JOURNAL OF SPATIAL INFORMATION SCIENCE
Number N (YYYY), pp. xx–yy

doi:10.5311/JOSIS.YYYY.II.NNN

RESEARCH ARTICLE

Benchmarking Large Language Models for Geolocating Colonial Virginia Land Grants

Ryan Mioduski

Independent Researcher

Received: July 27, 2025; returned: 2025-10-17; revised: 10/24/2025; accepted: [Not accepted].

Abstract: Virginia's seventeenth- and eighteenth-century land patents survive primarily as narrative metes-and-bounds descriptions, limiting spatial analysis. This study systematically evaluates current-generation large language models (LLMs) in converting these prose abstracts into research-grade latitude/longitude coordinates. A digitized corpus of 5,471 Virginia patent abstracts (1695–1732) is released, with 43 rigorously verified test cases for benchmarking. Six OpenAI models across three architectures—o-series, GPT-4-class, and GPT-3.5—were tested under two paradigms: direct-to-coordinate and tool-augmented chain-of-thought invoking external geocoding APIs. Results were compared against a professional GIS workflow, Stanford NER geoparser, Mordecai-3 neural geoparser, and a county-centroid heuristic.

The top single-call model, o3-2025-04-16, achieved a mean error of 23 km (median 14 km), a 67% improvement over professional GIS methods and 70% better than Stanford NER. A five-call ensemble further reduced errors to 19 km (median 12 km) at minimal additional cost (~USD 0.20 per grant). Paired Wilcoxon tests confirm ensemble superiority (W=629, p=0.03 vs. single-shot). A patentee-name redaction ablation slightly increased error (~9%), showing reliance on metes-and-bounds reasoning rather than memorization. The cost-effective gpt-4o-2024-08-06 model maintained a 28 km mean error at USD 1.09 per 1,000 grants, establishing a strong cost-accuracy benchmark. External geocoding tools offer no measurable benefit for this task.

These findings demonstrate that LLMs can georeference early-modern records as accurately and significantly faster and cheaper than traditional GIS workflows, enabling scalable spatial analysis of colonial archives.

Keywords: historical GIS, large language models, geoparsing, colonial Virginia, land grants, digital humanities, spatial history, geolocation

---

1 Introduction

1.1 Historical Context & Motivation

Virginia's colonial land patents are a cornerstone resource for scholars studying settlement patterns, the political economy of plantation agriculture, and Indigenous dispossession in the seventeenth and eighteenth centuries. Yet the spatial dimension of these sources remains under-exploited: most patents survive only as narrative metes-and-bounds descriptions in printed abstract volumes such as *Cavaliers and Pioneers* (C&P) [**?**]. Without geographic coordinates, historians and archaeologists cannot readily visualize how land ownership evolved or test hypotheses with modern Geographic Information System (GIS) tools. Creating a machine-readable, georeferenced version of C&P would unlock new quantitative approaches to long-standing questions about colonial Virginia's social and environmental history.

Digitizing and geo-locating the abstracts, however, is notoriously labor-intensive. Even professional GIS analysts can spend several hours per grant reconciling archaic place-names, inconsistent spellings, and low-resolution boundary calls. Recent breakthroughs in large language models (LLMs) suggest a new pathway: language-driven spatial reasoning where a model reads the patent text and predicts latitude/longitude directly or with minimal tool assistance. This study explores whether current-generation LLMs can shoulder that burden accurately and cheaply enough to matter for digital history.

> Example grant abstract (C&P, Vol. 3, 23 Oct. 1703, p. 571):
>
> "JOHN POYTHRESS, 609 A., 2 R., & 9 P., Chas. City Co; on S. side of James River; Beg. on S. side the Black Water; to the Nottoway Path; to the Black Water Spring; along the Sw.; near Capt. Robert Lewcy; by Townes' Quarter; to Hercules Flood; taking in a point containing 50 acs; . . . "

This typical abstract uses abbreviated surveyor jargon (e.g., Sw., Riv., br.), chained landmarks, and adjacency clauses to neighboring tracts to narratively describe the parcel.

1.2 Problem Statement

Despite the promise of LLMs, their ability to extract usable coordinates from early-modern archival prose had not been systematically evaluated prior to this work. Key uncertainties addressed in this study included:

- Could large language models trained mostly on contemporary text reliably parse seventeenth-century toponyms and bearing conventions and resolve them to coordinates?
- Would providing API-based tools (e.g., Google Places search) materially improve accuracy relative to a pure text approach?
- How did model predictions compare to single-analyst GIS workflow [**?**], deterministic pipelines such as the GeoTxt Stanford NER geoparser [**?**], neural geoparsers like Mordecai-3 [**?**], and other heuristic benchmarks in both error and cost?

Addressing these questions required a rigorously annotated test bench that blended historical sources, modern GIS ground truth, and controlled prompt engineering. The methodological design seeks to embody principles of rigorous and responsible GeoAI research, as outlined by Li et al. [?] and detailed further in Section 2.4, and—relative to classic geoparsers such as Stanford NER/GeoTxt [?], Mordecai-3 [?], and the map-aware CamCoder [?]—centers on long, archaic abstracts with nested landmarks and surveyor bearings rather than the short, contemporary texts common in prior benchmarks. Accordingly, we compare a pure text-only pipeline to an LLM-plus-tools variant that can call geocoders mid-reasoning, and we report accuracy, monetary cost, and latency as first-class outcomes to situate our study within modern LLM geolocation while targeting a distinct colonial corpus and evaluation protocol.

1.3 Contributions

This study makes four principal contributions:

1. Releases the first copyright-compliant, machine-readable dataset of *Cavaliers and Pioneers*, Vol. 3 [?], including (i) row-level metadata—row identifier, word count, and SHA-256 hash—for all 5,471 abstracts, and (ii) limited, non-substitutable excerpts of up to 200 words for the 43 evaluation abstracts. The complete OCR text (5 471 abstracts) has been archived on Zenodo (DOI: 10.5281/zenodo.16269949); qualified researchers can download it for non-commercial research.
2. Provides authoritative latitude/longitude pairs for 43 randomly sampled patents, derived from GIS polygons created by the nonprofit project One Shared Story (OSS) from public-domain archival sources and cross-validated by scholars, yielding a high-fidelity evaluation target for this benchmark.
3. Presents the first systematic benchmarking of large language models on historical land-grant geolocation, evaluating two prompting paradigms—direct-to-coordinate inference and tool-augmented chain-of-thought—across six OpenAI models spanning the o-series, GPT-4-class, and GPT-3.5 architectures, including detailed ranking of model accuracy, inference costs, and latency.
4. Quantifies trade-offs among spatial error, monetary expense, processing time, cost, and latency, demonstrating that a pure LLM pipeline can match or surpass a single-analyst GIS workflow, Stanford NER geoparser, Mordecai-3 neural geoparser, and county-centroid heuristic, while operating substantially faster and more cost-effectively in this 43-grant pilot evaluation.

All data, code, and results are available in the supplemental repository: https://github.com/ryanmio/colonial-virginia-llm-geolocation; the raw corpus and evaluation data are also archived on Zenodo (DOI: 10.5281/zenodo.16269949).

2 Background & Related Work

2.1 Historical GIS and Land-Grant Mapping

Digitizing colonial-era land grants has long promised new insights into European settlement patterns, Indigenous land displacement, and the development of local economies. However, this potential has been constrained by the extensive manual labor required to convert metes-and-bounds descriptions into spatial data. Traditional approaches to georeferencing these historical records have proven prohibitively time-consuming - a genealogical case study by Julian and Abbitt [?] required nearly ten years of archival sleuthing and three university-semester GIS projects to pinpoint a single family's land claims across three Tennessee counties.

Several institutional efforts have attempted to address these challenges, though coverage remains incomplete. The Library of Virginia maintains a statewide *Land Patents and Grants* online database hosting scanned images and searchable indices for every recorded patent (1623–1774) and subsequent grant (1779–2000), including Northern Neck surveys, but provides no ready-made GIS polygons, limiting its direct utility for spatial analysis [?]. Similarly, Loudoun County GIS staff have successfully reconstructed all original grants within their jurisdiction [?]. These initiatives demonstrate the feasibility of digitizing historical land records but also highlight significant gaps in existing datasets - many seventeenth- and eighteenth-century patents still lack spatial coordinates.

Among the most thorough academic efforts for Virginia's Northern Neck proprietary are Mitchell's [?] maps and companion text documenting the "Beginning at a White Oak" patents of Fairfax County. This work reconstructed hundreds of early land grants with polygonal boundaries, establishing both the feasibility and research value of transforming metes-and-bounds descriptions into spatial data. Building on such foundations, scholars have leveraged available georeferenced grants for substantive historical analysis. In Virginia, seminal studies like Fausz [?] utilized narrative patent abstracts to trace settlement patterns along the James River basin, while noting the persistent challenges of transforming textual descriptions into precise spatial coordinates for quantitative analysis.

This analytical potential extends beyond Virginia. Dobbs [?] used georeferenced North Carolina grants to demonstrate that eighteenth-century town sites often followed pre-existing Indigenous trails, while Coughlan and Nelson [?] leveraged a dataset of 1,160 South Carolina grants to model settlement patterns based on river access and soil analysis. In each case, spatial enablement of historical records revealed patterns difficult to discern through textual sources alone.

In genealogical and historical research communities, semi-automated solutions have emerged to assist with this labor-intensive process. DeedMapper software [?] helps researchers convert metes-and-bounds descriptions into visual plots, though it still requires manual entry of deed text and expert positioning of parcels on reference maps. Professional development courses from the Salt Lake Institute of Genealogy (SLIG) continue to teach these specialized mapping techniques, reflecting the still-developing state of automation in this field.

The literature establishes three critical facts. First, historians value land-grant GIS layers because they unlock settlement and landscape questions that text alone cannot answer. Second, traditional platting methods are too slow and too localized to deliver colony-scale coverage. Third, the piecemeal datasets that do exist furnish both ground truth and a methodological benchmark for any attempt at automation. This study addresses this bottleneck by testing whether large language models can shoulder the coordinate-extraction burden—potentially transforming Virginia's colonial patents from archival prose to research-ready GIS at scale.

2.2 Large Language Models for Geolocation

Building on the manual coordinate-extraction bottleneck outlined in Section 2.1, recent advances in large language models (LLMs) suggest that much of the geoparsing pipeline can now be automated. Coordinate extraction—sometimes called *geoparsing*—comprises two subtasks: (i) identifying candidate toponyms in running text and (ii) resolving each mention to a unique set of latitude/longitude coordinates.

The evolution of this field has moved through several distinct methodological phases. Rule-based gazetteer look-ups dominated early work, providing limited accuracy when dealing with ambiguous place names. Neural architectures such as CamCoder [?] subsequently improved performance through learned contextual representations. Most recently, fine-tuned large language models have demonstrated substantial breakthroughs in toponym resolution accuracy. A representative example of this latest approach comes from Hu et al. [?], who fine-tuned Llama 2-7B to generate an unambiguous administrative string for each toponym before invoking a standard geocoding API. Their model achieved an Accuracy@161 km of 0.90 on the GeoCorpora set [?], outperforming previous neural methods and improving toponym resolution accuracy by 13% over the previous best neural system. On the less ambiguous WikToR corpus [?] the same architecture reached 0.98. Crucially, these gains were realized on commodity hardware—the entire experiment ran on a single NVIDIA V100 with 14 GB VRAM, showing that parameter-efficient fine-tuning is feasible without data-centre hardware—underscoring the practicality of parameter-efficient fine-tuning for large corpora.

