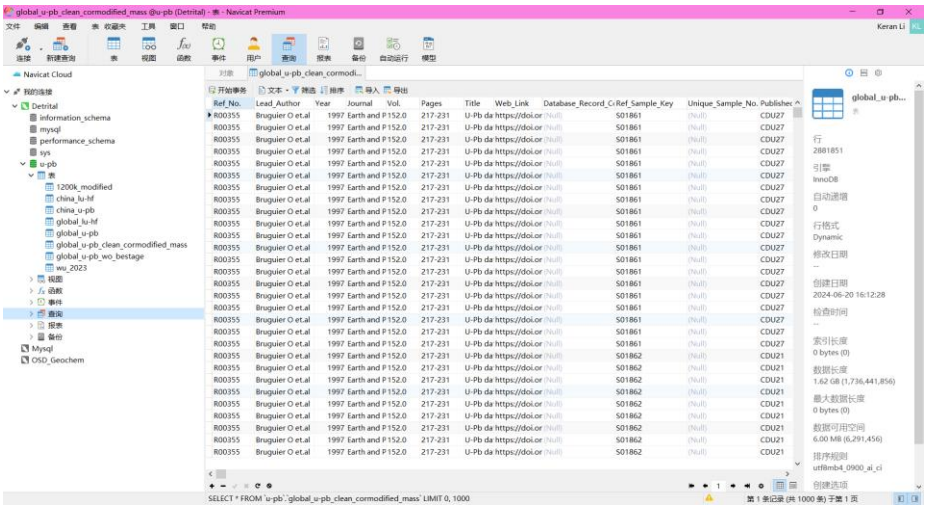

Figure S4: Temporary visual query interface used during internal data cleaning. Navicat (from PremiumSoft CyberTech Ltd. © Navicat.) or equivalent GUI tools such as open-source DBeaver Community (https://dbeaver.io/) was employed solely as an auxiliary environment for inspecting intermediate SQL deduplication results. The final OneDZ dataset is distributed exclusively as flat CSV files on Zenodo, and no database software is required to access or analyse the data.

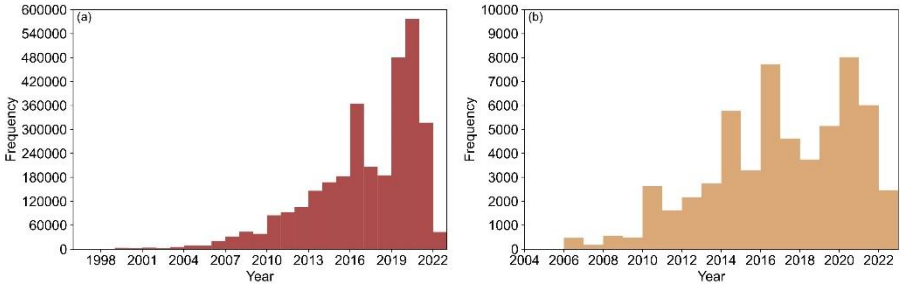

Figure S5: U-Th-Pb and Lu-Hf references variation with time. (a) U-Th-Pb; (b) Lu-Hf.

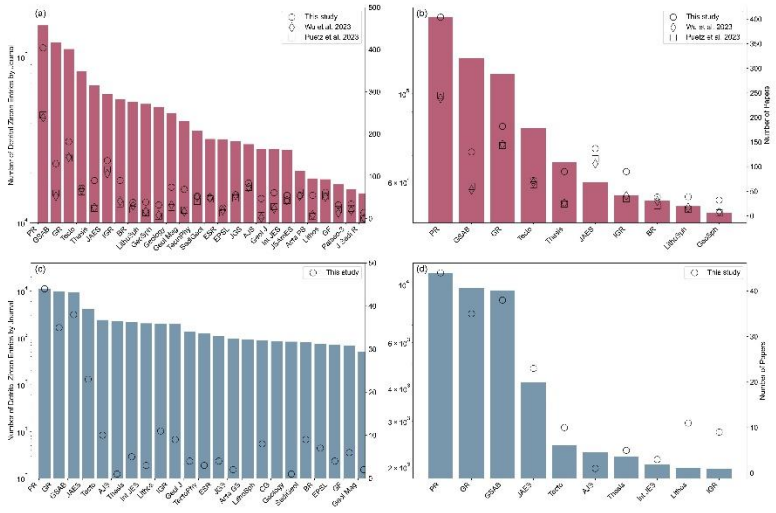

Figure S6: Bar plots of the references. (a) The single zircons from different journals in U-Th-Pb data (Only display journals contribute zircons supreme 20000); (b) Top-10 journals contribute the detrital zircon records in U-Th-Pb data; (c) The single zircons from different journals in U-Th-Pb data (Only display journals contribute zircons supreme 20000); (d) Top-10 journals contribute the detrital zircon records

