

RESEARCHING WITH AI.
BETWEEN OPPORTUNITY
AND IMPOSITION

February 11th, 2026, 13:30-17:15 CET
University of Bern, Hochschulstrasse 4, Room 115



b
UNIVERSITÄT
BERN

Kaspar Gubler: The Data Iceberg: Context and Interpretation in AI-Supported Digital History



Iceberg model for Digital History and AI

Above the surface (visible):

- structured data (*may contain uncertainty*)
- dates, places, names (*often vague or ambiguous*)
- events, affiliations (*fragmentary or disputed*)
- quantified patterns (*shaped by gaps and bias*)

Below the surface (invisible):

- historical context
- semantics and meaning
- social norms and practices
- source bias and silences
- interpretation and uncertainty



A **digital history** research database combining structured prosopographical data on approximately 70,000 students and scholars who attended European universities between 1250 and 1550.

Institute of History, Department of Medieval History

Data corpus in REPAC

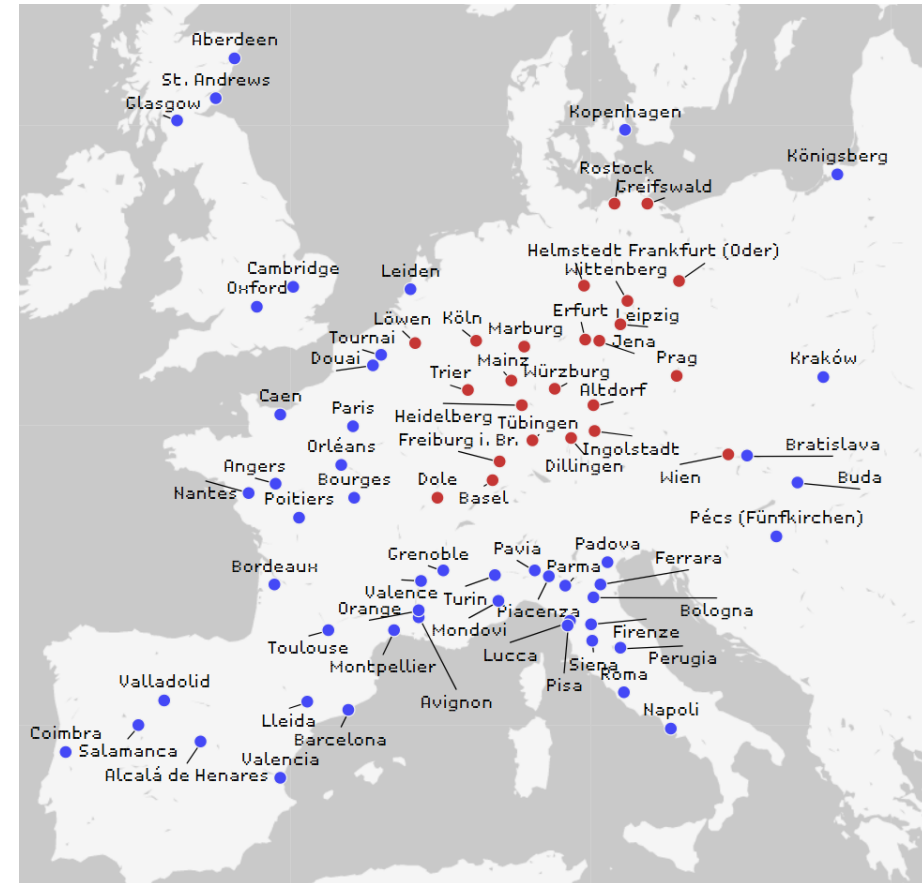
70,000 students and scholars

400,000 life events (biographical stages)

