Detecting Latin in Historical Books with Large Language Models: A Multimodal Benchmark

Yu Wu* and **Ke Shu*** and **Jonas Fischer*** and
Lidia Pivovarova and **David Rosson** and **Eetu Mäkelä** and **Mikko Tolonen**
University of Helsinki
{firstname.lastname}@helsinki.fi

Abstract

This paper presents a novel task of extracting low-resourced and noisy Latin fragments from mixed-language historical documents with varied layouts. We benchmark and evaluate the performance of large foundation models against a multimodal dataset of 724 annotated pages. The results demonstrate that reliable Latin detection with contemporary zero-shot models is achievable, yet these models lack a functional comprehension of Latin. This study establishes a comprehensive baseline for processing Latin within mixed-language corpora, supporting quantitative analysis in intellectual history and historical linguistics. Both the dataset and code are available at https://github.com/COMHIS/EACL26-detect-latin.

1 Introduction

Accurate language identification at a granular level within historical documents is a key component to the study of the early modern period at scale. Latin, as the primary written language of Western Europe for more than a millennium, has a unique position, gradually ceding to vernaculars at varying paces across regions and genres (Marjanen et al., 2025). Throughout this transition, Latin fragments frequently appeared within predominantly vernacular texts, in quotations, specialist terminology, and instances of code-switching. Automated extraction of diverse Latin uses in context from historical corpora is crucial to studying language evolution, the interplay between classical and modern thought, and the dissemination of ideas (Sprugnoli et al., 2024; Gorovaia et al., 2024; Perrone et al., 2021; Burns et al., 2021). The underlying research interest behind this paper is thus to enable quantitative, fragment-level measurement of Latin's presence in 18th-century British print, so historians and linguists can trace vernacularization across time and genres. However, this task poses challenges due to wide variations in Latin usage, scripts, complex page layouts, and inconsistent print and scan quality in historical book databases.

This study focuses on detecting instances of Latin within the *Eighteenth Century Collection Online* (ECCO) (Tolonen et al., 2022) corpus using both book page images and the corresponding text extracted by Optical Character Recognition (OCR). The lack of an existing dataset specifically designed for multimodal and code-mixed Latin detection motivated us to create an annotated dataset for this purpose. Our dataset contains 724 pages sampled from historical documents, validated by specialists in 18th-century publishing culture to represent diverse use cases, with our novel, fine-grained typology of 12 categories. In future work at ECCO scale, our focus on reliable detection and page-level quantification will support temporal trend analyses (e.g., decade-by-decade decline, genre-specific persistence, bilingual presentation with English glossing) and reconstructing the footprint of learned discourse in print culture. While we focus on Latin due to its aforementioned importance for historical study, our fully benchmarked, manually annotated scenario provides a solid template for extending the method to other languages as well.

Given all the complex nature of the task, we explore the capabilities of modern Large Language Models (LLMs), including multimodal models (MLLMs) for this task. The way these models handle contextual information, recognize patterns in noisy data, and integrate textual cues with visual layout information has been found to help disambiguate text and languages in historical documents (Luo et al., 2024; Boros et al., 2024; Kanerva et al., 2025; Xie et al., 2025) compared to traditional Natural Language Processing (NLP) methods. Our investigation exploits the new dataset to test a number of state-of-the-art models and finds that reliable Latin detection in such challenging his-

*These authors contributed equally.

Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics
Volume 1: Long Papers, pages 5305–5328

torical material is achievable. The benchmarking of different model architectures provides insight into their strengths and weaknesses when faced with the complexities inherent in the data. This work establishes a strong baseline for a novel NLP task and highlights the need for more modality and semantic-aware approaches, as well as robust evaluation frameworks in historical text analysis.

The main contributions of this article are:

- Defining the novel multimodal task of Latin detection in historical documents.
- Creating a new benchmark dataset of diverse Latin usage in 18th-century books.
- Developing a robust evaluation framework tailored to the multimodal challenges of the task.
- Providing a practical Latin detection pipeline that enables large-scale downstream applications in historical research.
- Systematically benchmarking contemporary LLMs for this task.

2 Problem Definition

We define the task of *Latin language detection in historical documents* as a two-stage classification and extraction problem, where the input consists of a scanned page image and/or its OCR transcription. The task is to automatically detect whether any segments in the text are written in Latin, and if so, to extract text of those specific segments.

Formally, given a document page D, let I_D denote its image and T_D denote its OCR-processed text. A system must perform the following two subtasks:

- **Task 1 (Page-level Latin Detection):** Predict a binary label $y_D \in \{0, 1\}$, where $y_D = 1$ indicates that the page contains at least one segment in Latin, and $y_D = 0$ otherwise.
- **Task 2 (Latin Segment Extraction):** If $y_D = 1$, extract a list of text spans $S_D = [s_1, s_2, \ldots, s_n]$, where each $s_i \in T_D$ is a contiguous Latin segment string.

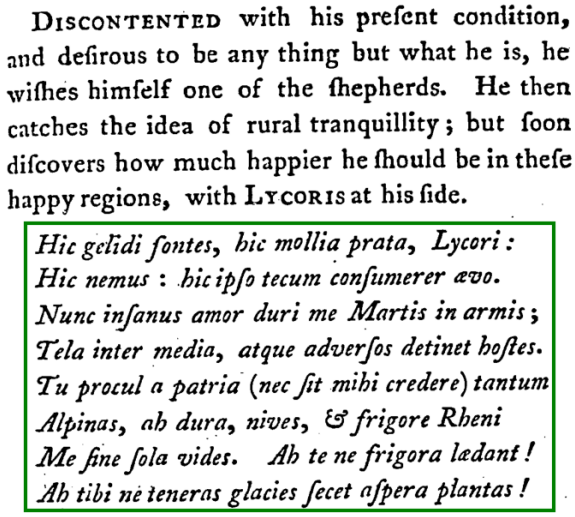
We structure our problem into two tasks for a more comprehensive evaluation. The core challenge, Task 2, is the fine-grained extraction of Latin text segments, evaluated using per-page token precision and recall. Since these metrics inadequately handle pages with no Latin, we introduce Task 1, a page-level binary classification, to assess performance on these non-Latin instances specifically. We require extracted segments as strings rather than image regions, as this output format better aligns with the capabilities of most MLLMs and enables a simpler, more direct comparison across different input modalities. Detailed definitions of our metrics are provided in Section 5.

3 Related work

Latin in NLP Given its historical importance, Latin has attracted considerable attention within the NLP community (e.g., Sprugnoli et al., 2024; Schulz and Keller, 2016; Gorovaia et al., 2024; Perrone et al., 2021; Burns et al., 2021), though much of this research has centered on small, clean corpora of ancient literary texts. While some recent studies have ventured into Early Modern mixed-language documents (Stüssi and Ströbel, 2024; Volk et al., 2024), these also predominantly rely on manually curated and annotated data. In contrast, our work focuses on the foundational task of Latin *discovery*: detecting Latin within extensive, unedited, and noisy digitized collections like ECCO (Tolonen et al., 2022). This computational approach aims to detect Latin in a vast corpora, while the identified fragments can subsequently be analyzed using a range of established NLP tools developed for classical languages (Johnson et al., 2021; Burns, 2023; Straka and Straková, 2020; Kupari et al., 2024).

Code-mixed Language Detection From a methodology perspective, identifying Latin segments within historical publications is a code-mixed language detection task (Aguilar et al., 2020). While extensive research in this area has focused on contemporary informal texts (Barman et al., 2014; Zhang et al., 2018), its application to historical documents, with challenges like archaic syntax, lexicon, and spelling, has been less explored (Schulz and Keller, 2016; Volk et al., 2022). Detecting classical languages in these complex historical contexts has traditionally involved rule-based systems and supervised machine learning approaches, notably Conditional Random Fields (CRFs) (Schulz and Keller, 2016; Sterner and Teufel, 2023; Volk et al., 2022). Alongside these, robust statistical tools like Lingua (Stahl, 2021) offer effective general language identification with support for mixed language. Given the recognized potential of modern LLMs to navigate linguistic nuances and noisy data, our work investigates their capacity to enhance detection performance.

LLMs for Historical Documents Recent LLMs, particularly Multimodal variants (MLLMs), have


DISCONTENTED with his preſent condition, and deſirous to be any thing but what he is, he wiſhes himſelf one of the ſhepherds. He then catches the idea of rural tranquillity; but ſoon diſcovers how much happier he ſhould be in theſe happy regions, with LYCORIS at his ſide.

Hic gelidi fontes, hic mollia prata, Lycori:
Hic nemus: hic ipſo tecum conſumerer ævo.
Nunc inſanus amor duri me Martis in armis;
Tela inter media, atque adverſos detinet hoſtes.
Tu procul a patria (nec ſit mihi credere) tantum
Alpinas, ah dura, nives, & frigore Rheni
Me fine ſola vides. Ah te ne frigora lædant!
Ah tibi ne teneras glacies ſecet aſpera plantas!

Figure 1: An example of an annotated Latin fragment and surrounding context.

shown considerable potential in historical document analysis, demonstrating top performance in tasks like OCR, named entity recognition, and general document understanding from historical sources (Bai et al., 2025; Luo et al., 2024; Boros et al., 2024; Kanerva et al., 2025; Backer and Hyman, 2025; Xie et al., 2025), and in assessing general historical knowledge (Hauser et al., 2024). Despite these advancements, a significant gap persists for more specialized, complex applications. Specifically, there is a notable lack of dedicated benchmarks and systematic exploration for the fine-grained, page-level multimodal detection and extraction of embedded secondary languages (e.g., Latin) (Aguilar et al., 2020; Guzmán et al., 2017). This task is demanding due to noisy scans from historical archives, diachronic language context, and orthographic variation (Volk et al., 2022). Our work contributes to this underexplored area by introducing a systematic evaluation methodology designed to be scalable also to other languages.

4 Dataset

4.1 Sampling and Annotations

Our approach to dataset construction began with a targeted sampling strategy to identify pages with a high likelihood of containing Latin text. We queried the Reception Reader database (Rosson et al., 2023), which indexed text reuses across the ECCO corpus using noise-resistant detection methods. From this, we randomly selected 800 reuse instances where one book was cataloged as Latin and the other as non-Latin. To ensure broad representation and reflect the diversity of the ECCO collection (approximately 200,000 books), each sampled page was drawn from a different book, covering varied publication dates and topics. However, ECCO's language metadata is book-level, meaning that "Latin" books often contain significant non-Latin text like English introductions. Also, the reuse offsets only mark the textual overlap without specifying the language of the text segment.

These pages were then manually annotated. Annotators were tasked with drawing bounding boxes around all Latin fragments on the page images (see example in Figure 1). These visual annotations are later reliably mapped to text offsets (locations within a string) using ECCO's OCR positional data for ground truth text extraction. Our annotation guidelines stated marking all instances of Latin text semantically used as Latin. This included single Latin words if presented with explanations in a dictionary, as well as Latin found in headlines, editorial annotations, or footnotes.

The annotation environment was Label Studio (Tkachenko et al., 2020-2025). The primary annotation was performed by three scholars familiar with Latin. Following this, an expert in historical texts meticulously reviewed and validated all annotations to ensure accuracy and consistency. During the process, the annotators veto the pages with incompletely OCR-transcribed regions.

4.2 Dataset Characteristics

In total, 724 pages were annotated, with 594 identified as containing Latin. An expected finding during annotation was the frequent presence of other languages, such as French, German, and Greek, highlighting the dataset's challenging multilingual nature beyond simple Latin-English code-mixing.