in U-Th-Pb data. PR = Precambrian Research, GSAB = GSA Bulletin, GR = Gondwana Research, Tecto = Tectonics, JAES = Journal of Asian Earth Sciences, IGR = International Geology Review, BR = Basin Research, LithoSph = Lithosphere, Geol Mag, TectoPhy = Tectonophysics, SediGeol = Sedimentary Geology, ESR = Earth-Science Reviews, EPSL = Earth and Planetary Science Letters, JGS = Journal of the Geological Society, Geol J = Geological Journal, Int JES = International Journal of Earth Sciences, JSAmES = Journal of South American Earth Sciences, Acta PS = Acta Petrologica Sinica, GF = Geoscience Frontiers, Palaeo-3 = Palaeogeography, Palaeoclimatology, Palaeoecology, J Sedi R = Journal of Sedimentary Research.

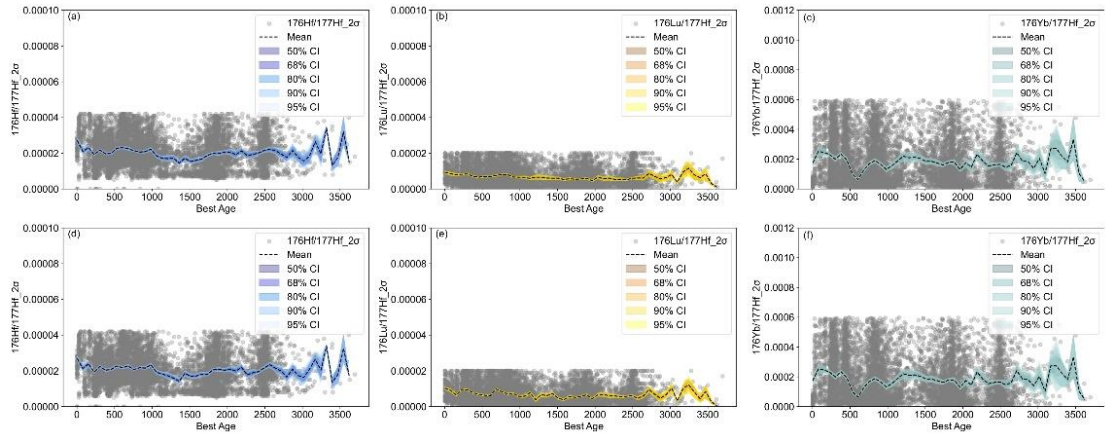

Figure S7: Temporal variations of isotopic uncertainties in Lu-Hf dataset. (a) $^{176}Hf/^{177}Hf$ with bootstrap resampling; (b) $^{176}Lu/^{177}Hf$ with bootstrap resampling; (c) $^{176}Yb/^{177}Hf$ with Monte-Carlo resampling; (d) $^{176}Hf/^{177}Hf$ with Monte-Carlo resampling; (e) $^{176}Lu/^{177}Hf$ with Monte-Carlo resampling; (f) $^{176}Yb/^{177}Hf$ with Monte-Carlo resampling.

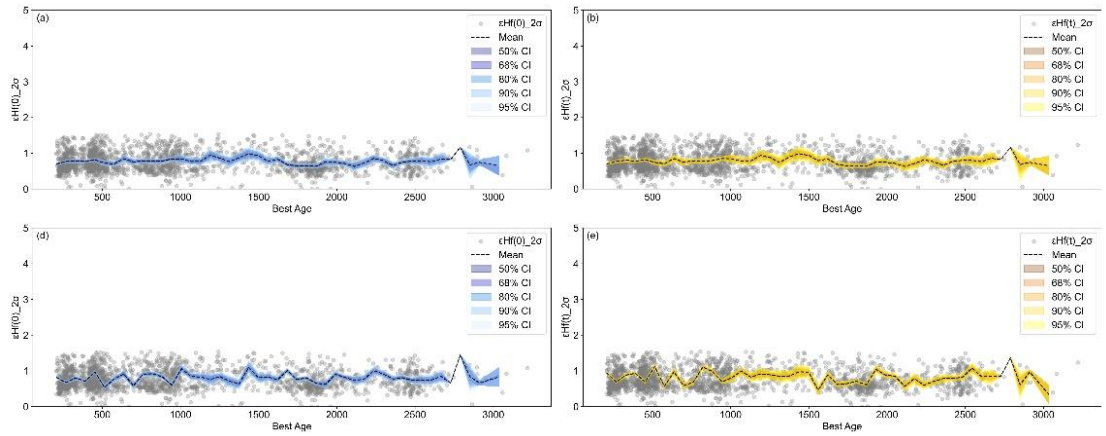

Figure S8: Temporal variations of εHf uncertainties in Lu-Hf dataset. (a) εHf(0) with bootstrap resampling; (b) εHf(t) with bootstrap resampling; (c) εHf(0) with Monte-Carlo resampling; (d) εHf(t) with Monte-Carlo resampling.