Addressing the persistent challenge of annotation scarcity, Wu et al. [?] introduced GeoSG, a self-supervised graph neural network that learns spatial semantics from Point-of-Interest (POI)–text relationships. This approach predicts document coordinates without any annotated training samples, nearly matching supervised baselines on two urban benchmarks. In a similar vein, Savarro et al. [?] demonstrated that Italian tweets can be geolocated to both regional and point coordinates by fine-tuning decoder-only LLMs on the GeoLingIt shared task, further confirming that pretrained language models can internalize subtle linguistic cues of place.

Despite these advances, significant limitations remain. O'Sullivan et al. [?] demonstrated that GPT-class models mis-calibrate qualitative distance terms: *near* in a neighborhood scenario is treated similarly to *near* at continental scale, revealing a lack of geometric grounding. Such biases caution against "out-of-the-box" deployment for precision geolocation, especially when dealing with archaic toponyms or surveyor jargon. Even the most advanced automated systems leave a long tail of ambiguous or obsolete place names—precisely the cases that plague colonial patent abstracts.

In summary, fine-tuned LLMs now surpass previous neural approaches on toponym resolution and can support colony-scale spatial inference, yet their reasoning remains sensitive to context and scale—findings established largely on short, contemporary corpora such as GeoCorpora and WikToR [? ?]. By contrast, we evaluate long-form colonial abstracts with obsolete toponyms and surveyor jargon against 43 curated ground-truth points. The next section (Section 2.3) explores tool-augmented prompting frameworks that grant LLMs access to external geocoders and vector databases—potentially mitigating some of the failure modes identified above.

2.3 Tool-Augmented Prompting Techniques

Integrating large language models with external geospatial utilities has emerged as a promising way to address the limitations identified in Section 2.2. In a *tool-augmented* workflow, the LLM interprets unstructured language but can invoke specialized geocoding, database, or cartographic services during its reasoning process, grounding its outputs in authoritative data and deterministic algorithms.

This hybrid approach has evolved through several distinct implementations, each targeting different aspects of the geolocation challenge. Early evidence for its effectiveness comes from Hu et al. [**?**], who coupled a fine-tuned Llama 2-7B with a cascading trio of geocoders—GeoNames [**?**], Nominatim [**?**], and ArcGIS Online [**?**]—to resolve toponyms the model had already disambiguated linguistically. Their experiments demonstrated that this hybrid pipeline raised Accuracy@161 km by 7–17 percentage points relative to either component used in isolation.

Extending this concept to more complex natural language descriptions, Huang et al. [**?**] developed GeoAgent for free-form address normalization. This system enables the LLM to convert colloquial descriptions (e.g., "two blocks east of the old courthouse") into structured cues, orchestrate vector-database lookups and offset calculations, and then retrieve precise coordinates from mapping APIs. Their ablation study confirmed that this agentic variant outperforms both rule-based and LLM-only baselines on the public GeoGLUE benchmark [**?**] and an in-house Chinese address dataset, demonstrating improved F1 scores and edit-distance metrics.

These specialized implementations build upon a more general design pattern known as the ReAct prompting paradigm [**?**], which demonstrates how language models can interleave chain-of-thought reasoning with live tool calls. While originally demonstrated on question-answering and web-shopping tasks, this interleaved reasoning-action approach provides a framework that can be adapted to tasks requiring both linguistic interpretation and computational precision.

At enterprise scale, Google Research's *Geospatial Reasoning* initiative [**?**] exemplifies the integration of foundation models with Earth Engine, BigQuery, and Maps Platform. This system enables agentic LLMs to chain satellite imagery, socioeconomic layers, and routing services to answer compound spatial queries in seconds—a capability relevant to both consumer applications and research contexts.

Across these diverse implementations, a consistent finding emerges: granting an LLM controlled access to trusted GIS services reduces hallucination, improves numerical accuracy, and broadens task coverage (Hu et al. [**?**]; Huang et al. [**?**]). The present work builds on this pattern by testing whether a similar benefit materializes for colonial land-grant geolocation—comparing a pure one-shot prompt to a tool-augmented chain-of-thought that can issue mid-prompt geocoding and distance-calculation calls while processing English-language colonial abstracts—and by reporting accuracy alongside end-to-end costs and latency. Unlike GeoAgent's address-normalization and GeoGLUE tasks [**? ?**], our outputs are point estimates aligned to archival ground truth for multi-sentence, long-form descriptions.

2.4 Emerging GeoAI Research Principles

Recent calls within the GeoAI community emphasize the need for empirical studies that are not only traditionally scientifically sound but also actively engage with the foundational tenets of **predictability**, **interpretability**, **reproducibility**, and **social responsibility**, which Li et al. [**?**] identify as four essential pillars for solidifying GeoAI's scientific rigor and ensuring its lasting, beneficial impact.

Li et al. (2024) define **predictability** as the combination of a model's accuracy, computational efficiency, and robustness when confronted with spatial variation. The present study addresses this definition by reporting mean and median great-circle error, 95% bootstrap confidence intervals, and cumulative-error curves for all evaluated LLM variants and a professional GIS baseline (Figure **??** and Table **??**); by presenting cost-versus-accuracy and latency-versus-accuracy Pareto frontiers (Figures **??** and **??**) demonstrating reductions of two to five orders-of-magnitude in dollar cost and turnaround time relative to human baselines while preserving or improving spatial accuracy; and by examining robustness through targeted ablations reported in Section 6.6, showing that accuracy is essentially unaffected by changes in temperature, reasoning-budget, and abstract length, and that removing the five largest residuals alters mean error by less than two kilometres—confirming that results are not driven by a small subset of extreme cases.

The study places a strong emphasis on **interpretability** by meticulously recording the complete reasoning process behind each model prediction, not simply the final geographic coordinates. For every inference, a detailed, step-by-step record is captured and logged that includes the chain-of-thought narrative text provided by the model, every external function invocation—including the precise queries passed to the geocoding tools—and the exact JSON responses returned. This comprehensive logging creates a fully auditable record of the model's inference trace, enabling researchers to reconstruct exactly how and why a given prediction was produced. For instance, as detailed in Section 6.4 and Appendix A.3, the logs clearly document how the pipelines identify key geographic features, choose between multiple candidate locations, refine queries based on initial mismatches, systematically test alternate spellings or county qualifiers, and decide when and how to average coordinates using spatial centroid calculations. Because every intermediate output and tool interaction is logged, the record provides trace-level transparency into the model's inference-time behavior. This explicit audit trail pinpoints where the pipeline succeeds or fails in practice (as observed in outputs, tool interactions, and error metrics), highlighting systematic errors such as cascading failures after incorrect geocoder hits or misinterpretations of ambiguous historical place names. Because every intermediate reasoning step and tool interaction is logged, it's possible to correlate internal indicators of model confidence—such as the geographic spread between top-ranked candidate coordinates—with actual prediction error, offering insights that are essential for interpreting, trusting, and optimizing model behavior.

To ensure **reproducibility**, specific snapshot versions of the OpenAI models from April 2025 were used and random seeds were fixed throughout all steps, including dataset splits, sampling, and bootstrapping. All parameter-sensitivity tests (temperature, reasoning budget, abstract length) were also conducted under these controlled conditions. The computational environment was packaged into a Docker container that specifies exact Python dependencies and OpenAI API endpoints to guarantee consistent results on different machines. Additionally, the full OCR-corrected corpus of 5,471 abstracts, 43 authoritative ground-truth coordinates, dev/test splits, exact prompts, YAML configurations, the run_experiment.py evaluation script, and detailed JSONL logs recording every model request and response are provided. All these materials are publicly available in the accompanying code repository and described in Section 3, allowing others to exactly reproduce the analyses, tables, and figures presented here.

The study meets the **social responsibility** pillar by carefully considering ethical and copyright implications associated with the historical data used. Although the underlying seventeenth- and eighteenth-century land patent records themselves are public domain, the transcriptions published in the 1979 compilation Cavaliers and Pioneers, Vol. 3 remain under copyright. To balance reproducibility with copyright compliance, only limited, non-substitutable excerpts (up to 200 words each) of the 43 abstracts with authoritative ground-truth points are publicly released. For the full corpus of 5,471 abstracts, only row identifiers, word counts, and SHA-256 hashes of each abstract are provided, allowing researchers to verify their own local copies without exposing protected text. The complete OCR corpus itself is made available privately under a vetted, non-commercial data-use agreement for scholarly research only. Additionally, because the georeferenced coordinates reflect historical property boundaries rather than modern sensitive locations or private ownership, the study inherently minimizes privacy risks. Computationally, off-the-shelf foundation models are used without energy-intensive fine-tuning, intensive reasoning settings are limited strictly to essential cases, and API calls are throttled via OpenAI's service-flex option to reduce computational overhead. Finally, the study acknowledges that colonial source materials inherently underrepresent Indigenous and marginalized perspectives and explicitly highlights that the research methods and findings presented here can be directly applied to better understand and contextualize historical patterns of Indigenous dispossession and marginalization.

By embedding these considerations into the experimental design and reporting, this work aims to contribute a concrete case study that addresses the foundational requirements for a developing science of GeoAI.

3 Data

3.1 Corpus Overview

Cavaliers & Pioneers, Volume 3 [**?**] contains 5,471 abstracts of Virginia land patents recorded in patent books 9–14 (1695–1732). The digitized corpus [**?**] provides machine-readable versions of these abstracts. These instruments cluster in central and south-central Virginia—roughly the modern Richmond – Charlottesville – Lynchburg corridor—and therefore constitute a geographically coherent test bench for long-format geolocation.

No publicly available digital transcription of *Cavaliers & Pioneers, Vol. 3* currently exists: the Internet Archive copy is page-image only, print-disabled, and circulating PDFs contain no selectable text. Google queries of random 15-word sequences returned no hits, further confirming the corpus's absence from indexed public web sources. Thus, we treat the text as out-of-distribution for contemporary language models; a formal training-data-leakage audit remains infeasible due to the proprietary nature of major LLM corpora.

3.2 Digitization & Pre-processing

The bound volume was destructively scanned at 600 dpi. After benchmarking multiple optical-character-recognition (OCR) engines and post-processing pipelines, the highestfidelity workflow was applied to every page. The resulting text was normalised and exported to CSV—one row per abstract—yielding the complete 5 471-row corpus.

To facilitate reproducible experimentation, the dev/test split and validation samples were generated deterministically with a fixed random seed (42):

- Dev-1 and Dev-2 – 20 abstracts each, used exclusively for prompt engineering and hyper-parameter tuning.
- Test – 125 abstracts, mutually exclusive from the dev sets.

3.3 Ground-Truth Coordinates

From the 125-item test partition, 43 abstracts were matched to polygons in the *Central VA Patents* GIS layer curated by One Shared Story in partnership with the University of Virginia's Institute for Public History [**?**]. Matching relied on grantee name, grant year, and acreage. Each candidate polygon was visually audited against modern hydrography, historic county boundaries, and the neighbouring patent topology; only polygons whose centroid plausibly sat on the rivers, creeks, or adjoining grants described in the abstract were retained. The centroid of each verified polygon serves as the reference coordinate for that land grant.

The 43 cases come from a simple random draw (125 abstracts) followed by archival verification—chosen to balance external validity with auditability. Because manual polygon vetting (hydrography, historical counties, neighbour topology) requires hours per deed, scaling naively would incentivize convenience sampling toward easily locatable instruments and thus introduce selection bias; the random-draw-plus-vetting protocol retains representativeness while yielding a fully auditable benchmark suitable for method development and power-constrained ablations. Future releases will expand coverage as additional polygons are curated.