To contextualize model performance and to better understand the dataset's composition, we divided the annotated Latin segments into 12 language integration categories. Each category represents a specific way in which Latin is used in 18th-century British books and how it relates to English-language text. Depending on their content, all Latin text segments were assigned to one or multiple categories, with some frequently appearing together (e.g., footnotes and code-switching), while others are mostly exclusive (e.g., bilingual). This novel categorization enables a fine-grained analysis of both historical linguistic practices and the performance of our Latin detection approach within different contexts of language integration.

Table 1 shows all the categories and the fre-

	Category	Count
1.	Direct Quote	258
2.	Independent	196
3.	Footnote	191
4.	Code-switching	100
5.	Bilingual	55
6.	Emblematic	34
7.	Indices and Catalogs	30
8.	Legal	30
9.	Ecclesiastical	23
10.	Tables and Charts	11
11.	Dictionary	9
12.	Side-note	8

Table 1: Page counts by Latin segment category.

quency of each annotation category within our dataset. The full definitions of 12 categories are listed in Appendix A.1, and categorized segment examples are shown in Appendix A.2.

5 Evaluation Setting

5.1 OCR Post-Correction and Normalization

Evaluating models on the ECCO corpus is complicated by significant OCR quality discrepancies: modern models with vision capabilities often produce cleaner text than ECCO's original OCR, while text-based models may or may not replicate the noise in their input. Such differences make direct string-based comparisons problematic and distort evaluation. To ensure meaningful assessment across all model types, we post-correct both the ground-truth Latin segments and the full input page texts. This OCR post-correction is performed using the OpenAI o1 model (Jaech et al., 2024) with a specialized prompt from (Kanerva et al., 2025).

Even after the post-correction, residual noise and other variation still remain in the data. Thus, for token-based evaluation, we apply a more traditional rule-based preprocessing pipeline to both predicted and reference strings. This deterministic pipeline, informed by our extensive experience with OCR data and domain-specific knowledge, targets common superficial textual variations and ensures a fair alignment. The pipeline includes Unicode normalization, ligature replacement, lowercasing, digit removal, de-hyphenation, and punctuation stripping. Subsequent to these cleaning operations, the strings are tokenized into word-level units. More details on the processing steps are presented in Appendix B.1.

5.2 Metrics

The goal of Task 1 is to detect whether a page has Latin on it. We measure this by reporting precision, recall, and F1 score in percentage, along with the F1 score for non-Latin pages to ensure balanced evaluation. To evaluate the Latin segment extraction performance in Task 2, we calculate precision, recall, and F1 score in percentage based on token-level matches between model predictions and the ground truth of the page. A fuzzy matching mechanism is applied to pair predicted and reference tokens one by one. A match is considered valid if the token-level edit distance is not larger than a tunable threshold proportion θ compared to the ground-truth token length. This approach provides a more flexible and robust evaluation than exact token matching by tolerating minor textual differences at the token level, such as lexical variations and OCR-induced distortions. The pseudocode of the fuzzy matching algorithm is shown in the Appendix B.2. Overall metrics are averaged across the full evaluation set.

6 Latin Extraction Pipeline with LLMs

Our evaluation investigates the application of general instruction-following LLMs, particularly multimodal variants, for Latin segment extraction from historical documents. We propose a unified, prompt-based pipeline designed to be both practical for real-world deployment and robust for systematically and fairly evaluating the capabilities of diverse LLMs on this task.

Unified Prompting Strategy We employ a minimal, high-level instructional prompt designed to elicit responses that inherently address both subtasks within one simply formatted output. This approach simplifies interaction with the models and the subsequent processing of their outputs, thereby contributing to the overall ease of application.

This unified prompt asks the LLM to extract all Latin segments to a list, without further instructions. The distinction in our experiments lies solely in the input provided to this consistent prompt, where the specific prompts are shown in the Appendix C:

- **Text-only:** The OCR-extracted and post-corrected text, appended to the prompt.
- **Image-only:** The page image, with the prompt guiding it to the visual signal.
- **Multimodal:** Both the scanned page image and the corrected OCR text are included.

Structured Output and Postprocessing The LLMs are instructed to output their predictions as a list of Latin segments, which directly corresponds to the output requirement for Task 2. The presence of a non-empty list implicitly indicates the presence of Latin script on the page (Task 1, $y_D = 1$), while an empty list indicates its absence ($y_D = 0$).

Model-Agnostic Compatibility Because the method does not rely on any model-specific architecture or training, it can be directly applied to a wide range of general-purpose foundation language models. This makes the approach particularly suitable for scalable deployment across large historical corpora with variable OCR quality and image-text alignment conditions.

7 Experiments

7.1 Experiment Setup

Model Selection To explore how modern instruction-tuned language models handle the new task of Latin segment detection and extraction in noisy multimodal historical documents, we benchmark a representative suite of LLMs across modalities, scales, and architectures. The model selection follows three guiding principles: (i) in the absence of dedicated multimodal benchmarks for historical language understanding in documents, we refer to leaderboard performance on **DocVQA** (Tito et al., 2021) and comprehensive open evaluations such as **OpenCompass** (Contributors, 2023) and **MMMU** (Yue et al., 2024); (ii) we prioritize lightweight to medium-scale models (7B-72B) to better reflect realistic research use cases in historical academic and limited-resource scenarios. Specifically, the selected models include:

- **GPT-4.1** (Achiam et al., 2023): proprietary frontier MLLM accessed via the OpenAI API, included as a strong reference point for the performance without explicit thinking mode.
- **Qwen2.5-VL series** (72B, 32B, 7B) (Bai et al., 2025): open-source flagship MLLMs with strong document understanding and visual grounding. We include both Vision-Language and text-only instruction-tuned variants to disentangle the multimodal inputs.
- **Qwen3 series** (235B-A22B, 30B-A3B, 32B, 14B, 4B) (Team, 2025): latest generation of Qwen text models with redesigned architecture and built-in *thinking* mode. We evaluate them to probe how reasoning-augmented LLMs transfer to the historical language task.
- **DeepSeek-R1 variants** (original and distilled) (Guo et al., 2025): pioneering reasoning-centric LLMs. We use the original large R1 to gauge frontier open-source LLMs, and distilled versions (based on Llama-3.3-70B and Qwen2.5-32B) to examine transfer of reasoning signals into smaller models.
- **InternVL3 series** (38B, 14B, 8B) (Zhu et al., 2025): top academic MLLMs with a two-stage visual encoder integrated into a transformer backbone.
- **Gemma3** (27B) (Team et al., 2025): new open-source MLLM from Google, optimized for efficiency and multilingual capability.

Baseline We employ **Lingua** (Stahl, 2021), a statistical language identifier based on character n-gram modelling, and the only off-the-shelf tool we found that supports token-level Latin detection in mixed-language text. While not designed for noisy OCR, it offers a practical baseline to contextualize the difficulty of our task and the potential advantages and drawbacks of LLM-based approaches. The configuration details are in Appendix D.

Implementation Details All models except GPT-4.1 are run on the supercomputer's AMD MI250X GPU nodes via the local `vLLM` (Kwon et al., 2023) server, using a maximum batch size of 16 with asynchronous requests and a Ray backend. Depending on model size, we allocate 1–6 nodes, totalling about 8 GPU hours per model on average. We use deterministic generation with a temperature set to 0 and a fixed seed to ensure reproducibility. We limit the output token length to 20k tokens. For models supporting thinking mode, we additionally enforce a thinking budget of 15k tokens to prevent looped or unbounded reasoning traces. The edit-distance threshold θ is set to 0.2 based on empirical evidence, discussed further in the Appendix E.3.

7.2 Model Results and Analysis

Table 2 presents the overall results on the two tasks. The traditional Lingua baseline remains competitive but is consistently surpassed by recent foundation models. Notably, several large open-source LLMs such as DeepSeek-R1 and Qwen3 not only outperform Lingua but also exceed multimodal GPT-4.1 across both tasks, highlighting the rapid progress of open-source development. The different strengths of DeepSeek and Qwen2.5-VL also reveal trade-offs between tasks and metrics, suggesting that thinking trace may shift error profiles.

Model Details				Page-level (Task 1)				Token-level (Task 2)		
Model	**Variant**	**Size**	**Mode**	**F1**	**Precision**	**Recall**	**NL. F1**	**F1**	**Precision**	**Recall**
Lingua	-	-	T	92.58	86.32	**99.83**	43.11	77.14	77.31	80.07
GPT-4.1	-	-	I+T	96.03	92.51	**99.83**	77.00	84.59	88.49	83.54
Deepseek-R1	-	671B	T	97.53	95.34	**99.83**	87.07	**87.00**	**89.69**	86.17
	Distill-Llama	70B	T	96.18	92.94	99.66	78.34	84.26	87.81	83.62
	Distill-Qwen	32B	T	95.35	92.55	98.32	74.44	82.84	86.14	82.66
Qwen3	MoE-A22B	235B	T	96.90	94.13	**99.83**	80.52	86.54	89.16	86.00
	MoE-A3B	30B	T	95.90	93.45	98.48	78.07	84.29	87.29	83.73
	-	32B	T	95.39	93.25	97.64	75.86	84.07	86.46	83.98
	-	14B	T	96.61	94.96	98.32	82.85	84.78	88.10	83.71
	-	4B	T	88.45	87.36	89.56	43.27	75.38	76.30	77.20
Qwen2.5	VL	72B	I+T	98.58	98.16	98.99	93.33	83.87	86.98	82.74
	VL	72B	I	**98.82**	98.66	98.99	**94.57**	80.00	82.36	79.72
	-	72B	T	97.85	**100.00**	95.79	91.23	80.95	85.31	79.33
	VL	32B	I+T	96.18	92.94	99.66	78.34	84.32	86.90	83.99
	VL	32B	I	96.80	94.40	99.33	82.97	79.82	82.64	79.18
	-	32B	T	97.26	99.13	95.45	88.65	81.57	84.46	80.99
	VL	7B	I+T	88.36	82.89	94.61	15.91	72.41	78.62	72.23
	VL	7B	I	95.75	96.74	94.78	81.62	71.42	76.99	70.39
	-	7B	T	91.86	92.49	91.25	64.18	60.52	65.78	64.57
InternVL3	-	38B	I+T	95.62	92.32	99.16	75.00	84.24	84.43	**86.67**
	-	38B	I	89.51	87.22	91.92	43.86	56.53	61.11	55.61
	-	14B	I+T	94.28	90.42	98.48	65.70	81.36	82.63	83.87
	-	14B	I	90.16	87.86	92.59	47.37	53.51	56.46	55.08
	-	8B	I+T	90.54	86.39	95.12	41.00	69.04	65.87	82.10
	-	8B	I	90.82	88.50	93.27	50.88	60.22	60.30	65.63
Gemma3	-	27B	I+T	90.77	83.57	99.33	18.92	82.50	84.00	84.79
	-	27B	I	88.42	82.60	95.12	12.94	60.03	62.78	61.68
	-	27B	T	94.22	90.03	98.82	64.36	83.79	86.09	84.20

Table 2: Experimental results on selected LLMs, compared with Lingua baseline. "**NL. F1**" denotes the F1 score for identifying Non-Latin pages. "I" and "T" indicate image and text input separately.

Vision-language models further demonstrate the potential benefit of multimodal input for OCR-heavy tasks, though vision-only settings remain challenging. Overall, the open-source LLMs with thinking mode now define the frontier for this task. Future MLLMs are expected to feature more deeply integrated reasoning capabilities.

