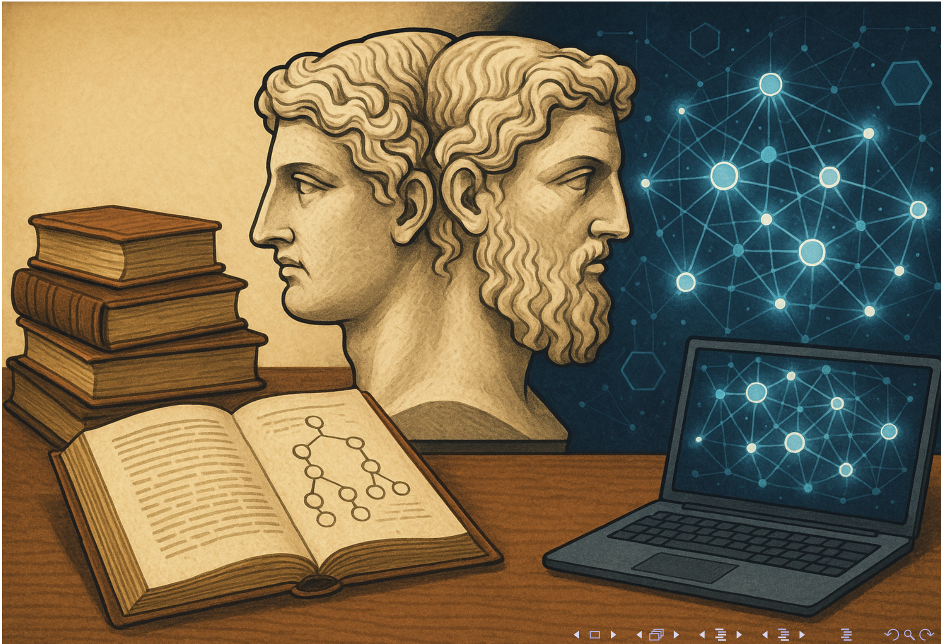
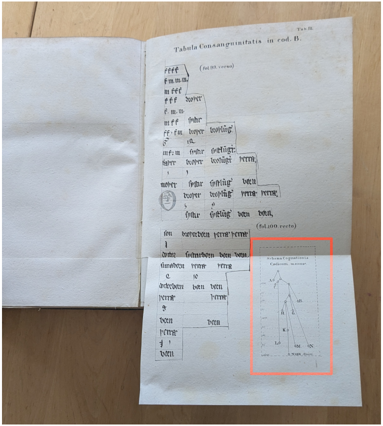
The Age of DH: LLM-mining Pre-stemmatological Philological Literature

Armin Hoenen

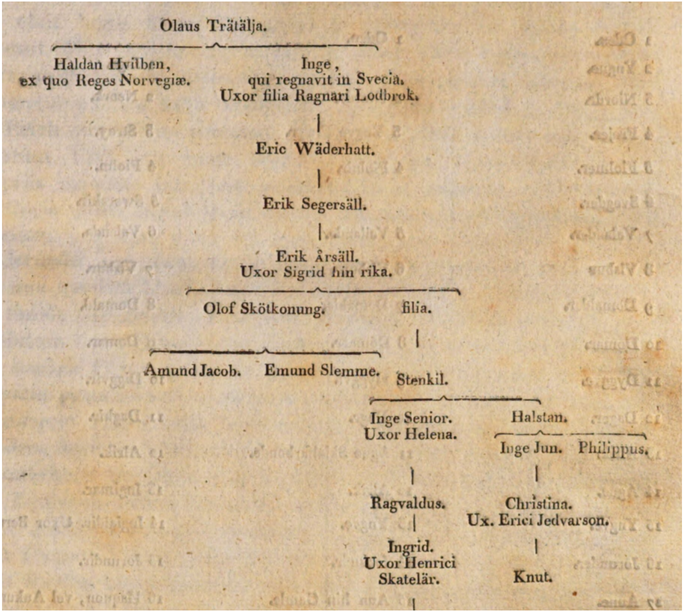
Applying the future to the past?



The first attested stemma.