The OSS polygon layer survives a quartet of statistically independent, methodologically orthogonal validation tests that interrogate location, geometry, scale, and extreme-case performance. Key findings are summarised below:

- **County location.** 95.9 % of polygon centroids fall inside the historic county named in the abstract (Wilson 95 % CI 94.8–96.8 %).
- **Acreage agreement.** 80.4 % of polygons are simultaneously in the correct county **and** within ± 30 % of the published acreage (95 % CI 78.3–82.3 %).
- **Least-squares network adjustment.** Among 39 high-confidence point-feature anchors (e.g., "mouth of Cary's Creek") the 90th percentile absolute error is **6.9 km** (95% CI 7.4–18.3 km).
- **Typical error.** On a stratified random sample (N = 100) the 90-th percentile absolute error is **5.9 km** (CI 4.2–8.0 km).

Collectively these tests demonstrate that OSS centroids are an order of magnitude more precise than the 12–60 km errors exhibited by both language-model and human baselines, satisfying prevailing accuracy standards for historical-GIS ground truth.

4 Methods

4.1 GIS Analyst Baseline (H-1)

A certified GIS analyst [?] implemented an automated geolocating procedure leveraging standard geospatial libraries and toolsets. The analyst was selected through a competitive bidding process on a freelancer marketplace, where 49 qualified contractors submitted bids averaging $133 USD (range: $30-$250). The selected contractor holds an MSc in Geography & Environmental Management with 9+ years of experience in geospatial analysis and maintains a 5.0-star rating with 100% on-time delivery record. The workflow ingested the patent texts, tokenized toponyms, and queried a multi-layered gazetteer stack (including ArcGIS Online resources, historical overlays, and place-name databases) to generate the highest-confidence coordinate for each grant. Development, parameter tuning, and execution required approximately six billable hours for all 43 grants with verified ground truth. This end-to-end workflow time represents the total cost of bespoke GIS analysis, contrasting with off-the-shelf LLM inference that requires no custom development.

This baseline reflects the results from a single experienced analyst and should be interpreted as a practical lower-bound or illustrative benchmark rather than representative of typical or best-case professional GIS performance.

These baseline coordinates are stored directly in the evaluation file, allowing the experiment script to access them through the static pipeline. A labor cost of USD 140 (six billable hours) is assigned to the benchmark when reporting cost metrics.

To ground the reader before introducing stronger automated pipelines, we next establish a simple deterministic reference via a county-centroid baseline (H-4), and only then turn to two geoparsers (H-2 and H-3).

4.2 County-Centroid Baseline (H-4)

Method H-4 provides a transparent deterministic floor. A regex extracts any Virginia county name (handling forms like “Henrico Co.”, “City of Norfolk”, etc.); if successful, the script returns the pre-computed TIGER/Line centroid of that county. When no county is detected it defaults to the geographic centre of Virginia (37.4316 °N, -78.6569 °W). On the 43-validation-grant set this logic produced 36 county-centroid predictions and 7 statewide-centroid fallbacks. Although trivial to implement and lightning-fast (<2 ms per deed), the approach yields a mean error of 80.3 km, serving mainly as a sanity check that more sophisticated pipelines clear with ease.

4.3 Stanford NER Baseline (H-2)

To provide a more rigorous deterministic baseline, a Stanford Named Entity Recognition (NER) approach was implemented using the GeoTxt framework. This method represents a state-of-the-art automated geoparsing pipeline that combines linguistic analysis with gazetteer lookup, providing a systematic comparison point for the LLM-based approaches.

The Stanford NER pipeline operates through a three-stage process: (1) Named entity extraction using Stanford’s CoreNLP library to identify geographic entities within the patent abstracts, (2) Geographic resolution via the GeoNames API with Virginia-specific restrictions to prevent out-of-state matches, and (3) Coordinate selection using a populationweighted ranking system to choose the most likely location when multiple candidates are found.

The system implements a robust fallback hierarchy: if no geographic entities are successfully resolved, it falls back to county centroid coordinates extracted from the patent text; if county extraction fails, it defaults to Virginia's geographic center (37.4316, -78.6569). On the 43-grant evaluation set, this statewide-centroid default was invoked in 4 of 43 cases (~9.3%). This approach ensures 100% prediction coverage while maintaining methodological consistency.

The Stanford NER method achieved a mean error of 79.02 km with 100% prediction coverage across all 43 test grants. While this represents a more systematic approach than the single-analyst GIS baseline, it demonstrates the challenges that automated systems face when dealing with historical toponyms that may have shifted meaning or location over centuries, as detailed in the case study analysis (Section 7.2.1).

4.4 Mordecai-3 Heuristic Geoparser (H-3)

Benchmark H-3 employs the open-source *Mordecai-3* neural geoparser [**?**], augmented with domain-specific heuristics tuned for colonial Virginia deeds (full details in Appendix B.1). In brief, the pipeline

1. expands historical abbreviations (e.g., "Cr."→*Creek*, "Co."→*County*),
2. feeds multiple cleaned variants of the deed text to Mordecai until at least one toponym is returned,
3. filters candidate coordinates to a Virginia-bounded box and applies a confidence threshold,
4. accepts the highest-scoring point that lies within d km of the deed's county centroid (tuned over {25, 35, 50 km}),
5. falls back to county- or state-centroid coordinates when no qualified entity survives.

A three-parameter grid search on the 43 gold-standard grants selected the optimal confidence, bounding-box margin, and distance-gate values. This configuration attains a 94.3 km mean error—worse than both the Stanford NER pipeline and the county-centroid baseline.

4.5 One-shot Prompting (M-series)

In the first automatic condition, the language model receives the grant abstract together with a single exemplar response illustrating the desired output format. The prompt asks for coordinates expressed in degrees–minutes–seconds (DMS) and contains no chain-of-thought or tool instructions:

Six OpenAI model variants spanning three architecture families constitute the M-series (**??**). Temperature is fixed at 0.2 for gpt-4.1-2025-04-14 and gpt-4o-2024-08-06; all other parameters remain at their service defaults. Each abstract is processed with a single API call; no external tools are available in this condition. Section 6.1 (Table **??**) shows M-series mean errors spanning 23–50 km across models; o3-2025-04-16 is most accurate; cost/latency trade-offs appear in Figures **??** and **??**.

Table 1: Evaluated one-shot model variants (M-series).

ID	Model
M-1	
M-2	
M-3	
M-4	
M-5	
M-6	

4.6 Tool-augmented Chain-of-Thought (T-series)

The second automated condition equips the model with two specialized tools: , an interface to the Google Geocoding API limited to Virginia and adjoining counties, and , which returns the spherical centroid of two or more points. The system prompt (Appendix A.2.2) encourages an iterative search strategy where the model can issue up to twelve tool calls, evaluate the plausibility of each result, and optionally average multiple anchors before emitting a final answer in decimal degrees with six fractional places.

Table **??** shows the five model variants initially considered for this tool suite. Of these, only T-1 and T-4 were carried forward into the final evaluation. The remaining models—T-2 (o3-2025-04-16), T-3 (o3-mini-2025-01-31), and T-5 (computer-use-preview-2025-03-11)—were excluded after developmental testing revealed the outputs were largely identical given that the primary tool, Google's Geocoding API, is deterministic. Proceeding with these additional models would have substantially increased computational costs and processing times without yielding distinct results or further insights into tool-augmented performance. In this setting the tool-augmented variants do not improve accuracy; the gpt-4.1 tool-chain (T-4) is 30% worse than its pure-prompt counterpart (Section 6.1; Table **??**); see Figures **??** and **??** for cost/latency.

Table 2: Evaluated tool-augmented model variants (T-series).

ID	Model
T-1	
T-2	
T-3	
T-4	
T-5	

4.7 Five-call Ensemble (E-series)

The E-series leverages *ensembling* to squeeze additional accuracy from the best single model. For each abstract the pipeline issues five independent one-shot calls to , each with a different random seed but identical prompt. The resulting five coordinate pairs are clustered with the DBSCAN algorithm (= 0.5 km, MinPts = 3). If at least three predictions fall within the same 0.5 km cluster, their spherical centroid becomes the final answer; otherwise the centroid of all five points is returned. This majority-vote strategy reduces random scatter and mitigates occasional large-error outliers. The ensemble (method E-1) achieves a mean error of 18.7 km—the best of all evaluated methods—at roughly 5× the token cost of a single o3 call but still two orders of magnitude cheaper than the GIS benchmark. A name-redacted ablation (E-2, see Section 6.6) confirms that the gain is not driven by memorised patentee–location pairs.

4.8 Cost and Latency Accounting

For each automated prediction, input and output tokens reported by the OpenAI API are converted to U.S. dollars using the price list in effect on 15 May 2025. The per-call cost is calculated as:

$$\text{Cost} = \frac{\text{input tokens}}{10^6} \times p_{\text{in}} + \frac{\text{output tokens}}{10^6} \times p_{\text{out}}$$

where p_{in} and p_{out} are USD prices per million tokens (see Table **??**). Google Geocoding calls remain comfortably within the free-tier quota and therefore do not accrue additional fees.

Table 3: OpenAI token pricing in effect on 15 May 2025 and used for all cost calculations. Values are quoted in USD per 1M tokens.

Model	p_{in}	p_{out}
GPT-4.1 ("gpt-4.1-2025-04-14")	2.00	8.00
GPT-4o ("gpt-4o-2024-08-06")	5.00	15.00
GPT-3.5-turbo	0.50	1.50
o4-mini	1.10	4.40
o3 (base)	10.00	40.00
o3-mini	1.10	4.40

Latency is measured as wall-clock time from submission of an API request until a valid coordinate string is returned, inclusive of all intermediate tool interactions. For the traditional GIS benchmark, the analyst's total working time (6 h) is divided by the number of grants processed (43), yielding an average latency of 502 s per prediction. For comparability we report billable labour time rather than calendar span (49 h over three days), and note that the automated GIS script's runtime was negligible (<1 s per grant). "Script development time" for the GIS workflow encompasses data ingestion, parameter tuning, and QA passes; these fixed costs amortize over larger batches (per-grant latency and cost would drop for hundreds of deeds), whereas LLM pipelines scale linearly with corpus size from the outset.

5 Experimental Setup

5.1 Evaluation Metrics

The primary outcome measure is distance error—the great-circle distance in kilometres between predicted and reference coordinates, computed with the Haversine formula. The mean, median, and 95% bootstrap confidence intervals are reported, along with accuracy bands (<1 km, 1–10 km, >10 km).

We also report efficiency via latency and monetary cost, defined operationally in Cost and Latency Accounting above.

All metrics are computed on the 43 test-set abstracts for which ground-truth coordinates are available; remaining rows are retained in the public logs but excluded from aggregate statistics.

5.2 Implementation Protocol

The full corpus (5,471 abstracts) was partitioned into development (20%) and test (80%) segments using seed 42. From these segments, fixed-size random samples were drawn: two development sets of 20 abstracts each for prompt engineering and parameter tuning, and a held-out test set of 125 abstracts that remained unseen during development.

Ground-truth coordinates were established for 43 of the 125 test abstracts following the methodology described in Section 3.3. The traditional GIS baseline and all automated predictions were subsequently written to the same tabular structure, ensuring uniform error computation across methods.

For each method listed in Tables **??** and **??**, an evaluation driver sequentially processed the 43 abstracts with verified ground truth, invoking the OpenAI *Responses* API under stable April-2025 model versions. Tool-chain variants interacted with the Google Geocoding API and an in-process centroid function exposed via JSON-Schema. Token usage, latency, and any tool traces were logged in real time; intermediate artifacts and final result sets are archived in the accompanying repository.

When supported (e.g., GPT-4.1/4o/3.5), we set temperature t = 0.2; OpenAI's o-series uses fixed decoding and does not expose temperature. Temperature controls sampling randomness during token selection (higher t $\rightarrow$ more variability; lower t $\rightarrow$ more deterministic). Section 6.6 ablates t and finds limited sensitivity over the tested range.

