

Classical DH Methods, Driven by Agents — TEI, NLP, GIS, Network Analysis

2026 DCI Summer Workshop, Day 3 Session 2

Kwok-leong Tang

2026-05-20

Table of contents

Today's session	1
The argument in one paragraph	2
Where this session sits in the workshop	2
Block 1 — Text-side: TEI as input, NLP as analysis	3
TEI in five minutes	3
NLP in five minutes — four jobs, four outputs	4
The pairing — TEI in, NLP out	5
Block 2 — Space-and-relations: GIS as place, network as relations	5
Historical GIS in five minutes	5
Network analysis in five minutes	6
The pairing — a person in CBDB has both a place and a network	6
When the agent is wrong — failure modes you should expect	7
Why you still need to learn the tools	7
Recap	8

Today's session

For thirty years, the field of digital humanities has produced four durable families of methods:
TEI encoding, **NLP analysis** (classification, POS tagging, sentiment, topic modeling),

historical GIS, and **network analysis**. Every Chinese-studies DH paper you have read in the last decade leans on at least one of them. Tools have names — **MARKUS**, **DocuSky**, **CBDB**, **CHGIS**, **Gephi**, **QGIS** — and learning each one used to be a semester of work.

Sessions 02 through 09 of this workshop introduced a different shape: a generative agent that reads your prompt and writes the pipeline for you. The natural reaction is to ask whether the classical methods are now obsolete. The answer this session defends is: **no** — **they are the verifiable substrate the agent runs on, and you still need to know what each one is for**. What changes is the front door.

Note

Teaching goal: After this session you should be able to (a) name the four classical method families and what each produces as output; (b) explain why an agent is the *interface*, not the *replacement*, for each of them; (c) sketch an agent prompt that pairs two methods — TEI + NLP, or GIS + network — against a real corpus you might use.

The argument in one paragraph

A classical DH method gives you a **verifiable artifact**: a TEI file, a part-of-speech tagged sentence, a coordinate on a map, an edge list. Each artifact is auditable — another scholar can open it, check it, cite it, redo the analysis. A generative agent gives you **fluency** — it can read your corpus, draft markup, propose tags, geocode placenames, extract relations. But its output is not auditable until you put it back into a classical form.

The agent is the front door. The classical method is the building you walk into.

Today's two blocks walk you through that door twice — once on the text side (TEI + NLP) and once on the space-and-relations side (GIS + network analysis).

Where this session sits in the workshop

Session	What you got	Day
05	HTML batch tool — calling LM Studio, Google AI Studio, and OpenRouter	Day 2
07	Codex on someone else's wiki — read, improve, query, lint	Day 2
08	Build your own LLM Wiki + Skills	Day 2

Session	What you got	Day
09	RESTful data APIs + the docs → Codex → Skill workflow	Day 3 morning
10	Classical DH methods, with the agent at the front door	Day 3 afternoon (now)

Today reuses the workflow from Session 09 — **docs** → **Codex** → **verify** → **Skill** — and applies it to four method families whose docs are decades older than any LLM.

Block 1 — Text-side: TEI as input, NLP as analysis

TEI in five minutes

The **Text Encoding Initiative (TEI)** is a text-encoding standard with roots in the late 1980s — today an XML vocabulary for marking up texts as *structured representations*. Where a plain .txt of a literary collection is just characters, a TEI version makes the *structure* explicit:

```
<TEI xmlns="http://www.tei-c.org/ns/1.0">
  <teiHeader>
    <fileDesc>
      <titleStmt><title>[your text title]</title></titleStmt>
      <publicationStmt><p>Workshop sample</p></publicationStmt>
      <sourceDesc><p>[edition statement]</p></sourceDesc>
    </fileDesc>
  </teiHeader>
  <text><body>
    <div type="juan" n="1">
      <head>[chapter heading]</head>
      <p>
        <persName ref="cbdb:[id]">[person name]</persName>
        ...
      </p>
    </div>
  </body></text>
</TEI>
```

Three things to notice:

1. **Hierarchy is explicit.** `juan` (卷) wraps `head` and `p`. A computer can find every chapter heading without a regex.
2. **References are stable.** A `cbdb`: ID is a stable identifier, not a name string. Two people who share a name are not confused.
3. **The header is metadata**, not text. Where the file came from, what edition, who encoded it — preserved for citation.