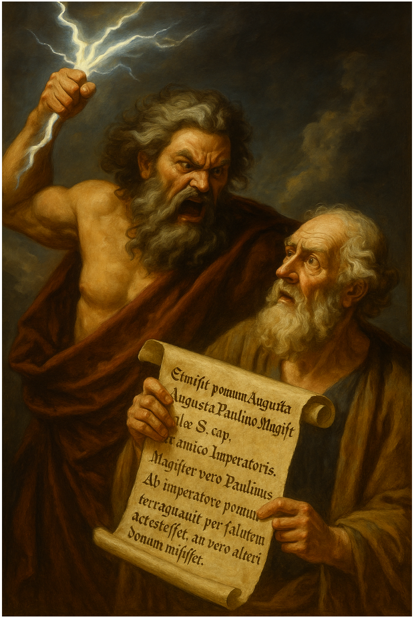
In the source of Collin & Schlyter; Fant 1818.



Transmission Types and Stemma Possibility

Type	**Types of work genesis**
Oral Transmission	performances (not works)
	(Oral-origin)
Written	Mixed (oral + written) origin
transmission	Purely written origin
Print	...
Digital	Stage skipped (earlier stemma known)

"You moroooon"



Transmission Types and Stemma Possibility

Early philological goals in the written age.

- **Normativity**: the ideal text in literary beauty, philosophical content or moral teaching
- **reconstruct original pronunciation** in Hebrew and Chinese in order to understand original semantics and thus moral lessons [hand in hand with exegesis]

A reconstruction of a *possibly flawed original* would not be helpful in pursueing these goals and thus probably contested.

"You moroooooon, start over again!"

Experiment I

Experiment Overview

Experiment 1: Evaluate whether GPT-4o-mini can recognize stemmata in images.

Experiment Overview

Experiment 1: Evaluate whether GPT-4o-mini can recognize stemmata in images.

- Prompted the model with: *"Is there a stemma codicum in this image?"*

Experiment Overview

Experiment 1: Evaluate whether GPT-4o-mini can recognize stemmata in images.

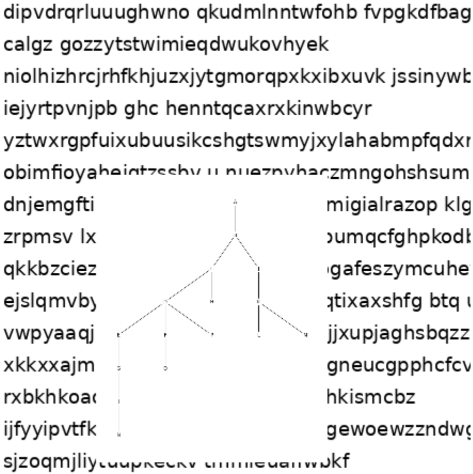
- Prompted the model with: *"Is there a stemma codicum in this image?"*
- Input:
    - 1 family tree
    - 50 synthetic pages with stemmata and text
    - 50 random pages from Collin & Schlyter corpus (no stemmata)
    - 75 additional pages from Collin & Schlyter (1 known stemma)

Experiment Overview

Experiment 1: Evaluate whether GPT-4o-mini can recognize stemmata in images.

- Prompted the model with: *"Is there a stemma codicum in this image?"*
- Input:
  - 1 family tree
  - 50 synthetic pages with stemmata and text
  - 50 random pages from Collin & Schlyter corpus (no stemmata)
  - 75 additional pages from Collin & Schlyter (1 known stemma)
- works through vectorization into text (base64)

Synthetic data



Qualitative Analysis

- The model distinguished between:
  - Ordinary family trees (e.g. genealogical trees)
  - True stemmata (textual transmission diagrams)

Qualitative Analysis

- The model distinguished between:
  - Ordinary family trees (e.g. genealogical trees)
  - True stemmata (textual transmission diagrams)
- Demonstrated contextual understanding of layout and structure

Qualitative Analysis

- The model distinguished between:
  - Ordinary family trees (e.g. genealogical trees)
  - True stemmata (textual transmission diagrams)
- Demonstrated contextual understanding of layout and structure
- No confusion with non-stemma diagrams

Qualitative Analysis

- The model distinguished between:
  - Ordinary family trees (e.g. genealogical trees)
  - True stemmata (textual transmission diagrams)
- Demonstrated contextual understanding of layout and structure
- No confusion with non-stemma diagrams
- in a follow-up: some graphs were misinterpreted as stemmata and some stemmata were overlooked. Some prompt engineering and fine-tuning is required.

