Systems-Level Framework for AI-Driven Transformation of Historical Data at Scale: Design Patterns for Modern Archival Ecosystems

Maksym Diachuk[1]

Received: 2026-02-17
Accepted: 2026-04-20
DOI: https://doi.org/10.5281/zenodo.21589261

Abstract. This paper aims to develop a systems-level framework for the AI-driven transformation of historical archival data at scale, with emphasis on architectural layers, reusable design patterns, and evaluation requirements for modern archival ecosystems. The study adopts a design science research approach and synthesizes literature, technical standards, and operational practices from archival informatics, data engineering, document image analysis, OCR/HTR, natural language processing, knowledge graph construction, metadata interoperability, and workflow orchestration. The proposed framework consists of five interrelated layers: Ingestion and Preservation, Recognition and Extraction, Semantic Enrichment, Knowledge Representation, and Orchestration and Delivery. It also identifies five reusable design patterns: Ingestion Gateway, Adaptive Recognition, Semantic Enrichment Chain, Graph Federation, and Orchestrated Provenance. These layers and patterns guide the integration of ETL pipelines, OCR/HTR models, NLP tools, CIDOC-CRM and PROV-O-based knowledge graphs, and cloud- or hybrid-orchestration workflows. The study provides a prescriptive reference architecture and evaluation protocol for scalable archival data processing. It concludes that a systems-level framework can bridge the gap between component-level AI research and deployment-oriented archival engineering, while providing a foundation for future empirical validation.

Keywords: archives, artificial intelligence, digitization, ontology, orchestration.

[1] MyHeritage Ltd., Israel, maxim.diachuk@yahoo.com

Introduction. Over the past few decades, the digitization of archival heritage has become a key area of development in the information society, ensuring the preservation, accessibility, and reuse of historical data for scientific, cultural, and social purposes.

The mass digitization of archival documents, manuscripts, cartographic materials, periodicals, and other historical sources has led to the creation of digital collections of unprecedented scale. At the same time, a significant portion of these resources remains insufficiently structured, contains heterogeneous data formats, text recognition errors, and limited capabilities for automated analysis. As a result, archival institutions face the challenge of effectively transforming historical data into information resources suitable for machine processing and intelligent use.

Recent advances in artificial intelligence (particularly in machine learning, natural language processing, computer vision, and generative models) are opening new possibilities for automating the extraction of knowledge from historical documents. Such technologies enable text recognition, semantic annotation, entity detection, data reconstruction, document classification, and the establishment of connections among historical objects.

However, implementing individual artificial intelligence tools does not guarantee the creation of a comprehensive and scalable historical data management system. The problem of integrating heterogeneous components into a unified architecture that ensures the reliability, reproducibility, interoperability, and scalability of archival processes remains particularly acute.

The research problem lies in the lack of a universal, systematic approach to designing archival ecosystems that effectively combine the capabilities of artificial intelligence with the requirements of long-term preservation, metadata management, and processing large volumes of historical information. Existing solutions primarily focus on individual stages of the data lifecycle and do not adequately address the need for comprehensive coordination of digital transformation processes across the entire archival infrastructure.

The relevance of this study stems from the need to develop a system-level conceptual framework that will ensure coordinated interaction among archival repositories, data processing services, artificial intelligence tools, and knowledge management mechanisms.

This approach will improve the efficiency of modern archival institutions, reduce the costs of processing historical collections, enhance metadata quality and expand the possibilities for the analytical use of archival resources in scientific research.

The purpose of this article is to develop and justify a systematic framework for the large-scale transformation of historical data using artificial intelligence technologies, and to identify key design patterns for building modern archival ecosystems.

To achieve this goal, the following steps are planned: to analyze the challenges of the digital transformation of historical data; to investigate the architectural principles for integrating AI components into archival environments; to systematize design patterns for processing, enriching, and managing historical data; and to develop a model of component interaction focused on scalability, interoperability, and long-term support for archival information systems.

The scientific novelty of this study lies in the development of a systematic view of the archival ecosystem as a multilevel intellectual platform, in which artificial intelligence serves not as a separate automation tool but as a fundamental element of the historical data management architecture. The proposed approach aims to establish the theoretical and practical foundations for building next-generation archives capable of effectively handling large volumes of cultural and historical information amid society's digital transformation.

The following research questions guide the study:

RQ1: What architectural layers are required for AI-driven processing of historical data at scale?

RQ2: Which reusable design patterns can support practical implementation across heterogeneous archival institutions?