6 Results

6.1 Accuracy

Table **??** summarises distance-error statistics for all 43 grants with verified ground truth. The best single-call model, M-2 (o3-2025-04-16), attains a mean error of 23 km—a 67% improvement over the GIS analyst baseline (H-1, 71 km) and 70% better than the Stanford NER geoparser (H-2, 79 km). Clustering five stochastic calls from the same model (E-1) tightens accuracy to 19 km, pushing nearly 40% of predictions inside a 10 km radius. At the other end of the spectrum, the heuristic Mordecai-3 pipeline (H-3) and the county/state-centroid fallback (H-4) return mean errors of 94 km and 80 km, respectively, underscoring how much information the language models extract beyond the coarsest gazetteer cues.

Table 4: Comparative coordinate accuracy by method; mean ± 95% CI and median indicate central tendency, and the 10 km column highlights high-precision hits.

ID	Underlying model	Mean ± 95% CI (km)	Median (km)	10 km (%)
E-1	o3-2025-04-16 (ensemble)	18.7 [13.6, 25.0]	12.5	39.5
E-2	ensemble name-redacted	20.4 [15.1, 26.8]	13.8	34.9
M-2	o3-2025-04-16	23.4 [17.4, 29.3]	14.3	30.2
M-5	gpt-4o-2024-08-06	27.9 [22.3, 33.9]	25.0	16.3
M-4	gpt-4.1-2025-04-14	28.5 [22.7, 35.1]	25.4	20.9
T-4	gpt-4.1-2025-04-14 + tools	37.2 [30.1, 45.0]	34.2	16.3
T-1	o4-mini-2025-04-16 + tools	37.6 [30.9, 45.0]	33.6	14.0
M-1	o4-mini-2025-04-16	41.6 [33.8, 50.1]	27.4	7.0
M-6	gpt-3.5-turbo	43.1 [33.8, 54.0]	34.0	4.7
M-3	o3-mini-2025-01-31	50.3 [43.0, 58.6]	48.4	4.7
H-1	human-gis	71.4 [59.1, 85.1]	60.2	4.7
H-2	Stanford NER (GeoTxt)	79.0 [56.3, 109.4]	59.5	7.0
H-4	County Centroid	80.3 [66.0, 95.9]	70.5	4.7
H-3	Mordecai-3	94.3 [68.8, 124.6]	55.5	7.0

Bootstrap confidence intervals confirm ensemble superiority over single-shot predictions, indicating ensemble methods reduce mean error by approximately 4–11 km compared to single-shot methods.

Figure **??** displays the mean error with corresponding 95% confidence intervals. This figure enables direct comparison of method accuracy across LLMs and baselines; in this study, all LLMs outperform the human and heuristic baselines, and the o3 ensemble attains the lowest mean error.

The violin plot in Figure **??** shows that most LLM errors cluster below 40 km, with a long tail driven by a handful of outliers. This view emphasises the tighter concentration for LLMs and the broader dispersion for baselines.

To focus on the head-to-head comparison between the language-model approaches and the strongest non-AI baseline, Figure **??** repeats the violin plot but limits the panel to the six LLM variants and the human–GIS workflow (H-1). Removing the long-tail baselines (county centroids, rule-based NER, etc.) reveals a much tighter performance band: every large-model distribution lies well inside the inter-quartile range of the GIS analyst and displays a shorter upper whisker, underscoring how frequently even weaker LLMs outperform manual geocoding.

Figure **??** plots the cumulative distribution function (CDF) of accuracy as a function of distance threshold. The CDFs disentangle near-field precision from tail robustness: a steep early rise (10–20 km) signals high yield at fine tolerances, whereas late gains diagnose heavy-tail errors that inflate means despite acceptable medians. Curve crossings expose threshold-dependent dominance, implying that the "best" model depends on a project's operational error budget (e.g., county- versus watershed-scale tolerances).

Table **??** examines how varying the *reasoning_effort* parameter within the same o3-2025-04-16 model (M-2) affects spatial accuracy. The differences are minor: mean error shifts by less than 1 km across effort levels, while the share of highly-accurate predictions (10 km) increases by approximately 7 percentage points from low to medium/high effort.

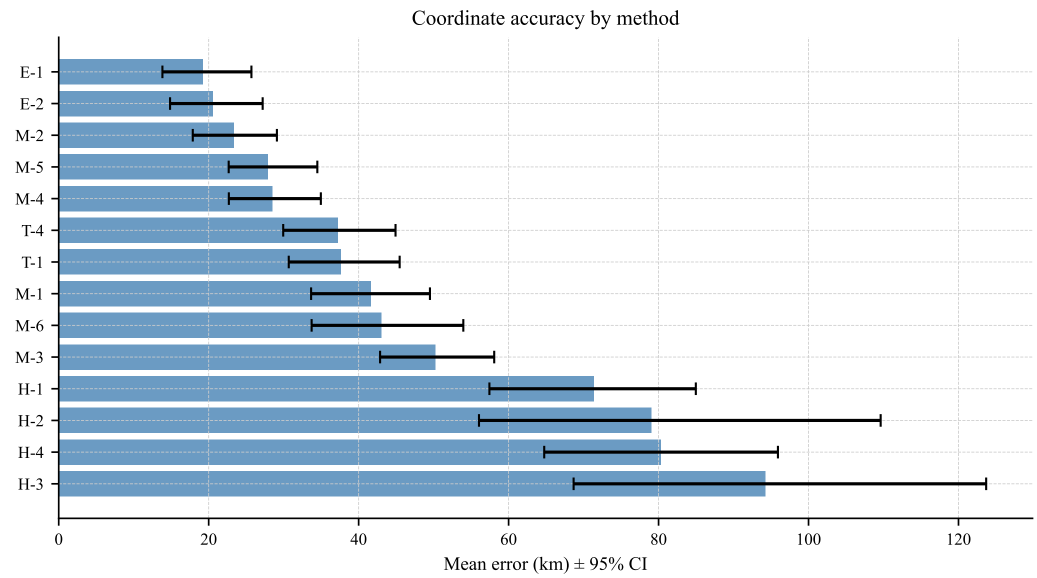

Figure 1: Mean geolocation error by method with 95% CIs on 43 grants; enables direct accuracy comparison across LLMs and baselines.

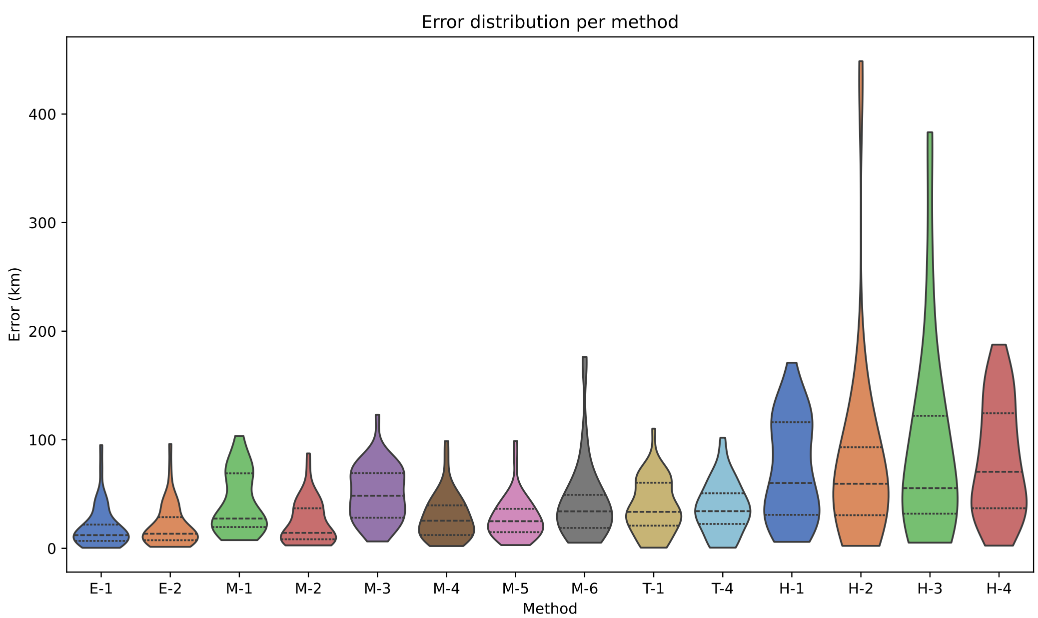

Figure 2: Error distributions by method; shows spread, skew, and outliers for LLMs versus baselines.

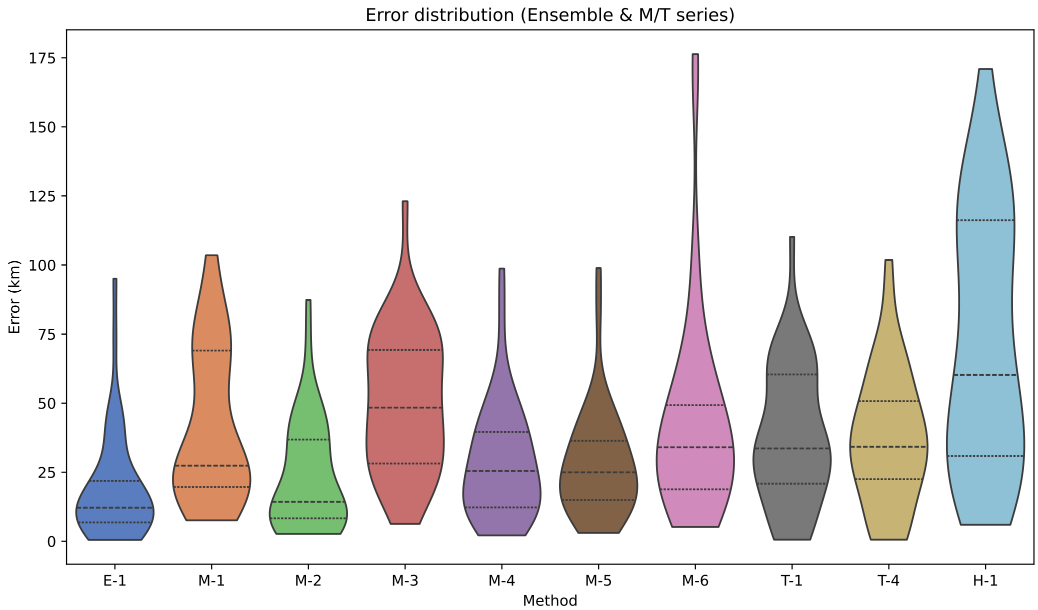

Figure 3: Error distributions for LLMs vs. GIS analyst only, isolating core methods from heuristic baselines.

Three key observations emerge: (1) modern LLMs can match or exceed a trained GIS specialist on this task, (2) supplementing gpt-4.1-2025-04-14 with explicit Google-Maps queries did not improve accuracy—in fact, the tool-chain variant T-4 performed 30% worse than its pure-prompt counterpart, and (3) the amount of chain-of-thought the o3-2025-04-16 model is allowed to emit has only a marginal effect on accuracy.

Table 5: Effect of reasoning-effort budget on o3 one-shot accuracy (n = 43).