Quantitative Results

- **Synthetic Set (100 pages):**
  - 50 with stemmata → all correctly identified ("yes")
  - 50 without stemmata → all correctly rejected ("no")
- **Collin & Schlyter Set (75 pages):**
  - Only 1 page contained THE stemma
  - Model correctly identified everything

Conclusion

- GPT-4o-mini exhibits an inherent understanding of stemmata:
  - Conceptually: differentiates stemmata from similar visuals
  - Visually: accurate recognition across varied sources

Conclusion

- GPT-4o-mini exhibits an inherent understanding of stemmata:
  - Conceptually: differentiates stemmata from similar visuals
  - Visually: accurate recognition across varied sources
- Suggests strong learned representation of textual transmission diagrams

Conclusion

- GPT-4o-mini exhibits an inherent understanding of stemmata:
  - Conceptually: differentiates stemmata from similar visuals
  - Visually: accurate recognition across varied sources
- Suggests strong learned representation of textual transmission diagrams
- Promising result for the detection of stemmata in mass digitized collections, language independently

Conclusion

- GPT-4o-mini exhibits an inherent understanding of stemmata:
  - Conceptually: differentiates stemmata from similar visuals
  - Visually: accurate recognition across varied sources
- Suggests strong learned representation of textual transmission diagrams
- Promising result for the detection of stemmata in mass digitized collections, language independently

Caveat: However LLMs tend to be slow and demanding in hosting. It might be, that one instead want to train one's **own object recognition model**. In 2021 Aouinti et al. in Paris used the predominant Object recognition technology YOLO for the detection of illumination.

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...
- shrunk trees and put them onto random text pages

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...
- shrunk trees and put them onto random text pages
- generated label files indicating object position by rectangle

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...
- shrunk trees and put them onto random text pages
- generated label files indicating object position by rectangle
- trained various yolo models: YOLOv5

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...
- shrunk trees and put them onto random text pages
- generated label files indicating object position by rectangle
- trained various yolo models: YOLOv5
- predicted on all pages of Collin & Schlyter

YOLO since you live only once

Training a yolo model for stemma-recognition. In order to synthetisize training data, I used various native python libraries.

- Synthetisizing trees with python graphviz (1000 train, 100 val set)
- yolo library has a **data augmentation pipeline**: random angle, overlay, ...
- shrunk trees and put them onto random text pages
- generated label files indicating object position by rectangle
- trained various yolo models: YOLOv5
- predicted on all pages of Collin & Schlyter
- best training set was only trees on random text

dipvdrqrluuughwno qkudmlnntwfohb fvpgkdfbag

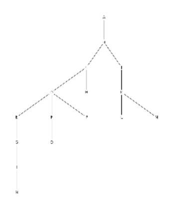
calgz gozzytstwimieqdwukovhyek

niolhizhrcjrhfkhjuzxjytgmorqpxkxibxuvk jssinywb

iejyrtpvnjpb ghc henntqcaxrxkinwbcyr

yztwxrgpfuixubuusikcshgtswmyjxylahabmpfqdxr

obimfioyaheiqtzsshv u nuezpvhaczmngohshsum



dnjemgfti migialrazop klg

zrpmsv lx oumqcfghpkodk

qkkbzciez gafeszymcuhe

ejslqmvby qtixaxshfg btq u

vwpyaaqj jjxupjaghsbqzz

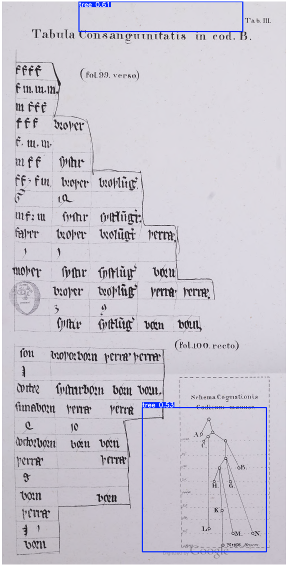
xkkxxajm gneucgpphcfcv

rxbkhkoac hkismcbz

ijfyyipvtfk gewoewzzndwg

sjzoqmjliytuupkeekv tnmiieuanwbkf

Model trained with composite random pages



Potential Impact

The detection of stemmata throughout digitized collections could not only discover preprintoric stemmata independent of language, but also help generate statistics about the distribution of more recent print age stemmata. The available technology is robust enough with LLMs such as GPT4o-mini. Smaller, quicker models such as YOLO are an alternative. For their training synthetic data may already work quite well, plus known images of stemmata for instance from open stemmata. For visually unusual stemmata the LLM may outcompete the YOLO model natively.