RQ3: What evaluation dimensions and indicative target values should guide the assessment of such systems?

Unlike prior studies that primarily focus on isolated OCR or NLP components, this work proposes a systems-level reference architecture and a set of reusable design patterns for large-scale historical data transformation.

The remainder of the paper is organized as follows. Section 2 reviews the related literature; Section 3 presents the methodology; Sections 4-7 describe the proposed framework, design patterns, interoperability standards, and evaluation protocol; and Sections 8-9 discuss implications, limitations, future research, and conclusions.

Background and related work.

Archival informatics and the digital turn

Archival informatics, the intersection of Archival Science and computer science, has undergone a qualitative shift in the age of mass digitization (Opgenhaffen, 2022). The Open Archival Information System (OAIS) and other foundational digital preservation models provided the first controlled, standards-based architecture for preservation and access of digital objects. The OAIS model is a reference standard for preservation architecture and will inform the framework's ingestion layer.

For the past twenty years, balancing preservation fidelity and the accessibility of the computation has been at the heart of archival informatics. Other more recent works use Artificial intelligence simply as a tool for more efficient cataloging, but as a more agentive tool for fundamentally restructuring the relationship between archive users and the archive, thus enabling more sophisticated and large-scale research (Jaillant & Zhao, 2025).

OCR and HTR for historical documents

Text recognition for history has been a main topic in document image analysis. Traditional OCR systems used in modern libraries fail on historical documents due to poor print quality, archaic typefaces, degraded substrates, and limited training data for rare languages (Lombardi & Marinai, 2020). Some historical OCR systems, such as OCRopus and Kraken, have introduced trainable, institution-specific neural network frameworks (Momtaz et al., 2025).

Handwritten text recognition represents an even greater challenge. The 2010s were dominated by LSTM-based HTR systems (Graves & Schmidhuber, 2009; Puigcerver, 2017). More recently, transformer-based systems for HTR have shown notable advances in generalization (Li et al., 2021; Kang et al., 2022).

The Transkribus platform has offered HTR at an institutional scale, with the recognition of multiple hands (Muehlberger et al., 2019). However, there is still zero-shot generalization to unseen historical hands, a need for better layout analysis to handle more complex documents, and the integration of non-textual elements (Seuret et al., 2021; Clausner et al., 2019). Layout analysis tools like Layout Parser (Shen et al., 2021) have made deep learning-based document segmentation easier for humanities researchers.

NLP for historical corpora

The main difference between historic text and modern text is that modern text is more standardized. Researchers studying modern text processing recognize that varied spelling and orthographic changes can adversely affect NLP-based tools. However, character-level neural machine translation has led to improved performance in most downstream NLP tasks that rely on spelling (Bollmann, 2019; Scherer & Ljubešić, 2016). The HIPE shared tasks serve as benchmarks for named entity recognition across different historical periods (Ehrmann et al., 2020; Ehrmann et al., 2022). They help establish baseline performance levels and encourage

future research into challenges related to historical named entity linking (Grover et al., 2012; van Hooland et al., 2013).

NLP for history has benefited from large language models via few-shot hypothesis generation. However, in historical research, their tendency to hallucinate is devastating, as archivists cannot be sure of the accuracy of the provenance record (Bender et al., 2021; Meng et al., 2022). NLP tools that focus on temporal expressions, like HeidelTime (Strötgen & Gertz, 2013), can help bind historical text to ISO 8601. The Stanza toolkit (Qi et al., 2020) and spaCy (Honnibal et al., 2020) assist in developing flexible frameworks for historical text.

Knowledge graphs and cultural heritage ontologies

Knowledge graph technology has become the principal way of representing the intricately connected field of cultural heritage knowledge (Hyvönen, 2012; *Hyvönen* 2020; Noy et al., 2019). The CIDOC Conceptual Reference Model (ISO 21127) offers an event-centric model (ontology) to document cultural heritage (Doerr, 2003; Doerr & Hagedorn-Saupe, 2020).

The W3C PROV-O ontology provides vocabulary for representing provenance of various pipelines (Lebo et al., 2013). To build a knowledge graph using these ontologies, the framework provides an alignment solution via a specialized mapping layer.

The Linked Art project has created a CIDOC-CRM JSON-LD profile to make the CIDOC-CRM more developer-friendly, and the Europeana Data Model (EDM) aims to provide a more lightweight, aggregation-based perspective (Isaac & Haslhofer, 2013).