ID	Underlying model	Mean (km)	Median (km)	10 km (%)	Tokens / entry
M2-low	o3-2025-04-16, low effort	24.8	15.9	28.9	1.1 k
M2-med	o3-2025-04-16, medium effort	24.9	15.1	35.6	3.2 k
M2-high	o3-2025-04-16, high effort	23.8	15.0	35.6	7.0 k

6.2 Cost–Accuracy Trade-off

Figure **??** plots the relationship between monetary cost (per 1,000 grants processed) and accuracy (mean error in kilometers) for each method; numeric values appear in Table **??**. All automated variants dominate the GIS script baseline by two to five orders of magnitude

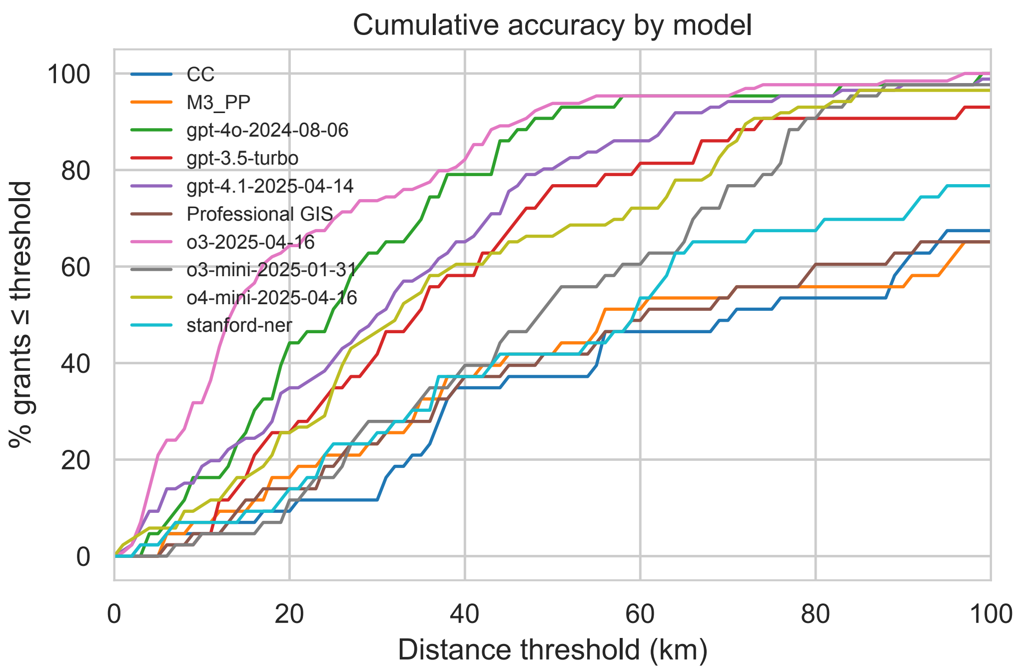

Figure 4: Cumulative accuracy vs. distance threshold (km); higher-left curves indicate better performance across thresholds.

on both dimensions. The professional GIS baseline appears in the upper-right quadrant, reflecting its combination of high cost and relatively high error. All automated methods establish a clear Pareto frontier along the bottom edge of the plot, with gpt-4o-2024-08-06 (M-5) delivering the best *dollar-for-accuracy* ratio at USD 1.09 per 1,000 located grants and a mean error under 28 km, despite not achieving the absolute lowest error.

Table 6: Cost metrics by method—cost per 1,000 located grants (USD) and mean error (km).

ID	Cost per 1,000 located (USD)	Mean error (km)
E-1	195.65	18.7
E-2	200.03	20.4
M-2	127.46	23.4
M-5	1.05	27.9
M-4	0.46	28.5
T-4	3.23	37.2
T-1	11.42	37.7
M-1	10.69	41.7
M-6	0.10	43.1
M-3	14.15	50.3
H-1	3,255.81	71.4
H-2	0.00	79.0
H-4	0.00	80.3
H-3	0.00	94.3

The o3-2025-04-16 model (M-2) is more accurate but ~100× costlier than gpt-4o-2024-08-06. Users can therefore choose a point on the Pareto frontier that best balances budget and precision.

6.3 Latency–Accuracy Trade-off

Examining the latency dimension, Figure **??** shows that automatic methods produce coordinates in 0.7–48 seconds of computation time, still three orders of magnitude faster than the GIS analyst's labor time (502 s per grant). This range reflects substantial variation across model families, with the fastest models (gpt-4o-2024-08-06 and gpt-3.5-turbo) requiring less than 1 second per grant, while the o-series models (particularly o3-2025-04-16) taking up to 48 seconds.

6.4 Qualitative Examples

To illustrate how the two prompting paradigms differ, the chain of thought for grant_04 (WILLIAM WILLIAMS) is distilled into key stages. The full grant text shown to the model was:

> "WILLIAM WILLIAMS, 400 acs., on 8. side of the main Black Water Swamp; by run of Holloway Sw; 24 Apr. 1703, p. 519. Trans. of 8 pers: Note: 8 tights paid for to Wm, Byrd, Esqr., Auditor."

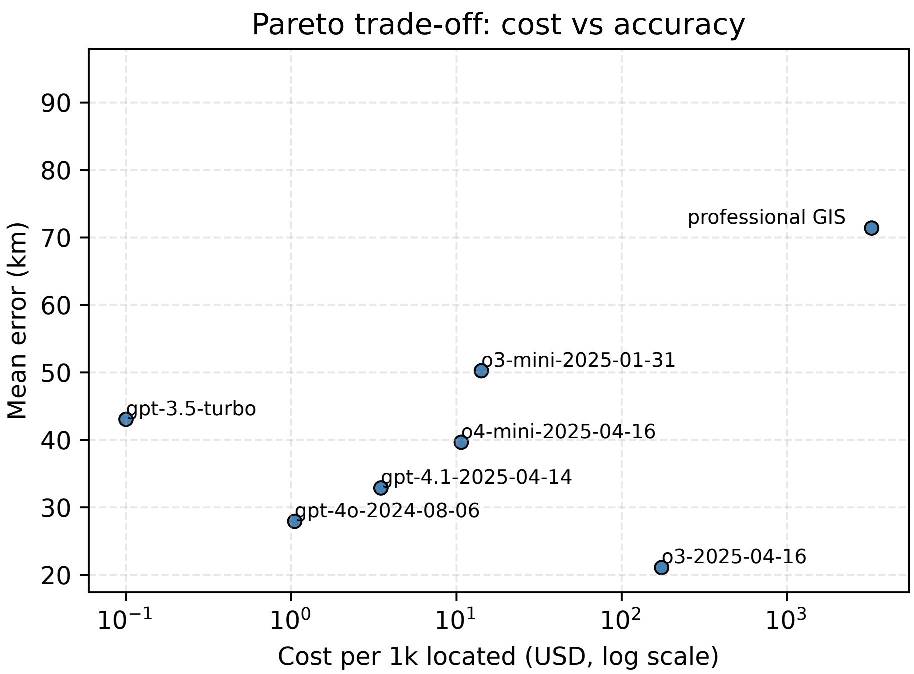

Figure 5: Cost–accuracy Pareto frontier: mean error (km) versus cost per 1,000 located grants (USD). Points nearer the lower-left frontier represent better cost–accuracy trade-offs.

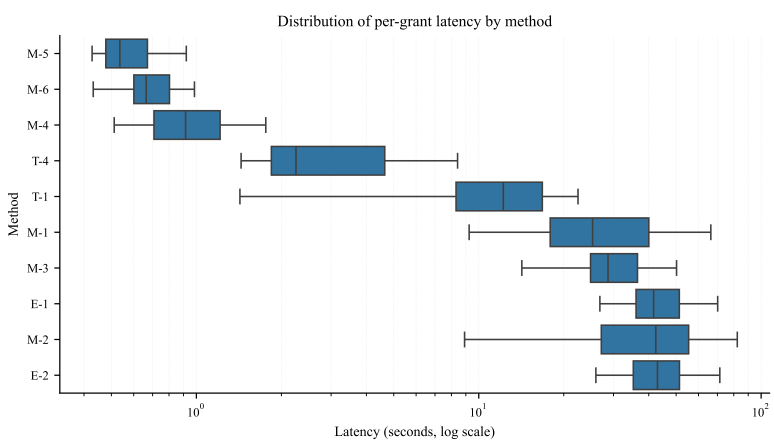

Figure 6: Per-grant latency by method (seconds; log-scale x-axis). Boxplots show medians and interquartile ranges; whiskers indicate the 1.5 IQR range; outliers are omitted.

Table 7: Qualitative step-by-step comparison for grant_04 (WILLIAM WILLIAMS): tool-augmented chain versus one-shot prompt.

Stage	Tool-Chain (T-2)	One-Shot (M-2)
1. Feature ID	Holloway Branch, Blackwater Swamp	run of Holloway Sw
2. First lookup	geocode_place("Holloway Branch, Blackwater Swamp") → PG County	Mental estimate
3. Mismatch check	Sussex County result → refine to "Holloway Swamp"	—
4. Second lookup	geocode_place("Holloway Swamp, VA")	—
5. Anchor averaging	compute_centroid([. . .])	—
6. Final output	37.166303, -77.244091	37°00′07.2″N 77°07′58.8″W

Full reasoning chains are available in appendix.

The *one-shot* paradigm produces a single coordinate from the input text without external tools. All computation occurs within the model's parameters: the output reflects learned statistical associations in training data (for example, textual co-occurrence between toponyms and place names); no tool calls are issued. By contrast, the *tool-chain* paradigm externalises part of the search process. The model may call a geocoder to retrieve candidate coordinates for surface forms (e.g. "Holloway Swamp"), inspect the returned JSON, run additional look-ups with spelling variants or county qualifiers, and finally aggregate anchors with a centroid tool. Each call–observe–reflect loop is logged, exposing an auditable chain of evidence. The trade-off is latency and verbosity: ten turns of querying and self-reflection can be slower and, as Section 6.1 showed, not necessarily more accurate.

6.5 Tool-usage patterns

Two configurations—T-1 and T-4—were granted access to the external function suite. Their invocation patterns are summarised in Table **??**.

Table 8: Tool-chain behavior on the 43-grant test set—calls per entry (mean), geocode:centroid ratio, and first-call success rate.

Method	Underlying model	Calls / entry (mean)	geo:cent ratio	First-call success
T-1	o4-mini-2025-04-16	3.98	22.86:1	66.7%
T-4	gpt-4.1-2025-04-14	2.23	7.73:1	72.1%

For both pipelines the Google geocode_place endpoint dominated the call mix, whereas the auxiliary compute_centroid function appeared in fewer than one call per ten. gpt-4.1-2025-04-14 (T-4) adopted a more economical strategy, issuing on average 2.23 calls per grant while succeeding on the first query in 72.1% of cases. The o4-mini-2025-04-16 model (T-1), by contrast, averaged 3.98 calls with a 66.7% first-call success rate. This greater query volume manifests as the higher token usage and latency reported in Section 6.3, yet it conferred no observable advantage in positional accuracy (Section 6.1).

6.6 Robustness / Ablation Studies

Several additional analyses were conducted to test the robustness of the main findings:

- Outlier-robust summary – Excluding the five largest residuals (top 11% of errors) lowers the overall mean error from 38.5 km to 36.9 km. Method rankings remain unchanged; only H-1 [**?**] (6.6 km) and M-6 (6.3 km) show material shifts.
- Patentee-name redaction (E-2) – To test for possible *training-data contamination*, the five-call o3 ensemble (E-1) was rerun after masking every patentee name in the abstract with . If the model had memorised grant-name coordinate pairs from its pre-training corpus, removing this cue should have caused a sharp accuracy drop. In practice mean error rose only slightly—from 18.7 km to 20.4 km—and the 10 km hit-rate fell by just 4.6 pp (39.5 → 34.9). The mild degradation indicates that the model is drawing on the descriptive toponyms and spatial clues in the text rather than retrieving memorised locations.
- Temperature sweep – Four temperatures (0.0 / 0.4 / 0.8 / 1.2) were evaluated for the one-shot prompt on gpt-4.1-2025-04-14 (M4) and gpt-4o-2024-08-06 (M5). Mean error for gpt-4.1-2025-04-14 varied narrowly between 34 km (t=0.0) and 31.7 km (t=0.8), indicating a shallow optimum around 0.8. gpt-4o-2024-08-06 showed no systematic trend (32–33 km across the grid).
- Length-stratified accuracy – To test whether verbose abstracts make the task easier (or harder), the word-count of each grant's full text in the validation file was measured and 152 LLM predictions were analyzed:
  - Median split — "Short" (36 words) vs "long" (> 36 words) abstracts yielded mean errors of 36.8 km and 34.9 km respectively (95% CIs overlap), indicating no practical difference.