Transmission Types and Stemma Possibility

Type	Types of Work Genesis	Print Technology
Print	Mixed (oral and written) origin	page-wise
	Written origin	woodblock
	Print origin	movable
	Mixed written and print origin,...	letters

- editions start to be designed

Experiment II

Narratives (before mid-19^{th} c.) vs. Stemma (after)

Forcedly sequential Narratives:

Narrative 1

"From a now lost root *r*, two lost copies were made: *a* and *e*. *a* was copied twice into *A* and *O*, the latter of which was itself copied into *B* and *C*..." SEQ: r>a&e>A&O>B&C

Narratives (before mid-19th c.) vs. Stemma (after)

Forcedly sequential Narratives:

Narrative 1

"From a now lost root *r*, two lost copies were made: *a* and *e*. *a* was copied twice into *A* and *O*, the latter of which was itself copied into *B* and *C*..." SEQ: r>a&e>A&O>B&C

Narrative 2

"From a now lost root *r*, there were two hypothetical descendants: *a* and *e*. Whereas *e* has one copy, *D*, the subtree of *a* was more complex, leading to *A*, *O*, *B*, and *C*..." SEQ: r>a&e>D>A&O&B&C

Narratives (before mid-19th c.) vs. Stemma (after)

Forcedly sequential Narratives:

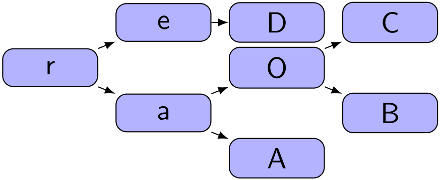
Narrative 1

"From a now lost root r, two lost copies were made: a and e. a was copied twice into A and O, the latter of which was itself copied into B and C..." SEQ: r>a&e>A&O>B&C

Narrative 2

"From a now lost root r, there were two hypothetical descendants: a and e. Whereas e has one copy, D, the subtree of a was more complex, leading to A, O, B, and C..." SEQ: r>a&e>D>A&O&B&C

Freely multi-informative stemma:



Advantage: Visualization (tree) displays all relationships at once, makes instantly comparable.

An experiment?

For comparison and the automatic analyses of large corpora of philological literature, one could ask, are LLMs able to **extract the relations from text** and ultimately stemmata. Not from the alignment, but from the meta-literature (possible also for descriptions of lost works).

An experiment?

For comparison and the automatic analyses of large corpora of philological literature, one could ask, are LLMs able to **extract the relations from text** and ultimately stemmata. Not from the alignment, but from the meta-literature (possible also for descriptions of lost works).

- **Task**: Extract a stemma (a tree) from a textual description of manuscript relationships.
- This is a specialization of relation extraction aimed at manuscript studies.
  - So far no literature in bio-sciences, papers publish the trees
  - relation-extraction for ontologies, species interactions, etc.
- **Goal**: Automatically extract the Newick tree format from text using LLMs.

Experimental Setup: Texts

Generate informationally equivalent textual descriptions which differ in sequence and wording. Input texts are built from ...

- 4 sentences with copy relations (1 for each parent): e.g. 'O was copied twice, resulting in C and B.', O > C,B)

Experimental Setup: Texts

Generate informationally equivalent textual descriptions which differ in sequence and wording. Input texts are built from ...

- 4 sentences with copy relations (1 for each parent): e.g. 'O was copied twice, resulting in C and B.', O $>$ C,B)
- 5 or 7 alternatives for each sentence,

Experimental Setup: Texts

Generate informationally equivalent textual descriptions which differ in sequence and wording. Input texts are built from ...

- 4 sentences with copy relations (1 for each parent): e.g. 'O was copied twice, resulting in C and B.', O $>$ C,B)
- 5 or 7 alternatives for each sentence,
- randomized sequence of the relations, randomized choice of the alternatives

Experimental Setup: Texts

Generate informationally equivalent textual descriptions which differ in sequence and wording. Input texts are built from ...

- 4 sentences with copy relations (1 for each parent): e.g. 'O was copied twice, resulting in C and B.', O > C,B)
- 5 or 7 alternatives for each sentence,
- randomized sequence of the relations, randomized choice of the alternatives
- 100 distinct permutations, 44 of which include distractors at random positions.

Experimental Setup: Texts

Generate informationally equivalent textual descriptions which differ in sequence and wording. Input texts are built from ...

- 4 sentences with copy relations (1 for each parent): e.g. 'O was copied twice, resulting in C and B.', O > C,B)
- 5 or 7 alternatives for each sentence,
- randomized sequence of the relations, randomized choice of the alternatives
- 100 distinct permutations, 44 of which include distractors at random positions.