💡 Tip

Why TEI still matters in 2026: it is the closest thing the humanities have to a citable schema. A paper that says “I trained a topic model on TEI-encoded Ming gazetteers” is reproducible. A paper that says “I prompted an LLM on a folder of PDFs” is not.

The Sinology-facing tool you should know by name is **MARKUS** (<https://dh.chinese-empires.eu/markus/>) by Hilde De Weerd’s team — a browser-based tagger that produces TEI-like markup with CBDB and CHGIS linking built in. **DocuSky** (<https://docusky.org.tw>) at NTU is the corresponding corpus-building platform. Both predate the LLM era; both are still maintained.

NLP in five minutes — four jobs, four outputs

The “NLP” you encounter in DH is rarely a single technique. It is four distinct jobs on the same text:

Job	What it produces	Classical tool	What an agent does today
Classification	A label per document (genre, sentiment, dynasty)	scikit-learn, fastText	Reads document, returns label with explanation
POS tagging	A label per token (noun, verb, particle)	jieba, Stanford NLP, spaCy	Tags inline; less accurate on classical Chinese
Sentiment analysis	A polarity / score per passage	Lexicon-based, BERT	Nuanced reading; needs a rubric to be reproducible
Topic modeling	A list of topics, each a weighted bag of words	gensim LDA, Mallet	Summarizes themes; not the same statistical object

The crucial line in that last column: **“not the same statistical object.”** LDA topics come from a probabilistic model fit to word co-occurrence. An LLM’s “topics” are a verbal summary of what it noticed. Both are useful. They are not interchangeable.

The pairing — TEI in, NLP out

Here is the workflow this block trains:

1. **Start with a corpus.** A folder of TEI files, or even plain text you ask Codex to TEI-encode. (Use any of the sources already in your wiki under `~/genai-workshop/`.)
2. **Ask Codex to parse the TEI.** The agent reads the XML, finds the `<persName>` elements, the `<placeName>` elements, the chapter boundaries.
3. **Run one of the four NLP jobs** on the extracted structure. For example: “For each *juan*, give me a topic summary in five Chinese phrases.”
4. **Verify.** Spot-check ten chapters by hand. Does the topic summary track what the chapter is actually about?
5. **Promote the prompt to a Skill** (Session 09 workflow) so you can re-run it across your full corpus.

Warning

Where the agent fails on classical Chinese text - POS tagging on premodern Chinese is consistently worse than on modern Chinese. Don't trust agent-generated POS tags without a sample audit against a known-good tagger. - “Topic” labels for premodern texts often collapse into a few flat categories (官員, 儒學, 戰爭) that hide the actual variation. Ask for *five distinct topics with example phrases*, not “the main topic.” - Named-entity recognition is unreliable when the same character refers to a person in one passage and a placename in the next. Always pair with CBDB / CHGIS lookup.

Block 2 — Space-and-relations: GIS as place, network as relations

Historical GIS in five minutes

A geographic information system stores points, lines, and polygons attached to attributes. **Historical GIS** adds *time*: the polygon for Suzhou prefecture in 1450 is not the polygon for Suzhou prefecture in 1820. The canonical Chinese-studies resource is **CHGIS** (<https://chgis.fas.harvard.edu>), built at Harvard, providing time-aware polygons and gazetteer entries for administrative units from 221 BCE through 1911.

What CHGIS gives you that an agent alone cannot:

- **Stable IDs** for placenames (`hvd_124518` always refers to the same place).
- **Time spans** — when a place exists under each name, when boundaries change.
- **Coordinate authority** — a peer-reviewed centroid, not the agent's best guess from Wikipedia.

The companion tool **TGAZ (Temporal Gazetteer)** at <https://chgis.hudci.org/tgaz/> is the API front-end. Session 09 had you build a Skill against TGAZ or CBDB; if you chose CBDB, the TGAZ Skill from `kltnng/humanities-skills` covers this half.

Network analysis in five minutes

A network is a set of **nodes** (people, places, books, offices) and a set of **edges** between them (kinship, friendship, citation, co-authorship). In Chinese historical research the dominant data source is **CBDB** (<https://projects.iq.harvard.edu/cbdb>), with relation tables for kinship, social, posting, and entry.