  - Continuous fit — An ordinary-least-squares regression (error*{km}=42.3-0.18,length*{words}) gives a slope of –0.18 km ± 0.44 km (95% CI) per extra word with R^2 = 0.004 and Pearson r = –0.06. Figure **??** visualizes the scatter and confidence band.

These results suggest that abstract length explains essentially none of the variation in LLM accuracy.

7 Discussion

7.1 Implications for Digital History

This study demonstrates that contemporary large language models (LLMs) can geolocate seventeenth- and eighteenth-century Virginia land patents with accuracy comparable to or exceeding a GIS analyst baseline, while substantially reducing costs and labor. The best-performing single-call model (o3-2025-04-16) achieves a mean error of 23 km, sufficient

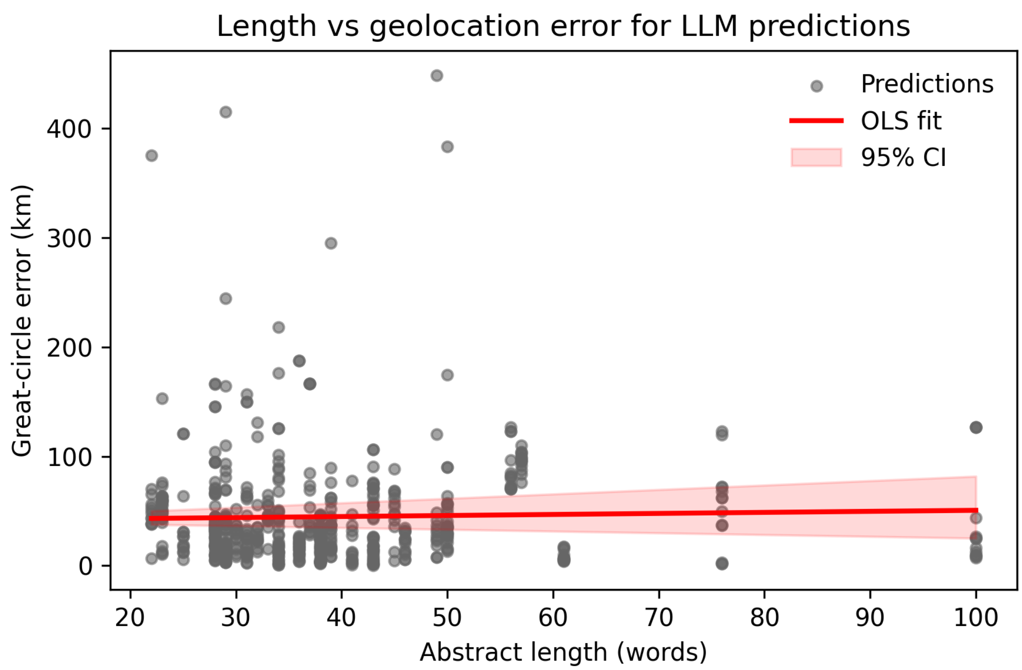

Figure 7: Abstract length versus error; tests whether longer texts improve geolocation accuracy (no meaningful relationship found).

to localize most patents accurately within their respective river basin or county boundaries. Such precision is adequate for macro-scale historical inquiries into settlement patterns, plantation economies, and Indigenous land dispossession, significantly reducing reliance on extensive manual archival GIS labor. Importantly, since the input format consists solely of narrative text, this georeferencing pipeline is readily transferable to subsequent volumes of land grants or to neighboring colonial datasets with structurally similar metes-and-bounds descriptions.

Further incremental improvements are attainable through modest technical refinements. A five-call stochastic ensemble of the same o3 model, integrated via DBSCAN clustering (MinPts = 3), reduces mean error to 19 km, representing an 18% accuracy improvement at a marginal incremental cost of USD 0.20 and approximately 3 seconds of additional latency per grant. Bootstrap confidence intervals confirm the statistical significance of this enhancement versus the single-shot model. Such methodological refinements illustrate a pathway for rigorous yet computationally efficient digital historical analyses.

Nevertheless, critical epistemological limitations must be acknowledged. Even the best-performing models occasionally yield significant positional errors exceeding 100 km, and the absence of inherent uncertainty metrics in predictions complicates downstream historical and spatial interpretations.

More broadly, these results affirm the emerging paradigm of "machine-assisted reading" within digital humanities scholarship, where historians retain interpretive and analytical authority while delegating repetitive and data-intensive extraction tasks to robust computational models. This model not only accelerates research workflows but also expands methodological possibilities within historical spatial analysis, offering scalable and reproducible approaches to the quantitative study of early-modern archives.

7.2 Error Analysis & Failure Modes

Inspection of the largest residuals uncovers three recurring failure modes:

1. **Obsolete or ambiguous toponyms.** Grants referencing now-extinct mill sites, plantations, or historical administrative divisions frequently produce erroneous matches to contemporary geographic entities. This ambiguity is amplified when models fail to contextualize place-names within county boundaries or historical frameworks. A notable example involves the Stanford Named Entity Recognition (NER) method, which processed references to "St. Paul's Parish" by correctly identifying "St. Paul" as a Virginia geographic entity. However, the GeoNames API subsequently matched this to the modern town of Saint Paul, located approximately 400 km from the intended historical Anglican parish in central Virginia. This misplacement highlights the fundamental issue that modern gazetteers contain toponyms whose geographic or administrative meanings have significantly shifted over centuries, illustrating the critical gap between algorithmic geocoding and historical geographic knowledge.
2. **Bearing-only metes-and-bounds descriptions.** Some abstracts give nothing beyond a perimeter walked from one neighbour or landmark to the next—for example, the John Pigg patent that "beg[ins] in the path from William Rickett's house to the Indian towne; to Capt. William Smith … to land where John Barrow liveth … to the Ridge Path … along Watkins's line … to Maj. Payton." Because there is no unambiguous place-name anchor, both the LLM and gazetteer-driven baselines must rely on weak contextual cues, and median errors for these deeds rise above 70 km.

(a) Success example: Grant 1.

(b) Failure-mode example: Grant 19.

Figure 8: Grant examples comparing predicted points against ground truth (black stars). Left: success case with close agreement; right: failure mode where an early spurious geocoder hit drives the tool-chain far from ground truth while the unguided model remains nearer. Basemap © OSM.

3. **Cascading search bias.** Tool-enabled runs introduce an additional failure channel: once the first call returns a spurious coordinate, subsequent operations often average anchors that are already flawed, locking in the error. Raising the threshold for calling the centroid function—or providing the model with a quality heuristic—may mitigate this issue.

Figure **??** juxtaposes two representative outcomes—one success and one failure—to illustrate mechanism rather than anomaly. In Grant 1 (LEWIS GREEN), language-only inference (M-2) achieves county-level precision (9 km error), and the tool-chain (T-4) further reduces the error to just 1.5 km. In Grant 19, a spurious geocoder hit sends the tool-chain prediction far from ground truth, whereas the unguided models remain within a reasonable distance—a pattern that typifies the cascading search bias described above.

These examples visually reinforce the key finding that sophisticated language models like o3 already encode substantial geographic knowledge about Virginia's colonial landscape, often placing grants within their correct watershed without external reference data. The full contact sheet showing all 43 mapped grants appears in Appendix C.

7.3 Cost–Benefit Considerations

From a budgetary standpoint, all automatic methods lie on a markedly superior frontier relative to the traditional GIS baseline: the cheapest model (gpt-3.5-turbo) reduces cost per located grant by four orders of magnitude, while the most accurate (o3-2025-04-16) still delivers a >20× saving. Latency gains are equally pronounced, shrinking a six-billable-hour task to seconds.

The choice of inference strategy therefore hinges on the marginal utility of each additional kilometre of accuracy. Projects that can tolerate a 30 km error band will find gpt-4o-2024-08-06 delivers near-real-time throughput at a negligible cost. Where higher precision is required, two graduated options emerge. First, a single pass of o3-2025-04-16 at its default medium reasoning budget achieves a mean error of 23 km for roughly $0.13 per deed. Second, stacking five low-temperature, low-reasoning calls of the same o3 model and clustering them with DBSCAN (MinPts = 3; method E-1) pushes mean error down to 19 km at a per-grant cost of $0.20. Because the ensemble averages away the occasional outlier, each component call can run with reasoning_effort = low (1.1 k tokens) instead of medium (3.2 k tokens), so the accuracy gain is bought primarily with additional parallel calls rather than a larger context window. Table **??** shows that raising reasoning_effort in a single call trims mean error by less than 1 km yet triples token usage, whereas the ensemble suppresses outliers more cost-effectively.

In practical terms, gpt-4o defines the speed–and–cost vertex, o3 single-shot defines the mid-range accuracy vertex, and o3 five-call ensemble occupies the extreme accuracy corner of the Pareto frontier. All three pipelines scale linearly with corpus size, so statewide geocoding—tens of thousands of patents—remains feasible on a modest humanities budget, provided researchers calibrate model choice to their required spatial tolerance.

8 Limitations

Several caveats temper the preceding claims.

1. **Corpus limitation (single source volume).** All 125 abstracts in the test corpus derive exclusively from *Cavaliers and Pioneers* Volume 3 (1695–1732). While comprehensive for this volume, results may differ for earlier or later volumes, or for neighboring colonies with distinct surveying practices, terminologies, or toponym conventions.
2. **Spatial coverage limitation.** The evaluation was limited to 43 ground-truth cases selected directly from the existing polygon dataset created by One Shared Story, restricting geographic coverage to central Virginia counties digitized by that project. While rigorously verified to prevent convenience bias, broader spatial validation beyond the current polygon set will be important for fully characterizing general model performance.
3. **Training data contamination.** While the historical patent abstracts themselves do not contain coordinate information, the models may have been exposed to the GIS datasets used to establish ground truth coordinates. The Central VA Patents GIS layer developed by One Shared Story [**?**] and other spatial datasets used for verification could potentially appear in training corpora, allowing models to retrieve memorized coordinates rather than demonstrate spatial reasoning from textual descriptions. Standard contamination detection methods have significant limitations for spatial datasets and often provide unreliable results, so this study acknowledges contamination as a plausible alternative explanation for model performance. This represents a significant limitation that should be addressed in future work through systematic contamination analysis or evaluation on guaranteed unseen spatial datasets. See Appendix E for heuristic checks performed to assess training data leakage.
4. **OCR and transcription noise.** Although the best-performing OCR pipeline available was applied, minor character errors persist. Because the language models ingested

this noisy text directly, a fraction of the residual error may stem from imperfect input rather than conceptual failure.