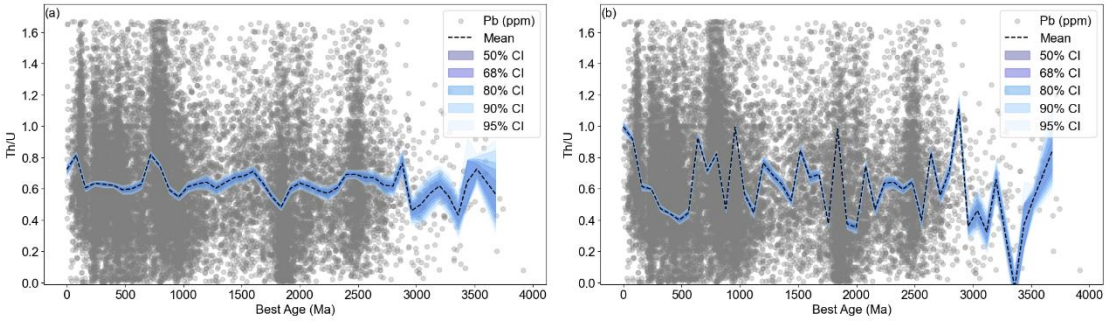

Figure S9: Temporal variations of Th/U in U-Pb dataset. (a) Th/U with bootstrap resampling; (b) Th/U with Monte-Carlo resampling.

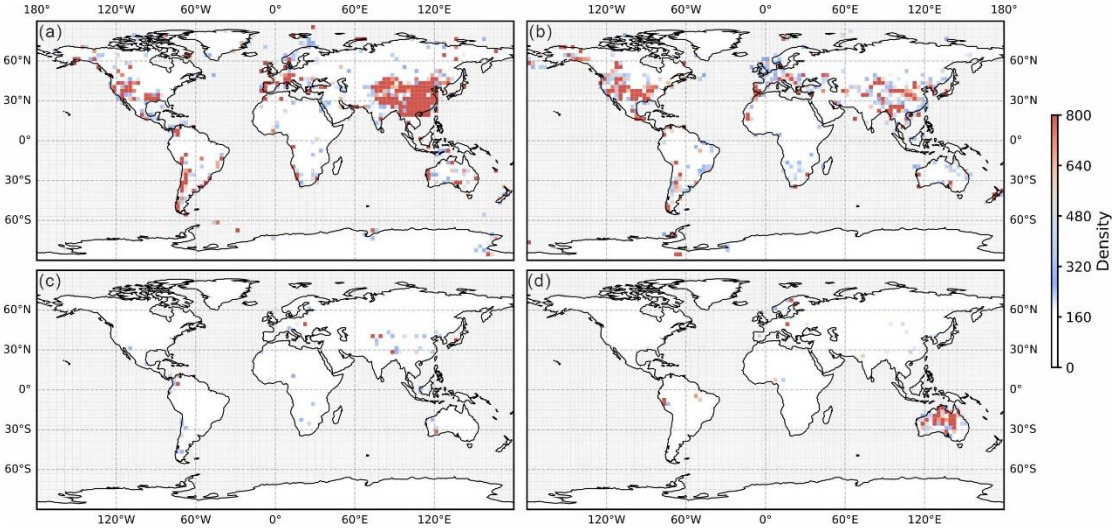

Figure S10: Kernel density estimate maps of detrital zircon records (the spatial resolution is 1°×1°). (a) database from Wu et al. (2023); (b) database from Puetz et al. (2024); (c) database from GeoRoc; (d) database from EarthBank.