The classical workflow is:

1. Query CBDB for the people you care about.
2. Export the relation tables.
3. Load into **Gephi** or **Cytoscape**.
4. Compute centrality, community detection, visualization.

Each step is a chunk of training. Each step is also a step where an agent can drive the tool for you — but the *interpretation* (what does this community detection actually mean for Ming intellectual history?) is still yours.

The pairing — a person in CBDB has both a place and a network

This is the integrating move of the session. A historical Chinese figure has **both** a geography (where they served, where they lived) **and** a network (kin, colleagues, students). The two views are not separate research projects. They are the same person.

Workflow:

1. **Pick a person from your wiki** — anyone with a CBDB ID will do.
2. **Pull places.** Query CBDB for their postings; get CHGIS / TGAZ coordinates for each.
3. **Pull network.** Query CBDB for their kinship and social relations.
4. **Render both.** A map of the career path; a graph of the closest 30 connections.
5. **Cross-read.** Are the people in the network clustered in the places this person served? Or do they cross the geography? *That* is a humanities question worth answering.

Tip

The agent's role here is to write the queries, fetch the data, geocode the placenames against CHGIS, and produce both the map and the graph as files. *Your* role is to look at the two outputs side by side and ask the question that needs asking.

When the agent is wrong — failure modes you should expect

A short catalog of where agentic pipelines fail on classical DH tasks. Memorize the categories; you will see them.

- **Hallucinated TEI tags.** Agent invents element names that look TEI-like but are not in the spec (<bookTitle> instead of <title type="book">). *Defense:* ask Codex to validate the output against the TEI RELAX NG schema; if it can't, you have a hint.
- **Wrong coordinates from ambiguous placenames.** 平陽 has at least four historical referents. Without CHGIS lookup, the agent picks one and moves on. *Defense:* always route placenames through TGAZ; do not let the agent geocode from its own knowledge.
- **Fabricated relations.** Agent extracts a “teacher-student” edge that is not in any source. *Defense:* every edge in the final network must cite a CBDB record or a TEI line; ask the agent to produce a `source_citation` column.
- **Statistical “topics” that are not topics.** Agent presents an LLM-summarized list of themes and calls them “topics.” *Defense:* if the audience expects LDA topics, run gensim — do not substitute. Be explicit about which kind of artifact you produced.
- **Silent dropouts.** Agent processes 5,000 records, says “done,” but skipped 200. *Defense:* ask for an explicit row count in the final report; check against the input count.

Why you still need to learn the tools

A natural question after this session: if Codex can drive MARKUS, Gephi, and QGIS for me, do I need to learn them?

The honest answer is **partial**. You do not need to memorize Gephi’s menu structure. You **do** need to know what betweenness centrality is, what a community detection algorithm assumes, what a TEI element does. The agent can run the tool. It cannot *understand the artifact* for you. When the agent says “I detected three communities in this network,” you need enough background to ask whether the algorithm it chose was appropriate, whether the resolution parameter makes sense, whether the communities are interpretable.

The pattern across the workshop holds here too: **agent as collaborator, not as oracle**. You verify. The tools you should know by name and by purpose:

Family	One tool worth knowing
TEI	MARKUS (browser); the TEI guidelines (https://tei-c.org/guidelines)
NLP	scikit-learn for classification; gensim for topic modeling
GIS	QGIS (free); CHGIS as authoritative data
Network	Gephi; CBDB as authoritative data

Recap

- Four classical DH method families: **TEI**, **NLP**, **GIS**, **network analysis**. Each produces an *auditable* artifact.
- The agent is the **front door** — natural-language interface, pipeline writer, gluer-together — not the replacement.
- Pair the methods: **TEI** + **NLP** for the text-side; **GIS** + **network** for the space-and-relations side. A single agent prompt can drive both halves and produce verifiable outputs.
- Failure modes are predictable: hallucinated markup, ambiguous placenames, fabricated relations, fake “topics,” silent dropouts. Each has a defensive habit.
- Tools you should still learn — by purpose, not by menu: MARKUS, gensim, QGIS, Gephi, plus the authoritative data layers CBDB and CHGIS / TGAZ.