Model scale is a key driver of performance, particularly within the same model family. Larger models have been shown to more effectively memorize and generalize low-resource language phenomena, consistent with neural scaling laws (Gordon et al., 2021; Kaplan et al., 2020). However, scale alone does not guarantee superior results. Our findings not only indicate some performance saturation among larger models (above 30B), but also highlight the potential of thinking-enabled models, where even relatively small Qwen3 variants capture sufficient knowledge to rival or surpass much larger counterparts, thanks to explicit reasoning traces. Besides the thinking mode, architectural and multimodal training strategies play a central role. For instance, Qwen3-A22B and A3B exploit a Mixture-of-Experts (MoE) design to combine high

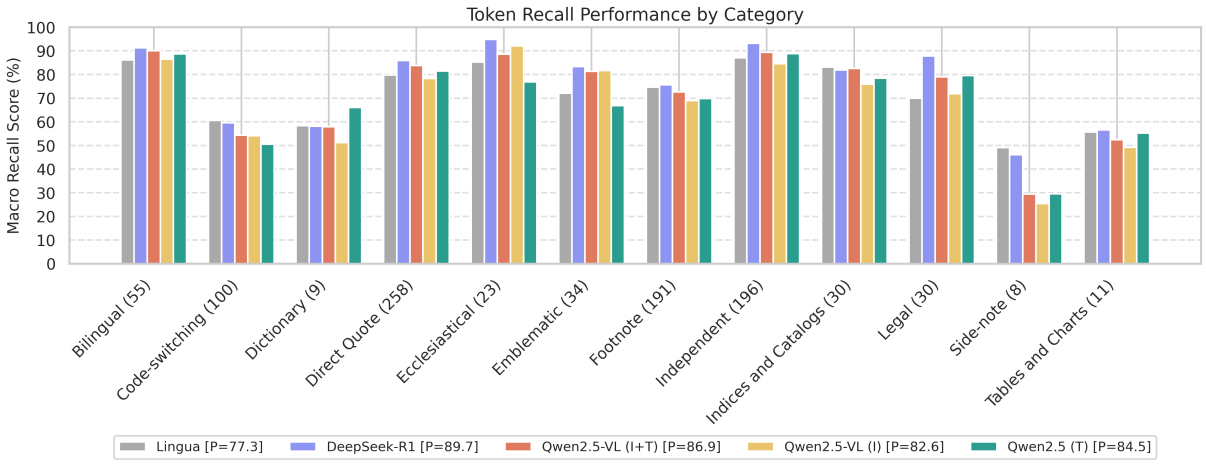


Figure 2: Macro token recall statistics on different category labels for 5 top-performing models. Qwen2.5 models are all with 32B parameters. The number of page instances with each label tagged is shown in parentheses. Values in the legend indicate each model's token precision from Table 2 to provide complementary performance context.

reasoning efficiency with strong accuracy. Likewise, when comparing Qwen2.5-VL-7B against InternVL3-8B, which shares the same language model backbone, the Qwen variant proves more robust on visual signals.

Multimodal inputs (I+T) generally improve performance, but outcomes vary with models. In the InternVL3 series and Gemma3, performance with image-only input lags far behind multimodal input for the same models, reflecting a limited ability to derive the required information solely from images. A different, even worse issue arises in Gemma3, where multimodal fusion degrades performance compared to text-only, likely due to overreliance on noisy visual features. By contrast, Qwen2.5-VL shows superior integration, successfully featuring a visual-only capability powerful enough to match text-only performance, and complementing text robustness with accurate visual cues. These mixed results underline both the difficulty of historical OCR and image-based Latin extraction, and the importance of our dataset for testing multimodal performance with preprocessed OCR text input.

Finally, we address the issue of robustness in rejecting non-Latin pages. Beyond standard detection metrics, the **NL. F1** column reveals a divergence in model behavior: while detection recall is generally high, the ability to correctly reject non-Latin pages varies. This instability is most pronounced in Gemma3 with image-only input. It corroborates our earlier observation regarding its weak visual grounding. Yet, this issue extends beyond a single model, and we analyze further in Section 7.3.

7.3 Behavior on Non-Latin Pages

To further quantify the over-sensitivity observed in the main results, we analyzed non-Latin class recall (for page-level detection) and false positive token rates (for false Latin segment extraction) as two additional metrics, shown in Table 3. The analysis reveals that most models tend to over-detect Latin, particularly the smaller ones. The statistical baseline Lingua performs poorly, misclassifying over 70% of these pages, which would be a significant issue for large-scale processing.

More critically, multimodal inputs can overwhelm smaller models: Qwen2.5-VL-7B suffers a catastrophic collapse, indicating that overlapped visual and textual signals cause the model to hallucinate Latin across half the page. Crucially, this is a failure of fusion rather than perception: when fed with only images, the same model recovers much of its rejection capability, confirming that long and noisy OCR text acts as a distractor that overrides visual grounding in smaller architectures.

In contrast, the most robust performance comes from Qwen2.5-32B with text-only input, likely due to a stronger sensitivity to linguistic context provided by model scale. However, a promising finding is that even on misclassified pages, the number of erroneously extracted Latin tokens by the LLMs is generally small, suggesting that simple downstream filtering could effectively mitigate this over-detection problem.

7.4 Performance by Category

Shown as Figure 2, we evaluate model performance across different functional text categories specified

Model Details				Page	Token
Model	**Variant**	**Size**	**Mode**	**NL. Recall ↑**	**FP Rate ↓**
Lingua	-	-	T	27.69	2.64
DeepSeek-R1	-	671B	T	77.69	0.43
Qwen3	-	4B	T	40.77	4.97
Qwen2.5	VL	32B	I+T	65.38	2.66
	VL	32B	I	73.08	2.09
	-	32B	T	**96.15**	**0.17**
	VL	7B	I+T	10.77	55.54
	VL	7B	I	85.38	2.97
	-	7B	T	66.15	10.07

Table 3: Analysis on non-Latin pages: **Non-Latin Recall** in pages and **False Positive Rate** showing averaged percentage of tokens falsely identified as Latin on truly non-Latin pages.

in Section 4.2. As the models are tasked with extraction rather than classification, we measure performance using token-level recall in categorized segments on each page by the same fuzzy matching process as stated in Section 5.2. There is a large disparity between category difficulty: the models yield almost perfect performance for longer text types like independent and bilingual categories while struggling with the typically shorter code-switching, dictionaries, tables and charts, sidenotes, and to a lesser degree with footnotes.

The consistent performance trend across all models, including the baseline, suggests a shared, statistically driven behavior that relies on Latin's vocabulary and paragraph patterns over deep functional semantic understanding. Under this view, vision's role is mainly to improve OCR accuracy, and the explicit thinking only helps invoke more basic knowledge of Latin. To probe this hypothesis, we designed a more demanding joint extraction and categorization task (see Appendix E.1 for the details). The resulting categorized token F1 scores were exceptionally low: 21.0% for DeepSeek-R1 and 14.6% for Qwen2.5-VL (I+T), which strongly support our hypothesis. We conclude that this weak functional understanding, compounded with known OCR challenges, is a key factor behind the low extraction recall in certain categories. This conclusion is also partly validated by our prompt engineering experiments in Section 7.5.

7.5 Impact of Prompt Variations

To assess the robustness of our findings, we first evaluated the performance stability of four representative models across six distinct prompt strategies, as summarized in Table 4. We observe a consistent insensitivity to instruction phrasing across diverse architectures, with minimal Token F1 variance ($\sigma < 1.1$) and no consistent gains, for both text-only and multimodal models. This statistical evidence suggests that extraction capabilities are largely anchored by the models' intrinsic domain knowledge rather than specific prompt engineering.

Model	**Variant**	**Size**	**Mode**	**Page F1**	**Tok. F1**
DeepSeek-R1	Distill	70B	T	$96.28_{\pm 1.06}$	$84.27_{\pm 0.20}$
Qwen3	-	14B	T	$96.47_{\pm 1.00}$	$84.42_{\pm 0.62}$
Qwen2.5	-	32B	T	$97.14_{\pm 0.75}$	$81.89_{\pm 1.06}$
Qwen2.5	VL	32B	I+T	$95.58_{\pm 1.46}$	$84.44_{\pm 0.55}$
	VL	32B	I	$95.72_{\pm 1.79}$	$79.51_{\pm 0.79}$

Table 4: Prompt robustness statistics: performance reported as $Mean_{\pm Std}$ across 6 prompt strategies.

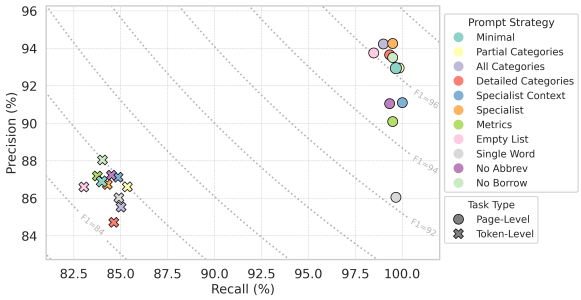


Figure 3: Impact of prompting on Qwen2.5-VL-32B.

Focusing on the detailed behavior of the top-performing Qwen2.5-VL-32B model with multimodal input (shown in Figure 3), most strategies result in only marginal changes, typically a trade-off between precision and recall, or different tasks, rather than a definitive improvement. We highlight the most salient strategies here, with full tables of all the prompts and results available in Appendix E.2. For instance, instructing the model to include "single-word" segments lowered page-level precision by encouraging over-extraction on short segments, while negative constraints like "no borrow" slightly increased token-level precision through a more conservative behavior by excluding the borrowed Latin words in other languages. Most revealingly, the "partial categories" strategy, directing the model to focus on its known weak areas, increased token recall but at the cost of a drop in precision. This indicates the prompt did not grant the model a deeper functional understanding; instead, it likely encouraged more speculative guessing in targeted contexts. This outcome reinforces our central hypothesis stated in Section 7.4 that the model's extraction is guided by statistical patterns rather than a semantic comprehension of the text's function.

Model & Input	Page F1	Tok. F1	Δ Tok. F1
Text-only Models			
DeepSeek-R1	96.65	72.26	-14.74
w/ Cleaned OCR	97.53	87.00	
Qwen3-14B	93.33	67.12	-17.66
w/ Cleaned OCR	96.61	84.78	
Qwen2.5-32B	96.02	67.48	-14.09
w/ Cleaned OCR	97.26	81.57	
Vision-Language Models			
Qwen2.5-VL-32B	95.38	79.24	-5.08
w/ Cleaned OCR	96.18	84.32	
InternVL3-38B	93.74	74.56	-9.68
w/ Cleaned OCR	95.62	84.24	
Image-only Models			
Qwen2.5-VL-32B (I)	**96.80**	**79.82**	-
InternVL3-38B (I)	89.51	56.53	-

Table 5: Ablation on OCR quality. Δ Tok. F1 denotes the token-level F1 change compared to the models with cleaned OCR text input.

7.6 Impact of OCR Quality

As detailed in Section 5.1, evaluating models on the ECCO corpus is complicated by significant discrepancies between legacy OCR and modern visual processing. To ensure meaningful assessment across all model types, we employ an OpenAI o1-based post-correction pipeline (Kanerva et al., 2025).

To quantify the necessity of this normalization, we conducted an ablation study using raw, uncorrected OCR transcripts, shown as Table 5. We observe that raw OCR creates a noise barrier that affects all models utilizing text input (including Vision-Language models fed with OCR text), with pure text models suffering the most severe degradation (> 14 point drop). Consequently, the performance degrades so severely that text-dependent models fall behind the best image-only model. This confirms that raw ECCO OCR constructs an excessive noise barrier that surpasses the detection task itself. Thus, our post-correction pipeline is not an o1-assisted "shortcut" for OCR-dependent models, but a necessary normalization to ensure fair cross-modal comparison.