Additionally, introductory sentences and a final sentence, which carry no relations.

Example Text

Text Description:

This text treats the tradition of Rabanus Testus Textus. The text has been transmitted in handwriting. We have located 5 extant copies in various libraries. e was copied, the copy is D. O and A are closely related, probably they have been copied from the same lost manuscript a. O was not copied from e. archetype r was copied into e and a O has been copied into B and C. The tradition is thus a limited one in size and scope but the relations are quite clear leading to a wonderful stemma albeit with descriptii and chains of hypothetical nodes.

Expected Tree:

$$r(e(D), a(O(C, B), A))$$

Prompt Engineering for GPT4-o

System role: You are a philologist...

Prompt Engineering for GPT4-o

System role: You are a philologist...
Zero-shot:

- Extract the Newick tree from this text:<text>

Prompt Engineering for GPT4-o

System role: You are a philologist...
Zero-shot:

- Extract the Newick tree from this text:<text>

One-shot: iteration of zero-shot with fake text and assistant stemma

Prompt Engineering for GPT4-o

System role: You are a philologist...
Zero-shot:

- Extract the Newick tree from this text:<text>

One-shot: iteration of zero-shot with fake text and assistant stemma

Chain of thought (CoT) Prompting

- Step 1: Extract the relations between witnesses mentioned in this text:<text>
- Step 2: Generate a tree in Newick fromat from:<res:step1>

Prompt Engineering for GPT4-o

System role: You are a philologist...
Zero-shot:

- Extract the Newick tree from this text:<text>

One-shot: iteration of zero-shot with fake text and assistant stemma

Chain of thought (CoT) Prompting

- Step 1: Extract the relations between witnesses mentioned in this text:<text>
- Step 2: Generate a tree in Newick fromat from:<res:step1>

In-context Learning:

- Added: For instance 'The archetype x has been copied into f and B. C and D are copies from B.' should result in x(f,B(C,D)).

Prompt Engineering for GPT4-o

System role: You are a philologist...
Zero-shot:

- Extract the Newick tree from this text:<text>

One-shot: iteration of zero-shot with fake text and assistant stemma

Chain of thought (CoT) Prompting

- Step 1: Extract the relations between witnesses mentioned in this text:<text>
- Step 2: Generate a tree in Newick fromat from:<res:step1>

In-context Learning:

- Added: For instance 'The archetype x has been copied into f and B. C and D are copies from B.' should result in x(f,B(C,D)).

Format augment:

- Provide only the tree

Well done

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering
- In-context learning/few-shot significantly improved performance.

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering
- In-context learning/few-shot significantly improved performance.
- Chain-of-Thought prompting likewise improved performance.

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering
- In-context learning/few-shot significantly improved performance.
- Chain-of-Thought prompting likewise improved performance.
- Presence of distractors did not significantly affect accuracy.

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering
- In-context learning/few-shot significantly improved performance.
- Chain-of-Thought prompting likewise improved performance.
- Presence of distractors did not significantly affect accuracy.
- The format augment worked extremely well

Results

Method	Hits	Misses	Accuracy (%)
Zero-shot	6	94	0.06
One-shot	89	11	0.89
CoT	9	81	0.09
Zero-shot with in-context	67	33	0.67
One-shot with in-context	94	6	**0.94**
CoT with in-context	82	18	0.82

Table: Performance of different approaches

Key Findings:

- cot-1shot and cot-1shot-incontext failed (0.08, 0.12) due to prompt engineering
- In-context learning/few-shot significantly improved performance.
- Chain-of-Thought prompting likewise improved performance.
- Presence of distractors did not significantly affect accuracy.
- The format augment worked extremely well
- With incontext or one-shot, almost all examples adhered to format from examples

Experiment III

Case Study: Chronicon Alexandrinum

- A Greek chronicle spanning Creation to Byzantine Emperor Heraclius
- Published in Latin by Matthaeus Raderus (Munich, 1615)
- Includes sources like Eusebius, Africanus, Epiphanius, etc.

Case Study: Chronicon Alexandrinum

- A Greek chronicle spanning Creation to Byzantine Emperor Heraclius
- Published in Latin by Matthaeus Raderus (Munich, 1615)
- Includes sources like Eusebius, Africanus, Epiphanius, etc.