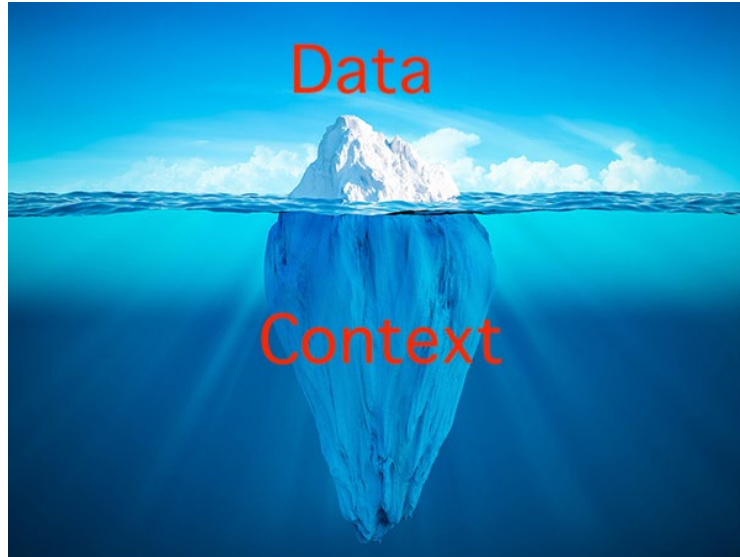
26,000 places and institutions

= A basis for data visualisations of:

- regions of origin
- places of study
- spheres of activity and knowledge spaces



Universities attended in Europe up to the mid-16th century



REPAC and AI

AI benefits for REPAC:

- data collection
- data analysis at scale
- pattern detection (mobility, networks, careers)
- data management (consistency, validation)

AI challenges for REPAC:

- **data analysis with AI**
- data iceberg: uncertainty in structured historical data
- AI models: uncertainty, bias, and probabilistic reasoning

REPAC and AI Data Analysis: An Approach to Managing Uncertainty

RAG (retrieval-augmented generation) & nodegoat:

REPAC uses retrieval-augmented generation (RAG) in the research environment nodegoat so that AI answers are based on retrieved historical data, reducing hallucinations and anchoring the model in controlled, context-rich sources.

Instead of letting a large language model generate answers from what it “remembers” internally (which can be uncertain or lead to hallucinations), REPAC’s system **retrieves relevant data from the structured historical dataset first, and only then feeds that into the AI model**. This means: you define what pieces of data are relevant, the system pulls them, and the AI uses *that context* to generate more precise, grounded results.

Workflow:

<https://lab1100.com/update.s/78/data-and-dialogue-retrieval-augmented-generation-in-nodegoat>

Institute of History, Department of Medieval History

RAG in nodegoat (REPAC) – workflow with vectors

- **Create object texts:** Using *Reversed Collections* (templates), nodegoat turns structured REPAC objects into a controlled textual representation (you decide which fields/relations are included)
- **Generate embeddings (vectors):** Each object text is sent to an LLM endpoint to create a semantic vector (embedding).
- **Store vectors in nodegoat:** The returned vector is stored with the object using *Linked Data Resources + Ingestion Processes* (new value type: “vector”).
- **Embed the user prompt:** When a user enters a prompt, nodegoat creates a vector for the prompt via a preconfigured Linked Data Resource.
- **Vector similarity retrieval:** nodegoat compares the prompt vector with all object vectors and selects the best-matching objects (semantic retrieval).
- **Generate the answer with evidence:** The *texts* of the best-matched objects are collected and sent to the LLM to generate the response - grounded in the retrieved REPAC data (reduced hallucination risk, but prompts must target identifiable content).

nodegoat advantage: relational (object-oriented) data model

Because nodegoat uses a relational, object-oriented data model, AI operates on explicitly defined entities and relationships. This improves retrieval precision, limits ambiguity, and makes AI outputs transparent and reproducible.



Relational tables: **proven way of organising knowledge.** This mathematical table originates from Mesopotamia (ca. 2600–2500 BCE) and shows that relational data structures long predate modern databases.

In this example, **columns 1 and 2 record side lengths,** while **column 3 records the calculated area.**

The meaning emerges not from the values alone, but from their **explicit relations across columns.**

<https://cdli.earth/artifacts/252059>