7.7 Qualitative Evaluation and Error Analysis

Our qualitative evaluation reveals that LLMs' performance is primarily limited by two interacting factors: the inherent challenges of the ECCO data and the models' systematic misinterpretations of the task. These issues set practical limits on upper-bound scores. See Appendix E.4 for a detailed, example-oriented discussion.

First, data-centric challenges stem from the ECCO collection. Poor image quality and complex page layouts, such as multi-column text, marginalia, and varied fonts (see Figures 4 and 5 in Appendix A.2), result in noisy and fragmented OCR, even after post-correction. This degradation directly impacts model performance and hampers reliable annotation, especially for the brief Latin snippets found in challenging categories like dictionaries or footnotes, which are more susceptible to severe OCR errors (see Figure 7).

Second, we observe model-centric challenges. Models appear to rely on vocabulary and paragraph patterns over a deep functional understanding, leading to poor performance on short fragments and on words English loaned from Latin. This results in a consistent definitional mismatch with our annotation guidelines: models frequently misidentify Roman named entities, common anglicized Latin phrases (e.g., "e.g.", "etc.", etc.), and Latin-derived jargon as Latin, which significantly harms precision (see Figure 8). In many instances, this definitional disagreement accounts for the entirety of the prediction error, suggesting that model precision could be theoretically stronger with minor adjustments to the task definition in prompting, a behavior also empirically observed during our prompt tuning experiments, e.g., "no borrow" strategy (Section 7.5).

8 Conclusion

This paper introduced and benchmarked the novel task of zero-shot Latin discovery in historical documents. Our analysis demonstrates that this task is solvable with excellent performance of LLMs without task-specific fine-tuning. However, our key finding is that this success appears to stem not from deep functional semantic understanding, but from the models' ability to leverage more superficial statistical cues like vocabulary and text patterns, a behavior similar to traditional methods. This may limit the current models' approach to more nuanced historical tasks, suggesting that foundation models cannot replace human interpretative expertise.

A particular implication of our work is the high performance of image-only models (up to approximately 99% page-level F1), which enables an efficient two-stage approach for processing vast non-OCR archives. This capability may effectively unlock the "dark archives" of digitized heritage. Building on these, our future work will focus on applying our pipeline to the entire ECCO collection and extending the methodology to other corpora, such as the French BnF's collection.

Limitations

Our findings rely on a single corpus (18th-century British ECCO). While this aligns with our research scope, corpus-specific biases may limit generalizability, necessitating future validation on diverse collections, such as Romance-language texts. Additionally, due to the high cost of annotation, we lacked a separate validation set for hyperparameter tuning. Model selection relied on literature baselines rather than independent optimization.

Computational constraints limited us to single experimental runs with deterministic settings. While ensuring reproducibility, this approach does not capture the potential variance inherent in stochastic generation. Finally, this study benchmarks off-the-shelf models in a zero-shot setting. We acknowledge that task-specific fine-tuning would likely yield higher performance ceilings. Moreover, such experiments would be invaluable for identifying the most persistent challenges of the task that remain even after direct, task-specific training—perhaps related to deep functional semantic understanding of Latin—thereby providing crucial insights for future model development. Our newly created dataset provides the first dedicated resource to enable this line of inquiry.

Ethical Considerations

The underlying literary works from which our dataset is derived, sourced from 18th-century texts within the Eighteenth Century Collections Online (ECCO), are in the public domain. The compilation and sharing of our dataset, which comprises annotated excerpts and portions of page images from this collection, are conducted for research purposes under the permissions granted. We are committed to ensuring that the creation and dissemination of this dataset adhere to relevant copyright considerations and ethical guidelines.

We used ChatGPT and Gemini for grammar and spell-checking and stylistic polishing of the draft of this manuscript. All suggestions were critically reviewed and edited by the authors to ensure factual accuracy and originality.

Acknowledgments

We thank the Area Chair and the anonymous reviewers for their constructive comments and suggestions. This project has received funding from the European Union's Horizon Europe programme for research and innovation under MSCA Doctoral Networks 2022, Grant Agreement No. 101120349 and Grant Agreement No. 101119511. We also acknowledge CSC – IT Center for Science, Finland, for awarding this project access to the LUMI supercomputer, owned by the EuroHPC Joint Undertaking, hosted by CSC (Finland) and the LUMI consortium.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 technical report. *arXiv preprint arXiv:2303.08774*.

Gustavo Aguilar, Sudipta Kar, and Thamar Solorio. 2020. LinCE: A centralized benchmark for linguistic code-switching evaluation. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 1803–1813, Marseille, France. European Language Resources Association.

Samuel Backer and Louis Hyman. 2025. Bootstrapping AI: Interdisciplinary approaches to assessing OCR quality in english-language historical documents. In *Proceedings of the 5th International Conference on Natural Language Processing for Digital Humanities*, pages 251–256.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. 2025. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*.

Utsab Barman, Amitava Das, Joachim Wagner, and Jennifer Foster. 2014. Code mixing: A challenge for language identification in the language of social media. In *Proceedings of the First Workshop on Computational Approaches to Code Switching*, pages 13–23, Doha, Qatar. Association for Computational Linguistics.

Emanuela Boros, Maud Ehrmann, Matteo Romanello, Sven Najem-Meyer, and Frédéric Kaplan. 2024. Post-correction of historical text transcripts with large language models: An exploratory study. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, pages 133–159, St. Julians, Malta. Association for Computational Linguistics.

Patrick J Burns. 2023. Latincy: Synthetic trained pipelines for Latin NLP. *arXiv preprint arXiv:2305.04365*.

Patrick J Burns, James A Brofos, Kyle Li, Pramit Chaudhuri, and Joseph P Dexter. 2021. Profiling of intertextuality in Latin literature using word embeddings. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4900–4907.

OpenCompass Contributors. 2023. OpenCompass: A universal evaluation platform for foundation models. https://github.com/open-compass/opencompass.

Mitchell A Gordon, Kevin Duh, and Jared Kaplan. 2021. Data and parameter scaling laws for neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5915–5922.

Svetlana Gorovaia, Gleb Schmidt, and Ivan P. Yamshchikov. 2024. Sui generis: Large language models for authorship attribution and verification in Latin. In *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, pages 398–412, Miami, USA. Association for Computational Linguistics.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, et al. 2025. DeepSeek-R1 incentivizes reasoning in LLMs through reinforcement learning. *Nature*, 645(8081):633–638.

Gualberto A Guzmán, Joseph Ricard, Jacqueline Serigos, Barbara E Bullock, and Almeida Jacqueline Toribio. 2017. Metrics for modeling code-switching across corpora. In *Interspeech*, pages 67–71.

Jakob Hauser, Daniel Kondor, Jenny Reddish, Majid Benam, Enrico Cioni, Federica Villa, James Bennett, Daniel Hoyer, Pieter Francois, Peter Turchin, et al. 2024. Large language models' expert-level global history knowledge benchmark (HiST-LLM). *Advances in Neural Information Processing Systems*, 37:32336–32369.

Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, et al. 2024. OpenAI o1 system card. *arXiv preprint arXiv:2412.16720*.

Kyle P. Johnson, Patrick J. Burns, John Stewart, Todd Cook, Clément Besnier, and William J. B. Mattingly. 2021. The Classical Language Toolkit: An NLP framework for pre-modern languages. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations*, pages 20–29, Online. Association for Computational Linguistics.

Jenna Kanerva, Cassandra Ledins, Siiri Käpyaho, and Filip Ginter. 2025. OCR error post-correction with LLMs in historical documents: No free lunches. *arXiv preprint arXiv:2502.01205*.

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. 2020. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*.

Hanna-Mari Kristiina Kupari, Erik Henriksson, Veronika Laippala, and Jenna Kanerva. 2024. Improving Latin dependency parsing by combining treebanks and predictions. In *Proceedings of the 4th International Conference on Natural Language Processing for Digital Humanities*, pages 216–228, Miami, USA. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Chuwei Luo, Yufan Shen, Zhaoqing Zhu, Qi Zheng, Zhi Yu, and Cong Yao. 2024. LayoutLLM: Layout instruction tuning with large language models for document understanding. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 15630–15640.

Jani Marjanen, Tuuli Tahko, Leo Lahti, and Mikko Tolonen. 2025. Book printing in Latin and vernacular languages in northern Europe, 1500–1800. In *The Hermeneutics of Bibliographic Data and Cultural Metadata*, pages 27–66. National Library of Norway.

Valerio Perrone, Simon Hengchen, Marco Palma, Alessandro Vatri, Jim Q Smith, and Barbara McGillivray. 2021. Lexical semantic change for Ancient Greek and Latin. *Computational approaches to semantic change*, 6.

David Rosson, Eetu Mäkelä, Ville Vaara, Ananth Mahadevan, Yann Ryan, and Mikko Tolonen. 2023. Reception reader: Exploring text reuse in early modern British publications. *Journal of Open Humanities Data*, 9(1).

Sarah Schulz and Mareike Keller. 2016. Code-switching ubique est - language identification and part-of-speech tagging for historical mixed text. In *Proceedings of the 10th SIGHUM Workshop on Language Technology for Cultural Heritage, Social Sciences, and Humanities*, pages 43–51, Berlin, Germany. Association for Computational Linguistics.

Rachele Sprugnoli, Federica Iurescia, and Marco Passarotti. 2024. Overview of the EvaLatin 2024 evaluation campaign. In *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, pages 190–197, Torino, Italia. ELRA and ICCL.

Peter M. Stahl. 2021. Lingua: The most accurate natural language detection library for Python. https://

github.com/pemistahl/lingua-py. Python bindings for the Lingua language detection library.

Igor Sterner and Simone Teufel. 2023. TongueSwitcher: Fine-grained identification of German-English code-switching. In *Proceedings of the 6th Workshop on Computational Approaches to Linguistic Code-Switching*, pages 1–13, Singapore. Association for Computational Linguistics.

Milan Straka and Jana Straková. 2020. UDPipe at EvaLatin 2020: Contextualized embeddings and treebank embeddings. In *Proceedings of LT4HALA 2020 - 1st Workshop on Language Technologies for Historical and Ancient Languages*, pages 124–129, Marseille, France. European Language Resources Association (ELRA).

Elina Stüssi and Phillip Ströbel. 2024. Part-of-speech tagging of 16th-century Latin with GPT. In *Proceedings of the 8th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature (LaTeCH-CLfL 2024)*, pages 196–206, St. Julians, Malta. Association for Computational Linguistics.

Gemma Team, Aishwarya Kamath, Johan Ferret, Shreya Pathak, Nino Vieillard, Ramona Merhej, Sarah Perrin, Tatiana Matejovicova, Alexandre Ramé, Morgane Rivière, et al. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.

Qwen Team. 2025. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Rubèn Tito, Dimosthenis Karatzas, and Ernest Valveny. 2021. Document collection visual question answering. In *International Conference on Document Analysis and Recognition*, pages 778–792. Springer.

Maxim Tkachenko, Mikhail Malyuk, Andrey Holmanyuk, and Nikolai Liubimov. 2020-2025. Label Studio: Data labeling software. Open source software available from https://github.com/HumanSignal/label-studio.

Mikko Tolonen, Eetu Mäkelä, and Leo Lahti. 2022. The anatomy of Eighteenth Century Collections Online (ECCO). *Eighteenth-century studies*, 56(1):95–123.