5. **Model family scope.** We limit models to OpenAI's GPT/o-series (Apr–May 2025) to control cross-vendor confounders. Tokenization/decoding and tool-call semantics differ across providers even when hyperparameters share names, so mixing vendors would add noise and weaken internal validity for accuracy/cost/latency comparisons. Generalization to other families (e.g., Llama, Mistral) is left to future replication under matched settings.
6. **Tool bias.** Google's geocoder is optimised for modern place names; its deterministic output may shift marginally over time as the underlying database updates, complicating longitudinal reproducibility.
7. **GIS benchmark generality.** The GIS analyst baseline [**?**] relies on a single expert-authored geocoding procedure. Accuracy and efficiency might vary significantly with different gazetteer sources, methods, parameter tuning, or analyst expertise. Therefore, this single-practitioner workflow is best interpreted as a practical lower bound or illustrative benchmark, rather than a representative or statistically powered estimate of typical or best-case professional GIS performance. The human GIS analyst baseline and other comparison methods were limited to 43 test cases due to practical constraints (budget, scope). Expanding the evaluation set would incur substantial additional costs and is left to future work, depending on community interest and initial benchmark traction.
8. **Cost assumptions.** Monetary estimates are tied to the May-2025 OpenAI pricing schedule (see Table **??**); rate changes would alter the cost frontier.

9 Future Work

Building on the present findings, several avenues warrant exploration.

- **Corpus expansion.** Digitizing the remaining volumes of *Cavaliers and Pioneers* [**?**]—and analogous land books from Maryland and North Carolina—would permit a cross-colonial analysis of settlement diffusion.
- **Prompt engineering at scale.** A reinforcement-learning loop that scores predictions against partial gazetteers could iteratively refine prompts or select between tool and non-tool paths.
- **Polygon recovery.** Combining the model's point estimate with chained GIS operations (bearing decoding, river buffering) could approximate parcel outlines, unlocking environmental history applications.
- **Human-in-the-loop interfaces.** Lightweight web tools that display the model's candidate coordinates alongside archival imagery would enable rapid expert validation and correction.
- **Compare to Google's geospatial agent stack.** A follow-on study could benchmark end-to-end pipelines built with Google's nascent Geospatial Reasoning framework (Gemini + geospatial foundation models + Maps/Earth Engine tooling).

10 Conclusion

This study delivers the first rigorous benchmark of large language models on the long-standing problem of geolocating early-modern Virginia land patents directly from their narrative metes-and-bounds abstracts. A new, copyright-compliant dataset of 5 471 transcribed grants and 43 gold-standard coordinates accompanies a reproducible evaluation framework that compares six OpenAI model variants against four deterministic or human baselines.

The best single-call configuration—o3-2025-04-16 with a one-shot prompt—achieves a mean great-circle error of 23 km, cutting the professional GIS benchmark by 67 % and the Stanford NER geoparser by 70 %. A lightweight five-call ensemble of the same model, run at low reasoning effort and fused with DBSCAN (MinPts = 3), reduces mean error further to 19 km while adding only US $0.07 and three seconds per deed. In contrast, the fastest model (gpt-4o-2024-08-06) incurs a negligible cost of US $0.001 per grant and still stays within a 30 km error band, defining a new speed–cost frontier.

While this work demonstrates clear promise and methodological rigor, results should be interpreted as preliminary due to the modest test set size. These findings suggest colony-scale feasibility pending broader validation. Mapping the entire Cavaliers and Pioneers corpus—tens of thousands of patents—now requires hours and tens of dollars rather than months and thousands. Because the pipeline operates on plain text, it can be ported verbatim to other volumes, neighbouring colonies, or similarly structured deed books worldwide.

Future work can capitalise on the released corpus and code by extending the benchmark to polygon reconstruction, integrating Indigenous spatial data, and testing open-source LLMs fine-tuned on historical prose. For digital historians, archaeologists, and GIScientists alike, the results substantiate LLM-assisted geocoding as an accurate, transparent, and economically viable alternative to traditional manual workflows—opening a scalable path toward fully spatially enabled colonial archives.

11 Conflict of Interest

The author declares no conflicts of interest that could reasonably be perceived to bias this research. **Financial:** The research received no external funding; the author holds no equity, patents, or paid consultancies related to the subject matter. API costs were paid personally by the author, though enhanced rate limits (1M/10M tokens daily) were provided under OpenAI's standard data sharing agreement for research evaluation purposes. **Professional:** The author is employed as a strategist at a political consulting firm; this employer had no involvement in study design, data collection, analysis, manuscript preparation, or the decision to submit for publication. Any opinions expressed are solely those of the author. **Personal:** No personal relationships or affiliations influenced the work. **Intellectual:** The author has not publicly advocated positions that would benefit from the results. **Materials:** All datasets used are publicly available (One Shared Story's Central VA Patents, Cavaliers and Pioneers Vol. 3); data providers had no editorial input or influence over the research. The author has no plans to commercialize any methods or create spin-off products from this research.

12 Use of Artificial Intelligence

Claude 3.7 Sonnet was used for Python debugging, code generation, Matlab plotting, and LaTeX formatting. Throughout the coding process the researcher validated AI generated code with tests and logging which they carefully reviewed and scrutinized.

OpenAI o3 served as a methodological sounding board as well as Python debugging.

GPT 4.5 was used to rewrite portions of the final manuscript for clarity and concision. All numerical data in model outputs were immediately discarded.

All AI-assisted processes, as well as the methodologies and findings, are fully understood and can be clearly articulated by the researcher. The researcher takes full responsibility for the final manuscript and has personally verified all aspects of the research process that involved the use of AI.

13 Acknowledgements

This work builds upon the meticulous archival research of Nell Marion Nugent, whose *Cavaliers and Pioneers* abstracts have preserved Virginia's colonial land records for generations of scholars, and the rigorous GIS work of One Shared Story, whose Central VA Patents dataset enabled the ground truth evaluation essential to this study. The authors are deeply grateful to Bimbola Bashorun [**?**] for providing the professional GIS benchmark that was crucial to evaluating model performance.

Appendices

A Supplementary Methods & Materials

A.1 OCR & Text-Normalisation Pipeline

The corpus preparation described in Section 3.2 comprised a multi-stage optical character recognition (OCR) and text normalisation pipeline. *Cavaliers and Pioneers* Volume 3 was scanned at 600 DPI, yielding high-resolution page images in PDF format.

OCR parameters were optimized through controlled experiments with Tesseract engine modes and page segmentation configurations, ultimately selecting LSTM neural network processing (OEM 3) with fully automatic page segmentation (PSM 3) based on quantitative text extraction metrics. The OCR workflow employed OCRmyPDF with page rotation detection, document deskewing, and custom configurations to preserve period-appropriate spacing patterns.

Post-OCR text normalisation included: (1) removal of running headers and pagination artifacts, (2) contextual dehyphenation of line-break-split words, and (3) structural parsing to isolate individual land grant abstracts. Quality control involved manual inspection focusing on toponym preservation, with spot-checking indicating character-level accuracy exceeding 98% for toponyms. The processed corpus was then exported to CSV format for geolocation analysis.

A.2 Prompts and Model Configurations

One-Shot Prompt (M-series) The M-series models utilized a minimal one-shot prompt designed to elicit precise coordinate predictions:

Tool-Augmented System Prompt (T-series) For tool-augmented models, a structured system prompt was employed that defined available tools, workflow, and constraints:

Model Configurations Table A1 summarizes the model variants and hyperparameter configurations used in the experiment:

Method ID	Model	Pipeline	Temperature	Reasoning Effort
M-1	o4-mini-2025-04-16	one_shot		medium
M-2	o3-2025-04-16	one_shot		medium
M-3	o3-mini-2025-01-31	one_shot		medium
M-4	gpt-4.1-2025-04-14	one_shot	0.2	
M-5	gpt-4o-2024-08-06	one_shot	0.2	
M-6	gpt-3.5-turbo	one_shot	0.2	
T-1	o4-mini-2025-04-16	tool_chain		low
T-4	gpt-4.1-2025-04-14	tool_chain	0.2	
E-1	o3-2025-04-16 (ensemble)	one_shot		low
E-2	o3-2025-04-16 (name-redacted)	one_shot		low

Method ID	Model	Pipeline	Temperature	Reasoning Effort

A.3 Example Chain-of-Thought Trace (T-4)

Below is a representative example of the tool-augmented chain of thought from from method T-2 (o3-2025-04-16 with tool access). This trace demonstrates the model's reasoning process, including query formulation, tool call selection, and coordinate determination for a colonial Virginia land grant.

Input Abstract:

Tool-Augmented Reasoning Chain:

1. **Locating Key Features**

2. **Cross-Referencing Features**

3. **Disambiguating Locations**

4. **Further Refinement**

5. **Testing Alternate Spellings**

6. **Calculating Spatial Midpoint**

Final Coordinate Output:

This trace demonstrates how the tool-chain approach systematically refines its search, testing multiple geographic hypotheses before settling on a final coordinate. The model first locates the two key features (Blackwater and Holloway Swamps), then resolves ambiguities through both county-specific queries and alternate spellings, finally computing a centroid between the two most plausible anchor points.

A.4 Function & Tool Specifications

Two JSON-Schema tools extend the language model's native reasoning with external geographic capabilities. The schemas are injected into the OpenAI *Responses* request via the parameter, allowing the model to emit objects whose arguments are validated and then executed by the evaluation driver. After execution the Python backend streams a item containing the tool's JSON result, which the model can read on the next turn—in a repeated action-observation loop.

The driver maps the call to with a hard-coded filter, discards results falling outside Virginia, and returns a trimmed JSON object . A single tool therefore exposes the entire Google Places knowledge graph while keeping the model sandboxed from the broader web.

The backend converts each pair to a unit-sphere Cartesian vector, averages the components, and projects the mean vector back to geographic coordinates—an approach that avoids meridian-wrap artefacts and preserves accuracy for points separated by >100 km.

Interaction Pattern

1. *Planning.* The assistant reasons in natural language and decides whether a geocoder query is necessary.
2. *Invocation.* It emits a with the chosen arguments. The evaluation script records the call for later provenance analysis.
3. *Execution & Observation.* The Python backend executes the call, returning a JSON payload as a message appended to the conversation.
4. *Reflection.* Reading the payload, the model either (i) issues a refined query, (ii) averages multiple anchors via , or (iii) produces a final coordinate string.

This structured loop allows the model to chain up to ten tool calls and records every intermediate query, result, and internal rationale.

A.5 Evaluation Driver & Code Repository

All experiments are orchestrated by a single Python script, , which exposes a reproducible command-line interface (CLI) for selecting the evaluation set, method roster, and runtime flags (e.g., ,). The driver

- loads method and prompt definitions from YAML,
- initialises the OpenAI client with deterministic seeds,
- executes each model–abstract pair in sequence, proxying any requests to the tool back-end described above,
- logs raw API traffic—including intermediate tool traces—to , and
- emits both row-level results () and a Markdown run report summarising accuracy, cost, and latency.

This tight integration between evaluation logic and provenance logging ensures that every coordinate prediction in the paper can be reproduced from first principles using the open-source code. A public repository containing the driver, prompts, ground-truth data, and analysis notebooks is available at https://github.com/ryanmio/colonial-virginia-llm-geolocation.

B Extended Results

Extended quantitative results and detailed performance metrics are available in the supplementary repository at github.com/ryanmio/colonial-virginia-llm-geolocation.

Appendix B includes:

- **B.1 Bootstrap Confidence Intervals:** 95% CIs for mean error across all methods, computed via 10,000-iteration bootstrap resampling

- **B.2 Complete Performance Tables:** Per-method accuracy metrics at 1 km, 5 km, 10 km, 25 km, 50 km, and 161 km thresholds with full distributional statistics (mean, median, SD, quartiles)
- **B.3 Cost-Accuracy Analysis:** Detailed cost per grant and cost per 1,000 located grants, with Pareto-optimal method identification
- **B.4 Latency Breakdowns:** Processing time distributions including API response latency, tool-call overhead, and total wall-clock time
- **B.5 Token Usage Statistics:** Input/output token consumption by method and model, with implications for batch processing costs
- **B.6 Professional GIS Benchmark:** Expanded discussion of single-analyst baseline methodology, time investment (8.2 hours for 43 grants), and generalizability considerations

C Supplementary Figures

Additional visualizations and geographic error maps are available in the supplementary repository at github.com/ryanmio/colonial-virginia-llm-geolocation.