Prompting excerpt: *Correcting small OCR inconsistencies, analyze this text and extract from the text any relations between manuscripts or versions in a structured machine readable way.*

Case Study: Chronicon Alexandrinum

Example

*Etmifit*pomumAugustaEudocia,AugustaPaulinoMagiftroAula S.cap.*
&amico Imperatoris. Magiſter vero Paulinus cum ignoraffet xv1. ab ImperatorepomumfuiſſeprimumAug.donatum,Augusto Theodofio (velut nouumdonum)remiſit,&quandoegreſſus estètpoſt biduű

Normalized Version:

Et misit pomum Augusta Eudocia, Augusta Paulino Magistro Aulae S. cap.
& amico Imperatoris. Magister vero Paulinus cum ignorasset XVI ab Imperatore pomum fuisse primum Aug. donatum, Augusto Theodosio (velut novum donum) remisit, & quando egressus est et post biduum

LLM GPT4o-mini

- Handled OCR inconsistencies and non-standard characters e.g. I) = D

LLM GPT4o-mini

- Handled OCR inconsistencies and non-standard characters e.g. I) = D
- handled Latin

LLM GPT4o-mini

- Handled OCR inconsistencies and non-standard characters e.g. I) = D
- handled Latin
- Summarized complex scholarly commentary

LLM GPT4o-mini

- Handled OCR inconsistencies and non-standard characters e.g. I) = D
- handled Latin
- Summarized complex scholarly commentary
- Extracted manuscript and edition relationships

LLM GPT4o-mini

- Handled OCR inconsistencies and non-standard characters e.g. I) = D
- handled Latin
- Summarized complex scholarly commentary
- Extracted manuscript and edition relationships
- Produced a structured, machine-readable representation

Extracted Stemma Summary

Root: Lost Greek original

Branch 1 – Sicilian MS:

- Surita → Agustín → Panvinio → Casaubon → Scaliger

Branch 2 – Bibliotheca Augustana MS:

- Sylburgius → Casaubon → Scaliger → Raderus

Machine-Readable Output


```
{
  "chronicle_name": "Chronicon Alexandrinum",
  "early_sources": ["Eusebius", "Africanus", ...],
  "early_transmission": [
    {"manuscript_location": "Sicily", "handlers": [...]},
    {"manuscript_location": "Bibliotheca Augustana", ...}
  ],
  "printed_editions": [...],
  "stemma_summary": {...}
}
```

Future work

The relation extraction enables us to do research into the reconstruction of the prehistory of the stemma:

- Develop a pipeline where LLMs read millions of scanned book pages

Future work

The relation extraction enables us to do research into the reconstruction of the prehistory of the stemma:

- Develop a pipeline where LLMs read millions of scanned book pages
- Automatically detect and extract stemmatic (textual transmission) relationships

Future work

The relation extraction enables us to do research into the reconstruction of the prehistory of the stemma:

- Develop a pipeline where LLMs read millions of scanned book pages
- Automatically detect and extract stemmatic (textual transmission) relationships
- Apply to large corpora like Corpus GB

Future work

The relation extraction enables us to do research into the reconstruction of the prehistory of the stemma:

- Develop a pipeline where LLMs read millions of scanned book pages
- Automatically detect and extract stemmatic (textual transmission) relationships
- Apply to large corpora like Corpus GB
- apply to series of editions of the same ancient work (for instance Lucretius)

Future work

When was the time right for the stemma?

- was the amount of discussion on variation decisive

Future work

When was the time right for the stemma?

- was the amount of discussion on variation decisive
- is there evidence that Scaliger in Europe was as important as is sometimes assumed

Future work

When was the time right for the stemma?

- was the amount of discussion on variation decisive
- is there evidence that Scaliger in Europe was as important as is sometimes assumed
- were stemmata given for certain works, geographical areas, in certain times

Future work

When was the time right for the stemma?

- was the amount of discussion on variation decisive
- is there evidence that Scaliger in Europe was as important as is sometimes assumed
- were stemmata given for certain works, geographical areas, in certain times

Reconstructing the full history of the stemma, its prehistory, its occurrence and its recent past has new possibilities. LLMs are here to stay.

Conclusion

Conclusion

I have presented 3 small, PoC-experiments for the detection and analysis of stemmata in digitized collections using LLMs and YOLO.

- image analysis with LLM or YOLO: stemma or not
- witness relation and stemma extraction on synthetic texts
- witness and relation extraction on authentic text with imperfect OCR in Latin

Retracing the history of the stemma seems possible!

Farewell

Farewell

Thank you very much for your kind attention. Grazie mille per Vostra gentile attenzione. Vielen Dank für Ihre werte Aufmerksamkeit.