Martin Volk, Dominic Philipp Fischer, Lukas Fischer, Patricia Scheurer, and Phillip Benjamin Ströbel. 2024. LLM-based machine translation and summarization for Latin. In *Proceedings of the Third Workshop on Language Technologies for Historical and Ancient Languages (LT4HALA) @ LREC-COLING-2024*, pages 122–128, Torino, Italia. ELRA and ICCL.

Martin Volk, Lukas Fischer, Patricia Scheurer, Bernard Silvan Schroffenegger, Raphael Schwitter, Phillip Ströbel, and Benjamin Suter. 2022. Nunc profana tractemus. detecting code-switching in a large corpus of 16th century letters. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2901–2908, Marseille, France. European Language Resources Association.

Yunting Xie, Matti La Mela, and Fredrik Tell. 2025. Multimodal LLM-assisted information extraction from historical documents: The case of Swedish patent cards (1945-1975) and ChatGPT. In *The 9th Digital Humanities in the Nordic and Baltic Countries Conference (DHNB 2025), March 5–7, 2025, Tartu, Estonia*, pages 1–15. University of Oslo Library.

Xiang Yue, Yuansheng Ni, Kai Zhang, Tianyu Zheng, Ruoqi Liu, Ge Zhang, Samuel Stevens, Dongfu Jiang, Weiming Ren, Yuxuan Sun, et al. 2024. MMMU: A massive multi-discipline multimodal understanding and reasoning benchmark for expert agi. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9556–9567.

Yuan Zhang, Jason Riesa, Daniel Gillick, Anton Bakalov, Jason Baldridge, and David Weiss. 2018. A fast, compact, accurate model for language identification of codemixed text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 328–337, Brussels, Belgium. Association for Computational Linguistics.

Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Yuchen Duan, Hao Tian, Weijie Su, Jie Shao, et al. 2025. InternVL3: Exploring advanced training and test-time recipes for open-source multimodal models. *arXiv preprint arXiv:2504.10479*.

A Segment Categories in Dataset

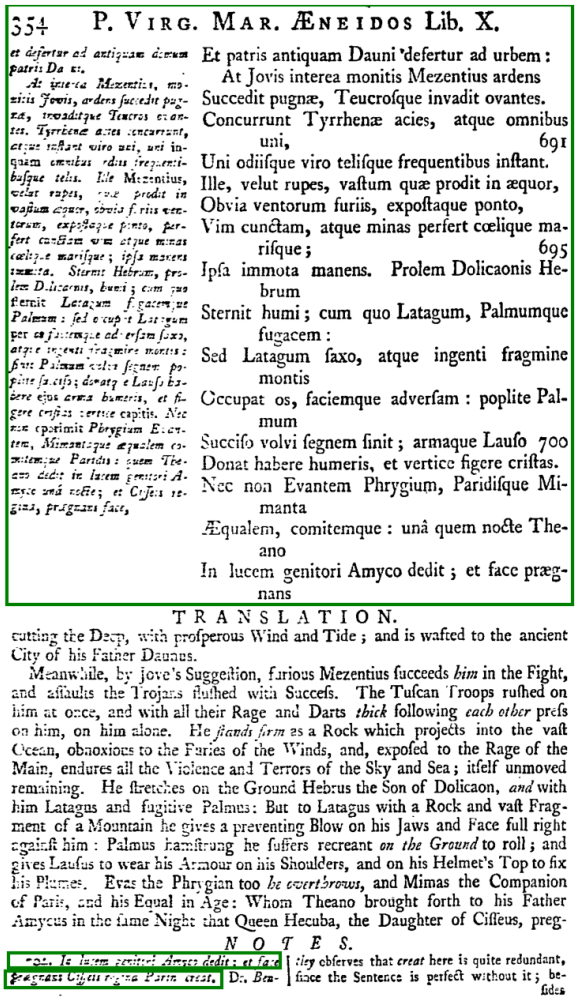
A.1 Categories of Latin Usage

We identified and annotated the following 12 categories of Latin usage found in the 18th-century documents:

1. Bilingual Editions (Latin/English): Original Latin text and its English translation right next to it.
2. Independent Latin Text: Original Latin text by the author, sometimes accompanied by English text on the same page.
3. Direct Quotations in Latin: Latin phrases or sentences that are quoted verbatim, often within an otherwise predominantly English text.
4. Code Switching: Text where the writer alternates between Latin and English within the same text, often for stylistic or rhetorical purposes.
5. Dictionaries: Latin text appears in a dictionary like context, for example, with entries that define individual Latin words, often with translations or explanations in another language.
6. Footnotes: Latin text appears in annotations or footnotes, often providing definitions or explanations for Latin words or phrases used in the main text.
7. Emblematic quotes: Latin phrases or sentences are used as symbolic or thematic elements, often serving as mottos, epigraphs, or maxims. Typically, set apart from the main text, such as at the beginning of chapters, sections, or works.
8. Sidenotes: Printed or authorial notes placed in the margins or alongside the main text.
9. Legal Formulae: Standardized Latin phrases used in legal contexts.
10. Ecclesiastical Formulae: Standardized Latin expressions used in religious or ecclesiastical contexts.
11. Tables and Charts: Use of Latin in tabular data, genealogies, calendars, scientific diagrams, inflection tables, or mathematical charts.
12. Indexes and Catalogs: Use of Latin in structured lists such as indices, bibliographies, or library catalogs.

A.2 Category Illustrations

Figures 4 and 5 show examples of Latin text categories. Both figures feature independent Latin text in the main text box at the top of the pages and footnotes at the bottom. Figure 4 is a bilingual edition of a Latin text, with an English translation directly below the Latin text at the top of the page. The footnotes in Figure 5 also include instances of code-switching and direct quotations of Latin text.



354 P. VIRG. MAR. ÆNEIDOS Lib. X.

et defertur ad antiquam Dauni patris Da ci.
At interea Mezentius, monitis Jovis, ardens succedit pugnæ, Teucrosque invadit antes. Tyrrhenæ acies concurrunt, atque instant viro uni, uni inquam omnibus odiis frequentibusque telis. Ille Mezentius, velut rupes, quæ prodit in vastum æquor, obvia furiis ventorum, expostaque ponto, perfert cunctam vim atque minas cœlique marisque; ipsa manens immota. Sternit Hebrum, prolem Dolicaonis, humi; cum quo Latagum, Palmumque fugacem: sed occupat Latagum per os faciemque adversam saxo, atque ingenti fragmine montis: sed Palmum volvi segnem, poplite succiso; donatque Lausus habere ejus arma humeris, et figere cristas vertice capitis. Nec non ejusmodi Phrygium Evantem, Mimantaque æqualem comitemque Paridis: quem Theano dedit in lucem genitori Amyco eâdem nocte; et Cisseis regina, prægnans face,

Et patris antiquam Dauni defertur ad urbem:
At Jovis interea monitis Mezentius ardens
Succedit pugnæ, Teucrosque invadit ovantes.
Concurrunt Tyrrhenæ acies, atque omnibus uni,
Uni odiisque viro telisque frequentibus instant.
Ille, velut rupes, vastum quæ prodit in æquor,
Obvia ventorum furiis, expostaque ponto,
Vim cunctam, atque minas perfert cœlique marisque;
Ipsa immota manens. Prolem Dolicaonis Hebrum
Sternit humi; cum quo Latagum, Palmumque fugacem:
Sed Latagum saxo, atque ingenti fragmine montis
Occupat os, faciemque adversam: poplite Palmum
Succiso volvi segnem finit; armaque Lauso
Donat habere humeris, et vertice figere cristas.
Nec non Evantem Phrygium, Paridisque Mimanta
Æqualem, comitemque: unâ quem nocte Theano
In lucem genitori Amyco dedit; et face prægnans

TRANSLATION.

cutting the Deep, with prosperous Wind and Tide; and is wafted to the ancient City of his Father Daunus.

Meanwhile, by Jove's Suggestion, furious Mezentius succeeds *him* in the Fight, and assaults the Trojans flushed with Success. The Tuscan Troops rushed on him at once, and with all their Rage and Darts *thick* following *each other* press on him, on him alone. He *stands firm* as a Rock which projects into the vast Ocean, obnoxious to the Furies of the Winds, and, exposed to the Rage of the Main, endures all the Violence and Terrors of the Sky and Sea; itself unmoved remaining. He stretches on the Ground Hebrus the Son of Dolicaon, *and* with him Latagus and fugitive Palmus: But to Latagus with a Rock and vast Fragment of a Mountain he gives a preventing Blow on his Jaws and Face full right against him: Palmus hamstrung he suffers recreant *on the Ground* to roll; and gives Lausus to wear his Armour on his Shoulders, and on his Helmet's Top to fix his Plumes. Evas the Phrygian too *he overthrows*, and Mimas the Companion of Paris, and his Equal in Age: Whom Theano brought forth to his Father Amycus in the same Night that Queen Hecuba, the Daughter of Cisseus, preg-

NOTES.

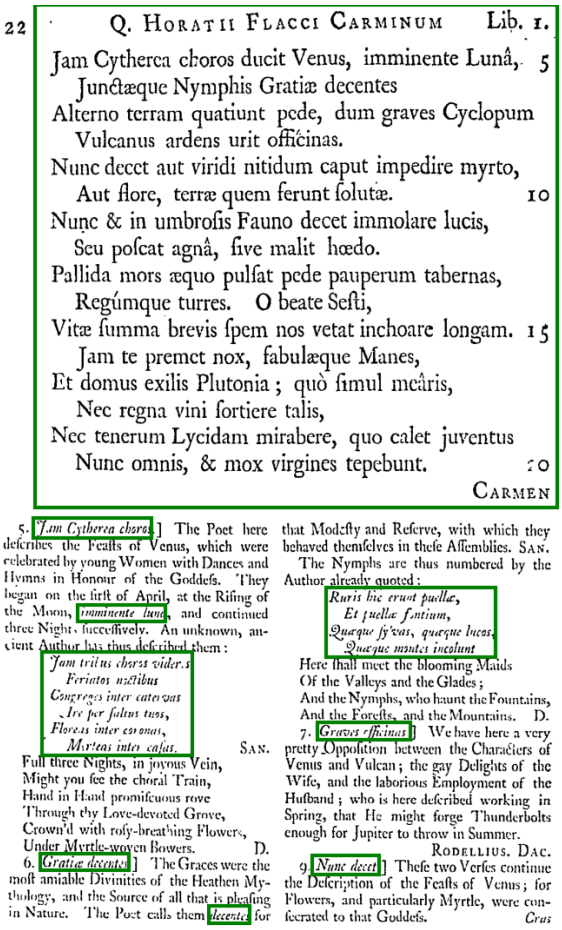
...te, in lucem genitori Amyco dedit; et face | *tley* observes that *erat* here is quite redundant,
prægnans Cisseis regina Parin erat. Dr. *Ben-* | since the Sentence is perfect without it; besides

Figure 4: An example page with Latin fragments.

B Evaluation Setting Details

B.1 Preprocessing

This section details the text preprocessing pipeline for evaluation, implemented in Python, to normalize both ground truth and predicted text strings before unigram token extraction. The primary goal of this pipeline is to standardize textual representations, thereby mitigating the impact of superficial variations (e.g., from OCR noise or stylistic differences) on downstream metrics calculation. Note,



22 Q. Horatii Flacci Carminum Lib. I.

Jam Cytherea choros ducit Venus, imminente Lunâ,
Junctæque Nymphis Gratiæ decentes
Alterno terram quatiunt pede, dum graves Cyclopum
Vulcanus ardens urit officinas.
Nunc decet aut viridi nitidum caput impedire myrto,
Aut flore, terræ quem ferunt solutæ.
Nunc & in umbrosis Fauno decet immolare lucis,
Seu poscat agnâ, sive malit hœdo.
Pallida mors æquo pulsat pede pauperum tabernas,
Regúmque turres. O beate Sesti,
Vitæ summa brevis spem nos vetat inchoare longam.
Jam te premet nox, fabulæque Manes,
Et domus exilis Plutonia; quò simul mearis,
Nec regna vini sortiere talis,
Nec tenerum Lycidam mirabere, quo calet juventus
Nunc omnis, & mox virgines tepebunt.