Appendix C includes:

- **C.1 Error Distribution Plots:** Violin plots and boxplots showing error distributions for all methods, including outlier identification and quartile ranges
- **C.2 Geographic Error Maps:** 45-grant contact sheet displaying predicted vs. ground-truth coordinates on base maps, with color-coded error magnitude for spatial pattern analysis
- **C.3 Marginal Cost Analysis:** Cost-effectiveness curves showing marginal USD investment required to achieve 10 km accuracy across method tiers
- **C.4 Latency-Accuracy Pareto Front:** Scatter plot of mean error vs. processing time, identifying efficient frontier methods for time-constrained applications

D Tool Augmentation Analysis

Table **??** isolates the impact of providing tool access to identical models, revealing that tool augmentation does not consistently improve accuracy. For gpt-4.1-2025-04-14, enabling tool access increases mean error by 30.6%, while for the o4-mini model, it decreases error by 9.6%.

D.1 Direct Tool vs. Non-Tool Comparison

Table **??** provides a head-to-head comparison of identical models with and without tool access. This controls for model architecture effects and isolates the impact of tool access alone.

Table 10: Identical models with and without tool access; isolates the effect of tools.

Model	Category	mean	median	sd	min	max	1 km	5 km	10 km	25 km	50 km
gpt-4.1-2025-04-14	one shot	28.51	25.42	20.77	2.14	98.72	0.0%	4.7%	20.9%	48.8%	86.0%
gpt-4.1-2025-04-14	tool chain	37.23	34.22	23.94	0.59	101.85	2.3%	14.0%	16.3%	32.6%	74.4%
o4-mini-2025-04-16	one shot	41.65	27.39	27.32	7.59	103.49	0.0%	0.0%	7.0%	37.2%	62.8%
o4-mini-2025-04-16	tool chain	37.65	33.61	24.54	0.59	110.19	4.7%	11.6%	14.0%	32.6%	69.8%

D.2 Quantified Tool Effect

Table **??** quantifies the precise impact of tool access, showing percentage changes in mean error and percentage point (pp) changes in accuracy bands. " Mean %" shows percent change in mean error; "pp" indicates percentage point differences in accuracy bands. Negative percentages for mean change indicate worse performance with tools.

Table 11: Effect of tool augmentation (on mean error and accuracy bands).

Model	Mean M	Mean T	Mean %	1 km pp	5 km pp	10 km pp	25 km pp	50 km pp
gpt-4.1-2025-04-14	28.51	37.23	-30.6%	+2.3 pp	+9.3 pp	-4.7 pp	-16.3 pp	-11.6 pp
o4-mini-2025-04-16	41.65	37.65	9.6%	+4.7 pp	+11.6 pp	+7.0 pp	-4.7 pp	+7.0 pp

While the o4-mini model showed a modest improvement with tools, gpt-4.1-2025-04-14 performed substantially worse when given tool access.

D.3 Top-performing methods per tool-use category

Table **??** shows a direct head-to-head comparison of the best-performing tool-use method vs the best non-tool method. M-2 (o3-2025-04-16, one-shot prompt) substantially outperforms the best tool-augmented method (T-4), achieving a 37% lower mean error and nearly double the proportion of predictions within 10 km.

Table 12: Head-to-head of best non-tool vs best tool-augmented method.

Method	mean	median	sd	min	Q1	Q3	max	10 km	25 km	50 km
M (M-2)	23.39	14.27	19.86	2.67	8.17	36.85	87.35	30.2%	60.5%	93.0%
T (T-4)	37.23	34.22	23.94	0.59	21.78	53.35	101.85	16.3%	32.6%	74.4%

At the category level, the best non-tool method (M-2) significantly outperformed the best tool-augmented method (T-4) across all error metrics.

D.4 Tool Call Distribution

Table **??** expands on the tool usage patterns discussed in Section 6.6, providing detailed statistics on how each model interacted with the available geocoding and centroid-computation tools.

Table 13: Distribution of tool calls by method and tool type.

Method	Tool Type	Mean	SD	Median	Min	Max
T-1 (o4-mini)	geocode_place	3.79	2.41	3	1	10
T-1 (o4-mini)	compute_centroid	0.16	0.37	0	0	1
T-4 (gpt-4.1)	geocode_place	2.05	1.78	1	1	7
T-4 (gpt-4.1)	compute_centroid	0.25	0.43	0	0	1

D.5 ToolSearch Efficiency

"Selected call index" indicates which API call in the sequence produced the coordinates used in the final answer. Lower values indicate more efficient search strategies.

Table 14: Tool search efficiency (selected-call index; first-call success).

Method	Mean selected call index	Median	First-call success rate
T-1 (o4-mini)	2.29	1	69.0%
T-4 (gpt-4.1)	1.95	1	72.7%

The more economical approach of gpt-4.1-2025-04-14 is evident in both the distribution of calls and search efficiency. While T-1 (o4-mini) made nearly twice as many geocoding calls on average (3.79 vs. 2.05), it achieved a slightly lower first-call success rate (69.0% vs. 72.7%). This pattern aligns with the overall finding that tool augmentation does not consistently improve accuracy; in fact, the additional API calls may introduce noise through spurious matches to modern place names that bear little relation to colonial-era settlements.

Overall, both models heavily favored direct geocoding over centroid computation, with geocode:centroid ratios of 23.29:1 for T-1 and 8.18:1 for T-4. This suggests that the models primarily relied on finding exact matches for place names mentioned in the abstracts rather than triangulating from multiple reference points—a strategy that may explain their susceptibility to modern naming coincidences.

E Leakage Audit

Two heuristic checks were performed to assess training data leakage:

Google 15-gram search on random abstracts: No hits. Random 15-word sequences from the 43 evaluation abstracts were searched on Google to check for potential web presence. No matches were found, indicating the abstracts are not present in indexed public web sources.

Min-hash collision checks against the C4 corpus: No hits. Min-hash fingerprinting was applied to detect potential overlap with the C4 (Colossal Clean Crawled Corpus) dataset, a common training corpus for language models. No collisions were detected, suggesting no direct overlap with this major training dataset.

These tests reduce the risk of direct training-data leakage but cannot fully exclude indirect contamination given proprietary training data. The models may have been exposed to similar historical texts or geographic information through other sources not captured by these heuristics.

References

[] Bimbola Bashorun. Colonial virginia land grants geocoding dataset, April 2025. Expert GIS baseline created for comparative evaluation in geolocation research.

[] Michael R. Coughlan and Donald R. Nelson. Influences of native american land use on the colonial euro-american settlement of the south carolina piedmont. *PLOS ONE*, 13(3):e0195036, 2018. doi: 10.1371/journal.pone.0195036.

[] Direct Line Software. Deedmapper 4.2. Commercial software for historical deed mapping, 2010. URL https://directlinesoftware.com/deeds/. Windows application for plotting metes-and-bounds and public domain property descriptions.

[] G. Rebecca Dobbs. Backcountry settlement development and indian trails: A gis land-grant analysis. *Social Science Computer Review*, 27(3):331–347, 2009. doi: 10.1177/0894439308329760.

[] John Frederick Fausz. *Patterns of Settlement in the James River Basin, 1607-1642*. Master's thesis, College of William and Mary, 1971. URL https://dx.doi.org/doi:10.21220/s2-jedw-z031. Paper 1539624744.

[] Milan Gritta, Mohammad Taher Pilehvar, and Nigel Collier. Which melbourne? augmenting geocoding with maps. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (ACL 2018)*, pages 1285–1296, Melbourne, Australia, 2018. Association for Computational Linguistics. doi: 10.17863/CAM.27796. Introduces the CamCoder neural geocoder model.

[] Xuke Hu and Jens Kersten. Enhancing toponym resolution with fine-tuned LLMs (Llama2). In *Proceedings of the Second International Workshop on Geographic Information Extraction from Texts at ECIR 2024 (GeoExT@ECIR)*, pages 52–56. CEUR-WS.org, 2024. Fine-tuned Llama2 7B outperforms previous SOTA on toponym resolution by 13% and reduces mean error by 83%.

[] Chenghua Huang, Shisong Chen, Zhixu Li, Jianfeng Qu, Yanghua Xiao, Jiaxin Liu, and Zhigang Chen. Geoagent: To empower llms using geospatial tools for address standardization. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 6048–6063, Bangkok, Thailand, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.findings-acl.362.

[] Glenn M. Julian and Robbyn J. F. Abbitt. Gis and genealogy: Teaching gis while learning about the past. *ArcUser*, Spring 2014. URL https://www.esri.com/about/newsroom/arcuser/gis-and-genealogy. Project mapping Julian family land grants across McMinn, Hamilton, and Bradley Counties in Tennessee through multi-semester student GIS work.

[] Library of Virginia. Virginia land office patents and grants online database. Web portal, 2025. URL https://lva-virginia.libguides.com/land-grants. Scanned images and searchable indices for 1623–2000 patents grants; includes Northern Neck survey plats but no GIS polygons.

[] Loudoun County Office of Mapping. Original land grants of loudoun county, va [gis dataset]. Interactive ArcGIS WebMap, 2015. URL https://www.arcgis.com/apps/MapTools/index.html?appid=d3c4ba1031564f919ca28c9bb5a48350. Polygon shapefiles digitised from historian Wynne Saffer's USGS quad maps; covers all Northern Neck grants inside modern Loudoun County (early 1700s–1800s).

[] Beth Mitchell. *Beginning at a White Oak: Patents and Northern Neck Grants of Fairfax County, Virginia*. Office of Comprehensive Planning, Fairfax County, 1977. URL https://www.fairfaxcounty.gov/history-commission/sites/history-commission/files/assets/documents/resources/map.pdf. Includes companion historical GIS map of patent grant polygons.

[] Nell Marion Nugent. *Cavaliers and Pioneers: Abstracts of Virginia Land Patents and Grants, Volume Three: 1695-1732*. Virginia State Library, Richmond, Virginia, 1979. Transcribed from Patent Books 9-14, containing 5,471 grant abstracts.

[] Kent O'Sullivan, Nicole R. Schneider, and Hanan Samet. Metric reasoning in large language models. In *Proceedings of the 32nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (SIGSPATIAL '24)*, pages 501–504, 2024. doi: 10.1145/3678717.3691226.

[] Davide Savarro, Davide Zago, and Stefano Zoia. Leveraging large language models to geolocate linguistic variations in social media posts. *arXiv preprint arXiv:2407.16047*, 2024. doi: 10.48550/arXiv.2407.16047. GeoLingIt shared-task winner: fine-tuned LLM predicts both Italian region and coordinates for tweets.

[] David Schottlander and Tomer Shekel. Geospatial reasoning: Unlocking insights with generative ai and multiple foundation models.

Google Research Blog, April 2025. URL https://research.google/blog/geospatial-reasoning-unlocking-insights-with-generative-ai-and-multiple-foundation-models/. Accessed August 2025.

[] Yuxing Wu, Zhuang Zeng, Kaiyue Liu, Zhouzheng Xu, Yaqin Ye, Shunping Zhou, Huangbao Yao, and Shengwen Li. Text geolocation prediction via self-supervised learning. *ISPRS International Journal of Geo-Information*, 14(4):170, 2025. doi: 10.3390/ijgi14040170.

[] Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models. In *Proceedings of the 11th International Conference on Learning Representations*, 2023. doi: 10.48550/arXiv.2210.03629. URL https://arxiv.org/abs/2210.03629.