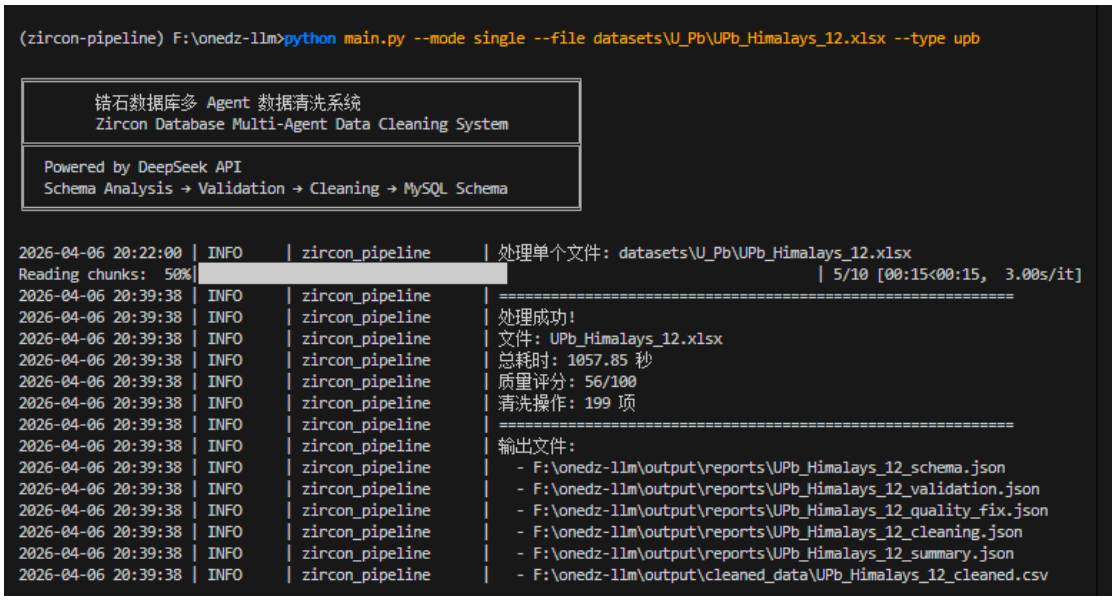

Figure S11: The screenshot of the LLM-driven system in this research. The software shown is Visual Studio Code, developed by Microsoft Corporation.


```
{
  "task_id": "a3385e32-3aa6-4e15-92c4-bb38760410b1",
  "papers": [
    {
      "doi": "10.1130/abs/2017am-303687",
      "title": "CA-TIMS U-PB DATES FROM HADEAN ZIRCON FROM THE JACK HILLS, AUSTRALIA:
      "authors": [
        "James L. Crowley",
        "Mark D. Schmitz",
        "John S. Myers",
        "Jesse B. Walters"
      ],
      "journal": "Geological Society of America Abstracts with Programs",
      "year": 2017
    },
    {
      "doi": "10.1130/0091-7613(1989)017<1076:upaotb>2.3.co;2",
      "title": "U-Pb age of the Baltoro granite, northwest Himalaya, and implications for monazite
      "authors": [
        "Randall R. Parrish",
        "R Tirrul"
      ],
      "journal": "Geology",
      "year": 1989
    },
    {
      "doi": "10.1016/j.chemgeo.2009.12.007",
      "title": "Detrital zircon U–Pb and Hf isotopic data from the Xigaze fore-arc basin: Constraint
      "authors": [
        "Fu-Yuan Wu",
        "Wei-Qiang Ji",
        "Chuan-Zhou Liu",
        "Sun-Lin Chung"
      ],
      "journal": "Chemical Geology",
      "year": 2009
    },
    {
      "doi": "10.1016/j.precamres.2015.12.004",
      "title": "Detrital zircon U–Pb, Lu–Hf, and O isotopes of the Wufoshan Group: Implications fc
      "authors": [
        "Hong-fu Zhang",
```


Figure S12: Parts of the irrelevant articles that people uploaded. The software shown is the Microsoft Windows text editor (Notepad).

S7. Reference

Zhang, S., Hu, X., Zhang, J., Li, Q., Xu, Y., Yu, Y., and Han, L.: A database of detrital zircon U–Pb ages and Hf isotopic compositions from the Tarim, West Kunlun, Pamir, Tajik and Tianshuihai terranes, Geoscience Data Journal, 11, 2, 118-127, https://doi.org/10.1002/gdj3.213, 2023a.

Zhang, S., Jia, Y., Xu, H., Wang, D., Li, T. J.-j., Wen, Y., Wang, X., and Zhou, C.: KnowledgeShovel: An AI-in-the-Loop Document Annotation System for Scientific Knowledge Base Construction, arXiv preprint arXiv:2210.02830, 2022a.

Zhang, S., Jia, Y., Xu, H., Wen, Y., Wang, D., and Wang, X.: Deepshovel: An online collaborative platform for data extraction in geoscience literature with ai assistance, arXiv preprint arXiv:2202.10163, https://doi.org/10.48550/arXiv.2202.10163, 2022b.

Zhang, S., Xu, H., Jia, Y., Wen, Y., Wang, D., Fu, L., Wang, X., and Zhou, C.: GeoDeepShovel: A platform for building scientific database from geoscience literature with AI assistance, Geoscience Data Journal, 10, 519-537, https://doi.org/10.1002/gdj3.186, 2023b.

Puetz, S. J., Spencer, C. J., Condie, K. C., and Roberts, N. M.: Enhanced U-Pb detrital zircon, Lu-Hf zircon, δ18O zircon, and Sm-Nd whole rock global databases, Scientific Data, 11, 56, https://doi.org/10.1038/s41597-023-02902-9, 2024b.

Wu, Y., Fang, X., and Ji, J.: A global zircon U–Th–Pb geochronological database, Earth System Science Data, 15, 5171-5181, https://doi.org/10.5194/essd-15-5171-2023, 2023.