Carmen

5. *Jam Cytherea choros.*] The Poet here describes the Feasts of Venus, which were celebrated by young Women with Dances and Hymns in Honour of the Goddess. They began on the first of April, at the Rising of the Moon, *imminente Luna*, and continued three Nights successively. An unknown, ancient Author has thus described them:

Jam tribus choros videris
Feriatos noctibus
Congreges inter cateruas
Ire per salum tuos,
Floreas inter coronas,
Myrteas inter casas. San.

Full three Nights, in joyous Vein,
Might you see the choral Train,
Hand in Hand promiscuous rove
Through thy Love-devoted Grove,
Crown'd with rosy-breathing Flowers,
Under Myrtle-woven Bowers. D.

6. *Gratiæ decentes.*] The Graces were the most amiable Divinities of the Heathen Mythology, and the Source of all that is pleasing in Nature. The Poet calls them *decentes* for that Modesty and Reserve, with which they behaved themselves in these Assemblies. San.

The Nymphs are thus numbered by the Author already quoted:

Rurit hic errant puellæ,
Et puellæ fontium,
Quæque fylvas, quæque lucos,
Quæque montes incolunt

Here shall meet the blooming Maids
Of the Valleys and the Glades;
And the Nymphs, who haunt the Fountains,
And the Forests, and the Mountains. D.

7. *Graves officinas.*] We have here a very pretty Opposition between the Characters of Venus and Vulcan; the gay Delights of the Wife, and the laborious Employment of the Husband; who is here described working in Spring, that He might forge Thunderbolts enough for Jupiter to throw in Summer.

Rodellius. Dac.

9. *Nunc decet.*] These two Verses continue the Description of the Feasts of Venus; for Flowers, and particularly Myrtle, were consecrated to that Goddess. *Cras*

Figure 5: An example page with Latin fragments.

that this applies to the evaluation step only, while Latin extracting models take an input text without these steps.

For each text string, the following sequential operations are performed:

1. **Unicode Normalization:** Each string undergoes Unicode normalization using the `normalize` with "NFKD" method from Python's built-in `unicodedata` module. This step decomposes characters into their canonical forms, for example, separating accents from base characters, which helps in standardizing character representation.

2. **Ligature Replacement:** A predefined set of common ligatures is replaced with their constituent characters. Examples of replacements include 'ff' to `ff`, 'æ' to `ae`, and importantly for some historical contexts, '&' to `et`.

3. **Lowercasing:** All alphabetic characters in the string are converted to lowercase.

4. **Digit Removal:** All sequences of digits are removed from the string to avoid prediction ambiguity on digits, e.g., OCR digits in footnote notations.

5. **De-hyphenation (Word Merging):** This step addresses common OCR inconsistencies in handling end-of-line hyphens from historical documents. To ensure textual uniformity for subsequent analysis, word segments that were hyphenated, typically due to line breaks in the original source, are consistently merged into single tokens.

6. **Punctuation Stripping:** All standard punctuation marks, as defined by Python's `string.punctuation` constant, are removed from the string.

7. **Word Tokenization:** After the above cleaning steps, each processed sequence is tokenized into a list of individual words using the `word_tokenize` function from the NLTK library (`nltk.tokenize`, version 3.9.1).

B.2 Fuzzy Matching Algorithm in Token-level Metrics

To evaluate segment correspondence, we apply a fuzzy matching algorithm to compare lists of preprocessed ground truth tokens against predicted tokens for each sample. This approach calculates Precision, Recall, and F1 score while being robust to minor textual variations. The core matching logic is outlined in Algorithm 1.

The algorithm performs a greedy, one-to-one fuzzy match: each predicted token is compared against available ground truth tokens using a match indicator function (`IsFuzzyMatch`) based on edit distance and a predefined proportion threshold θ. A match only holds when the edit distance is less than or equal to θ proportion of the length of the ground truth token string. A ground truth token can only be matched once to ensure an accurate count of distinct true positive matches. This fuzzy approach is beneficial as it offers robustness to minor textual variations that may persist even after preprocessing, leading to a more meaningful evaluation of segment correspondence.

C Main Prompt Details

This section details the exact prompt templates employed to instruct the LLMs for the main experiment of Latin script detection and extraction. The

Algorithm 1 Fuzzy Matching and Token Metrics Output

procedure CALCULATEFUZZYMETRICS($GT\_Tokens, Pred\_Tokens, \theta$)
▷ Input: GT_Tokens, $Pred\_Tokens$ (lists of preprocessed tokens)
▷ θ (edit distance ratio threshold for a match)
▷ Output: Precision, Recall, F1 score
    $TP \leftarrow 0$
    $matched\_gt\_indices \leftarrow \emptyset$
    for each $pred\_token$ in $Pred\_Tokens$ **do**
        for each gt_token in GT_Tokens (with index gt_idx) **do**
            if $gt\_idx \in matched\_gt\_indices$ **then continue**
            end if
            if ISFUZZYMATCH($gt\_token, pred\_token, \theta$) **then**
                $TP \leftarrow TP + 1$
                Add gt_idx to $matched\_gt\_indices$
                break ▷ Current $pred\_token$ matched
            end if
        end for
    end for
    $FP \leftarrow$ length$(Pred_Tokens) - TP$
    $FN \leftarrow$ length$(GT_Tokens) - TP$
    $Precision \leftarrow TP/(TP + FP)$
    $Recall \leftarrow TP/(TP + FN)$
    $F1 \leftarrow 2 \times (Precision \times Recall)/(Precision + Recall)$
    return $Precision, Recall, F1$
end procedure

prompts were adapted based on the input modality being used. In the templates below, the placeholder `{page_text}` indicates where the OCR output corresponding to the processed page image was dynamically inserted.

Image + Text

```
Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference. Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

OCR Text: {page_text}
```

Image-only

```
Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image. Return the results as a list of strings in the JSON format: ["text1", "text2", ...].
```

Text-only

```
Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the OCR text of an image. Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

OCR Text: {page_text}
```

D Configuration of Baseline

For comparative language identification, we employed Lingua (version 2.1.0) (Stahl, 2021) as a baseline. The `LanguageDetector` was specifically configured to operate with a predefined restricted set of eight languages: English, French, German, Greek, Italian, Spanish, Portuguese, and Latin. This selection aims to encompass Latin itself and a set of the most frequently occurring languages within our target corpus ECCO (English, French, German, and Greek), while also including languages present in ECCO that share orthographic or lexical similarities with Latin (Italian, Spanish,

and Portuguese). Including these similar languages was intended to create a more global and robust test scenario for accurate Latin identification in ECCO.

In our pipeline, Lingua's function to detect multiple languages within a given text (`detect_multiple_languages_of` method) was utilized on each page's OCR output. From the resulting language segments identified by Lingua, only those substrings classified as Latin were subsequently extracted for our analysis and evaluation.

E Additional Results

E.1 Categorization Task Results

This section details the design and results of our complementary joint extraction and categorization experiment stated in Section 7.4, a more demanding task created to assess the models' deeper, functional understanding of text. The following subsections are structured to first outline the task's setup, including the specific prompt and the evaluation protocol. Following this methodological overview, we will present and analyze the detailed performance of the top models to highlight their capabilities and limitations on this challenge.

Task Design To move beyond simple Latin text retrieval and directly assess the models' ability to comprehend the functional role of Latin segments, we designed a more challenging joint extraction and categorization task. In this task, a model is required not only to identify and extract all Latin text from a given page but also to simultaneously assign each extracted segment to one of twelve predefined functional categories. The required output is a structured JSON object where keys correspond to the predefined category names and the values are lists of text segments assigned to each category. This task design compels the model to make explicit judgments about the semantic and contextual purpose of the text, thereby providing a clearer signal of its deeper comprehension abilities than a simple extraction task would require.

Task Prompt and Evaluation Methodology The core prompt for the joint task is slightly adapted based on the model's input modality (e.g., text-only, image-only, or multimodal). The version shown below is for the multimodal (I+T) setting:

> `Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image.`
>
> `After extracting all the Latin segments, assign each to one or some of the following categories:`
>
> - **`Bilingual`**`: Original Latin text with its English translation immediately following.`
> - **`Independent`**`: Standalone Latin text by the author, possibly with adjacent English.`
> - **`Direct Quote`**`: Latin text quoted verbatim within primarily English content.`
> - **`Code-switching`**`: Alternation between Latin and English within the same passage.`
> - **`Dictionary`**`: Latin entries in dictionary-style definitions or explanations.`
> - **`Footnote`**`: Latin in annotations or footnotes clarifying main text.`
> - **`Emblematic`**`: Latin used as mottos, epigraphs, or thematic standalone phrases.`
> - **`Side-note`**`: Marginal notes or annotations in Latin beside main text.`
> - **`Legal`**`: Standard Latin phrases used in legal contexts.`
> - **`Ecclesiastical`**`: Standard Latin phrases used in religious contexts.`
> - **`Tables and Charts`**`: Latin in tables, charts, genealogies, calendars, scientific or inflection data.`
> - **`Indices and Catalogs`**`: Latin in lists, indices, bibliographies, or catalog entries.`
>
> `Return a JSON object mapping each category to a list of Latin text segments, using exactly this format (no extra text or modifications): {"Bilingual": [...], "Independent": [...], "Direct Quote": [...], "Code-switching": [...], "Dictionary": [...], "Footnote": [...], "Emblematic": [...],`

```
"Side-note": [...], "Legal":
[...], "Ecclesiastical": [...],
"Tables and Charts": [...],
"Indices and Catalogs": [...]}

If a category has no results,
include it with an empty list.

OCR Text: {page_text}
```

Evaluation for this joint task is performed on a per-category basis. For each of the twelve categories, the list of text segments returned by the model under that category's key in the JSON output is treated as the predicted set. This set is then compared against the ground-truth list of segments annotated for that same category.

Precision, Recall, and F1 scores are then calculated at the token level for each category independently, using the same fuzzy matching process described in Section 5.2. The final Macro F1 score, reported in Table 6, is the unweighted arithmetic mean of these twelve individual F1 scores. This method allows us to assess not only the model's overall performance but also its specific strengths and weaknesses with respect to the understanding of each functional type of Latin text.

Results and Analysis Table 6 presents the per-category F1 (F), Precision (P), and Recall (R) scores for three top-performing models. As noted in our main text, the overall performance on this demanding task is terrible, with the best model, Deepseek-R1, achieving a Macro F1 score of only 20.98%.

Several key observations can be drawn from these results. Firstly, the text-only Deepseek-R1 significantly outperforms the multimodal Qwen-VL models, suggesting that the underlying language model's reasoning capability, rather than visual cues, is the current dominant factor for this specific categorization task. Secondly, there is a drastic variance in performance across categories. Notably, all models completely fail on the "Side-note" category, each scoring an F1 of 0.00, although it should have strong visual layout evidence. Performance is also exceptionally poor on categories requiring high contextual awareness, such as "Code-switching" and "Dictionary" entries. The relatively highest scores are achieved on categories with more distinct and self-contained structures, like "Direct Quote" and "Independent" sections, though even here the best F1 scores remain below 45%.

The overall low scores and high variance across categories underscore the challenge faced by current LLMs. Specifically, performance appears to be heavily influenced by the models' intrinsic biases, likely reflecting the training data distribution. The models exhibit partial capability on common, text-centric categories like "Direct Quote" but almost completely fail to classify rarer or more specialized, layout-dependent categories such as "Side-note". This area warrants significant further investigation as a dedicated future work. Focused research should first aim to diagnose the primary failure modes more precisely. For instance, future studies could seek to disentangle the core classification challenge from the requirement of generating structured joint output, and to determine whether poor performance stems from a fundamental lack of historical knowledge, poor text grounding in noisy document images, or suboptimal prompt design. Answering these foundational questions is a necessary next step before exploring more complex interventions like specialized training or novel model architectures.

E.2 Prompt Experiments

This section provides the full content of the prompt strategies used in our prompt engineering experiments, along with a table of their complete result numbers.

The following are the specific instructions given to the Qwen2.5-VL-32B model for each prompt strategy. The base prompt shown in Appendix C, used for the **Minimal** strategy, forms the core of most other prompts. For brevity, the final line of the prompt providing the OCR text input, `OCR Text: {page_text}`, is omitted from each example below as its format is consistent across all experiments.

Partial Categories

```
Identify and extract all segments
written in Latin (e.g., Classical
or Medieval Latin) from the
provided image, using the
accompanying OCR text as a
reference.

Return the results as a list
of strings in the JSON format:
["text1", "text2", ...].

Pay particular attention to
```

Model	*Bili*			*Code*			*Dict*			*Quote*			*Eccl*			*Embl*			*Foot*			*Indep*			*Index*			*Legal*			*Side*			*Table*			Macro F1
	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	F	P	R	
Deepseek-R1 (T)	18.82	19.81	18.48	3.16	4.50	3.07	6.51	7.58	6.45	43.01	46.44	43.00	27.39	28.29	28.11	16.28	16.47	16.16	30.11	36.03	28.99	44.71	47.13	44.21	25.38	29.26	24.02	23.84	26.99	23.25	0.00	0.00	0.00	12.49	17.84	10.86	20.98
Qwen2.5-VL-32B (I+T)	18.13	19.07	17.86	0.23	0.13	0.77	6.04	7.25	5.59	32.10	35.30	31.78	14.28	19.80	13.01	16.36	16.55	16.19	18.70	22.27	20.41	29.17	32.97	27.99	12.26	20.04	9.68	13.02	22.56	11.68	0.00	0.00	0.00	15.19	21.71	13.71	14.62
Qwen2.5-VL-32B (I)	16.15	17.77	15.78	0.00	0.00	0.00	3.56	4.72	2.86	25.81	30.13	25.23	9.61	21.13	8.48	7.00	7.04	6.97	13.24	16.47	13.68	28.60	31.14	28.26	2.28	5.83	1.45	8.38	18.76	7.16	0.00	0.00	0.00	3.88	6.90	2.70	9.88

Table 6: Categorization performance of the best models, where Macro F1 denotes the unweighted average of category-wise F1 scores. Category names in the header are abbreviated for brevity (e.g., *Bili.* for *Bilingual*).

identifying Latin segments in code-switching, dictionaries, footnotes, sidenotes, tables, and charts, while maintaining accuracy across all other categories.

All Categories

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Please pay attention to Latin in all of those categories: Bilingual, Independent, Direct Quote, Code-switching, Dictionary, Footnote, Emblematic, Side-note, Legal, Ecclesiastical, Tables and Charts, Indices and Catalogs.

Detailed Categories

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Please pay attention to Latin in all of those categories:

- Bilingual: Original Latin text with its English translation immediately following.

- Independent: Standalone Latin text by the author, possibly with adjacent English.

- Direct Quote: Latin text quoted verbatim within primarily English content.

- Code-switching: Alternation between Latin and English within the same passage.
- Dictionary: Latin entries in dictionary-style definitions or explanations.
- Footnote: Latin in annotations or footnotes clarifying main text.
- Emblematic: Latin used as mottos, epigraphs, or thematic standalone phrases.
- Side-note: Marginal notes or annotations in Latin beside main text.
- Legal: Standard Latin phrases used in legal contexts.
- Ecclesiastical: Standard Latin phrases used in religious contexts.
- Tables and Charts: Latin in tables, charts, genealogies, calendars, scientific or inflection data.
- Indices and Catalogs: Latin in lists, indices, bibliographies, or catalog entries.

Specialist Context

You are a specialist in classical languages and historical documents. You are given a scanned image of a page from an 18th-century document and its corresponding OCR result.

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Specialist

You are a specialist in classical languages and historical documents.

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Metrics

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Please ensure your extraction is both precise (no non-Latin segments are included) and comprehensive (all Latin segments are found).

Empty List

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Return an empty list if no Latin text is found.

Single Word

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Include segments even if they consist of only a single Latin word.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

No Abbrev

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Please do not include any abbreviations that are commonly used in contemporary languages, such as "etc.", "e.g.", "i.e.", "et al.", "a.m.", "p.m.", "A.D.", "B.C.", "P.S.", and similar.

No Borrow

Identify and extract all segments written in Latin (e.g., Classical or Medieval Latin) from the provided image, using the accompanying OCR text as a reference.

Return the results as a list of strings in the JSON format: ["text1", "text2", ...].

Please consider the language context and do not include Latin words or phrases that are used as loanwords or integrated into other languages, unless they function as Latin text in the context.

As summarized in the main text, most prompt variations result in a precision-recall trade-off. Table 7 provides the detailed metric numbers. Notably, adopting a "Specialist" persona yields the highest overall page-level performance (F1 96.81). Interestingly, adding more specific situational details to this persona ("Specialist Context") achieves perfect page-level recall (100.00%) but at the cost of significantly lower precision, resulting in a lower F1 score. This highlights the delicate balance in prompt design. Furthermore, simply providing more instructions on specialized knowledge, such as in the "All Categories" and "Detailed Categories" prompts, does not guarantee a significant improvement over the "Minimal" baseline, indicating that a detailed and informative prompt is not directly correlated with better performance. This reveals that a core deficit in contextual understanding remains the primary bottleneck.

Prompt Strategy	Page-Level			Token-Level		
	F	P	R	F	P	R
Minimal	96.18	92.94	99.66	84.32	86.90	83.99
Partial Categories	96.27	92.95	99.83	84.90	85.35	**86.61**
All Categories	96.55	94.23	98.99	84.20	85.53	85.01
Detailed Categories	96.41	93.65	99.33	83.63	84.72	84.63
Specialist Context	95.35	91.10	**100.00**	84.79	87.12	84.85
Specialist	**96.81**	**94.26**	99.49	84.47	86.74	84.28
Metrics	94.56	90.09	99.49	84.40	87.19	83.75
Empty List	96.06	93.75	98.48	83.84	86.60	83.03
Single Word	92.36	86.05	99.66	84.54	86.02	84.90
No Abbrev	95.01	91.05	99.33	84.85	87.23	84.50
No Borrow	96.41	93.51	99.49	**85.09**	**88.04**	84.02

Table 7: Impact of prompting on Qwen2.5-VL-32B.

For the token-level task, negative constraints like in the "No Borrow" and "No Abbrev" prompts yield the higher F1 scores (85.09 and 84.85 respectively), primarily by increasing precision. This indicates that providing a more precise, linguistically-grounded definition of the task is moderately effective. Ideally, however, a model should possess this specialist language identification capability intrinsically, distinguishing true Latin from common borrowings or abbreviations without requiring such explicit constraints. This requires moving beyond simple etymological recognition to a pragmatic understanding of a word's *function*, enabling the model to differentiate between genuine code-switching into Latin and fully assimilated loanwords (e.g., "status quo") or abbreviations (e.g., "et al."). The fact that explicit negative constraints are needed even to approximate this behavior highlights a key limitation of current models. It reveals that they still rely on manually encoded rules to navigate nuanced linguistic boundaries that a human expert would discern implicitly. Achieving this level of intrinsic, context-sensitive discernment remains a significant and challenging long-term goal for the development of truly knowledgeable AI agents as domain experts.

E.3 Fuzzy Matching Threshold

The fuzzy matching threshold, θ (representing the maximum allowed normalized edit distance relative to ground truth token length), was empirically set to 0.2 for all the experiments. This choice aligns with

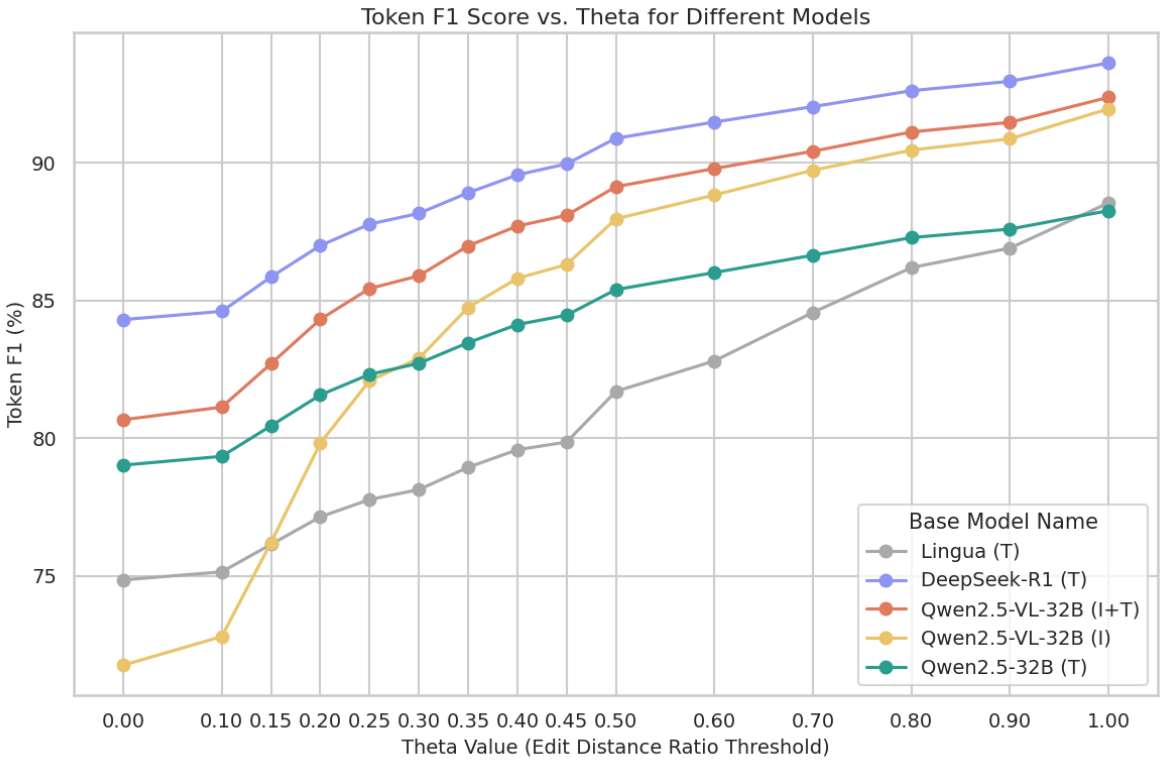


Figure 6: Token F1 scores on different θ value.

a common heuristic of tolerating approximately "1 error in 5 characters," suitable for OCR-derived text, and is supported by our sensitivity analysis in Figures 6. It consistently shows that while F1 scores generally increase with θ, the most substantial and steepest F1 score improvements for the majority of evaluated models are concentrated in the range leading up to $\theta \approx 0.2$, effectively compensating for common, fine-grained textual variations attributable to OCR noise. Although metrics may continue to rise beyond this point for some configurations on our dataset, we maintain $\theta = 0.2$ as a principled trade-off. A higher universal threshold could risk over-tolerating more substantial prediction errors beyond typical OCR noise, potentially prioritizing the matching of token quantity or approximate form over precise content fidelity. This could also obscure true output quality differences, especially when comparing models with varying input noise levels (e.g., image-only versus OCR-input systems).

E.4 Qualitative Results

More qualitative results are shown as examples to illustrate the best model's performance and the error modes.

Figure 7 shows an example of a mismatch caused by significant OCR noise caused by poor original image quality. Here, the post-corrected OCR of our ground truth differs so much from the OCR visual or MLLMs produced during the prediction process that not even our edit distance-based fuzzy ground truth matching can recover what is essentially a full match. This kind of error especially affects pages in the footnotes, code switching and dictionary categories, since the Latin texts in these categories tend to be printed in harder to detect fonts and layouts, which are additionally more likely to be affected by bad scan quality.

Figure 8 shows an example of a definitional misunderstanding of the predictions, which is the typical phenomenon also discussed in the prompt experiment in Section 7.5. Although there is no Latin text on the page, the prediction contains the Roman names appearing in the page text.

Figure 9 shows an example of a page where the prediction contains hallucinations. The model took part of the text and translated it into Latin in the prediction, without being prompted to do so.

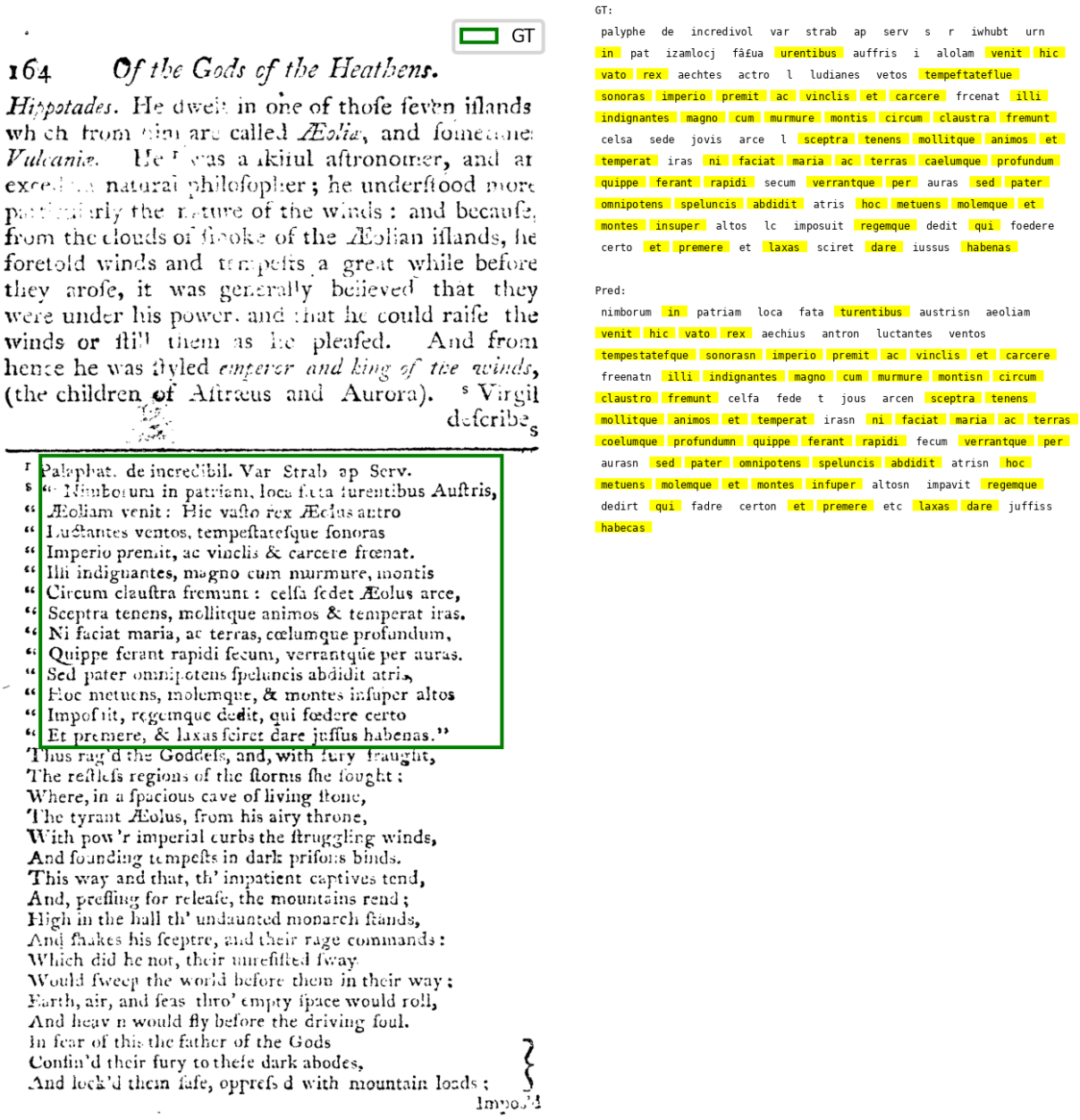
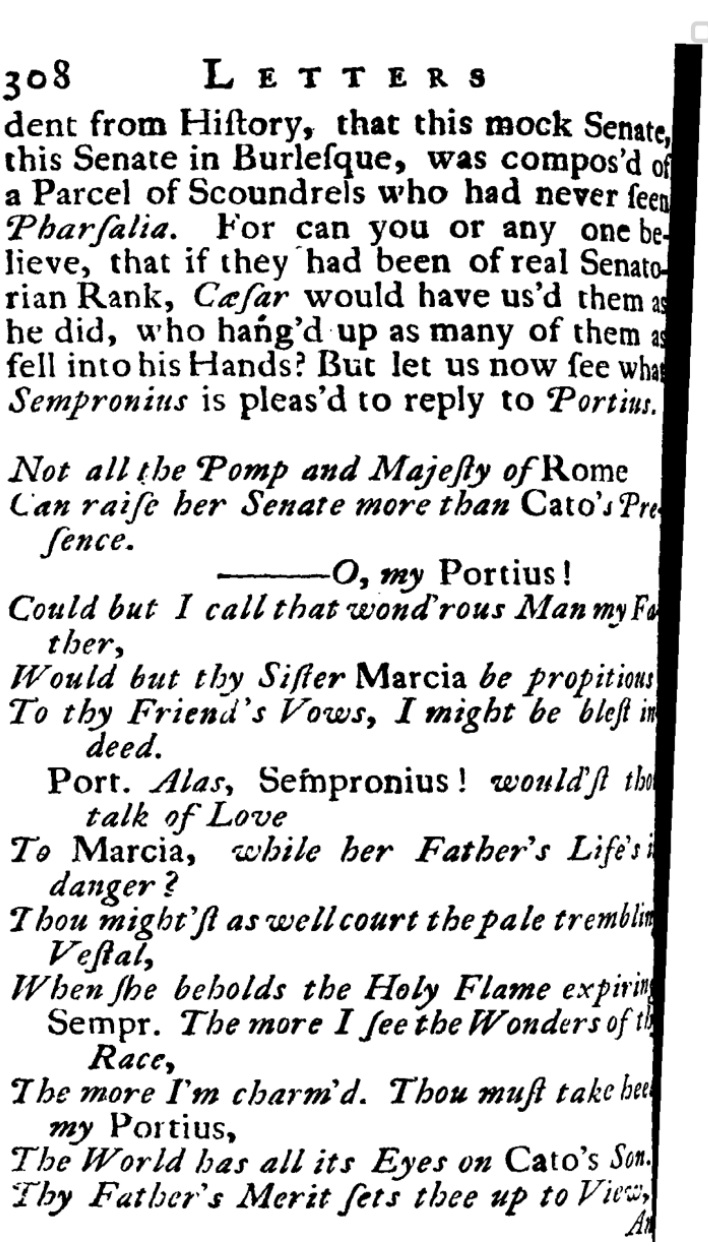


Figure 7: An example page with Latin fragments, together with our ground truth and prediction for that page.



308 LETTERS

dent from Hiſtory, that this mock Senate, this Senate in Burleſque, was compos'd of a Parcel of Scoundrels who had never ſeen *Pharſalia*. For can you or any one believe, that if they had been of real Senatorian Rank, *Cæſar* would have us'd them as he did, who hang'd up as many of them as fell into his Hands? But let us now ſee what *Sempronius* is pleas'd to reply to *Portius*.

Not all the Pomp and Majeſty of Rome
Can raiſe her Senate more than Cato's *Preſence.*
———*O, my* Portius!
Could but I call that wond'rous Man my Father,
Would but thy Siſter Marcia *be propitious*
To thy Friend's Vows, I might be bleſt indeed.
Port. *Alas,* Sempronius! *would'ſt thou talk of Love*
To Marcia, *while her Father's Life's in danger?*
Thou might'ſt as well court the pale trembling Veſtal,
When ſhe beholds the Holy Flame expiring
Sempr. *The more I ſee the Wonders of thy Race,*
The more I'm charm'd. Thou muſt take heed my Portius,
The World has all its Eyes on Cato's *Son.*
Thy Father's Merit ſets thee up to View,
An

GT:

Pred:
sempronius portius marcia pharsalia cato vesta

Figure 8: An example page without Latin fragments, together with our ground truth and prediction for that page.

" corruption, accuſtoming ourſelves by little and little, to " comprehend and bear the Majeſty of God." And S. *Cyprian*. " If he be made the Temple of God, I aſk, Of *Epiſt* 73. " what God? If he anſwer, Of the Creator, he could not " be His Temple, becauſe he did not believe in him. If " he ſay, Of Chriſt, neither can he be made His Temple, " becauſe he denies Chriſt to be God. Or if he ſay, Of " the Holy Ghoſt, yet ſince theſe Three are One, how can " the Holy Ghoſt be reconciled to that Man, who is an " Enemy either to the Father or to the Son?"

Very and Eternal God] The moſt notorious Oppoſer of the Godhead of the Holy Ghoſt, was *Macedonius*, Patriarch of *Conſtantinople*. The Hereſy itſelf is called the Hereſy of the *Pneumatomachi*, or *Fighters againſt the Spirit*; as denying the Divinity of the Holy Ghoſt, and aſſerting that he is only a created Energy or Power, attending upon and miniſtring unto the Son. In order to put a Stop to this Hereſy, the firſt Council of *Conſtantinople*, to theſe Words in the *Nicene* Creed, *I believe in the Holy Ghoſt*, added, *The Lord, and Giver of Life, who proceedeth from the Father and the Son, Who with the Father and the Son together is worſhipped and glorified, Who ſpake by the Prophets*. See *Pearſon* on the Creed, p. 325.

ARTICLE VI.

Of the Sufficiency of the Holy Scriptures for Salvation.

HOLY Scripture containeth all Things neceſſary to Salvation: So that whatſoever is not read therein, nor may be proved thereby, is not to be required of any Man, that it ſhould be believed as an Article of the Faith, or be thought requiſite or neceſſary to Salvation. In the Name of the holy Scripture we do underſtand thoſe Canonical Books of the Old and New Teſtament, of whoſe Authority was never any Doubt in the Church.

Of

GT:

Pred:

credo in spiritum sanctum dominum et vivificantem qui ex patre filioque procedit qui cum patre et filio adoratur et conglorificatur qui locutus est per prophetas

Figure 9: An example page without Latin fragments, together with our ground truth and hallucinated prediction for that page.