At the graph population level, the combination of Sentence-BERT embeddings and structural graph features has been explored for determining whether two mentions referring to a real-world entity are the same (Grover et al., 2012).

Data engineering, etl pipelines, and workflow orchestration

AI archival data transformation requires substantial architecture for industrial-grade data engineering (Kleppmann, 2017; Reis & Housley, 2022). The orchestration of workflows using directed acyclic graphs (DAGs) to improve resilience to pipeline failures is supported by Apache Airflow, Prefect, Dagster, and Argo Workflows (Harenslak & de Ruiter, 2021).

Additionally, the deployment of scalable AI processing pipelines relies on standard containerization and Kubernetes orchestration (Burns et al., 2022). The hidden technical debt of production machine learning systems remains a concern and requires management through model monitoring and versioning (Sculley et al., 2015).

FAIR data principles and archival interoperability

The FAIR data principles: Findability, Accessibility, Interoperability, and Reusability, help in the assessment of research data management (Wilkinson et al., 2016). Their adaptation to archived data has been discussed in the digital humanities research infrastructure (Harrower et al., 2020; Mons et al., 2017).

The International Image Interoperability Framework (IIIF) is the standard for delivery and annotation of images in the cultural heritage sector (Snydman et al., 2015; Cramer, 2011). The primary harvesting protocols for Europeana and DPLA are OAI-PMH, and linked data queries in federated graph stores are made via SPARQL endpoints.

Methodology: Design science research approach

Research design

This study adopted a Design Science Research (DSR) approach to develop a systems-level framework for the AI-driven transformation of historical archival data at scale. DSR was considered appropriate because the purpose of the study was not to test a single deployed system but to design a prescriptive artifact to guide future implementation, evaluation, and adaptation across archival institutions. The methodological orientation follows the DSR

tradition, which emphasizes identifying a practical problem, defining solution objectives, developing an artifact and communicating its relevance to both research and practice.

In this study, the artifact is a five-layer reference architecture supported by reusable design patterns and an evaluation protocol. The framework was developed to address the systems-engineering gap between component-level AI research and the practical requirements of scalable, interoperable, and auditable archival data processing.

Source selection and data collection

As a framework-development study, the data consisted of published literature, technical standards, methodological frameworks, and operational examples from archival and AI-driven information systems.

Sources were selected from areas directly relevant to the transformation of digitized historical records, including archival informatics, digital preservation, document image analysis, OCR/HTR, natural language processing, entity linking, knowledge graph construction, metadata interoperability, ETL pipelines, and workflow orchestration.

The source materials included peer-reviewed studies on archival informatics and digital preservation; technical literature on OCR, HTR, layout analysis, historical NLP, named entity recognition, and entity linking; and standards and frameworks such as OAIS, PREMIS, CIDOC-CRM, PROV-O, RDF/SPARQL, OAI-PMH, and IIIF.

Operational examples and implementation practices from platforms and infrastructures such as Transkribus, Europeana, Apache Airflow, Kubernetes, and graph database systems were also considered, as they illustrate how archival, data engineering, and AI components can be deployed in real-world environments.

The selected sources were not treated as empirical performance data. Instead, they provided the conceptual and technical basis for identifying common archival processing requirements, recurring integration challenges, and reusable architectural solutions.

Analytical and synthesis procedure

The analysis was conducted through conceptual synthesis and comparative systems analysis. First, the reviewed literature and operational examples were examined to identify recurring problems in large-scale archival processing. These problems included heterogeneous input formats, limited content-level access, recognition errors, weak semantic enrichment, fragmented metadata, poor provenance tracking, limited interoperability, and difficulties in scaling AI workflows. (Jaillant & Zhao, 2025).

Second, the identified problems were mapped to corresponding technical functions required in an archival AI ecosystem. These functions included ingestion, preservation, recognition, extraction, semantic enrichment, knowledge representation, workflow orchestration, quality control, human review, and delivery through interoperable interfaces.

Third, the mapped functions were grouped into broader architectural categories. This process led to the development of five interrelated layers: Ingestion and Preservation, Recognition and Extraction, Semantic Enrichment, Knowledge Representation, and Orchestration and Delivery. Each layer was defined by its purpose, inputs, outputs, core technologies, quality-control mechanisms, and its relationship to the other layers.

Framework development

The framework was developed by aligning archival requirements with AI/ML, data engineering, semantic web, and workflow orchestration components. The development process focused on ensuring that each layer addressed a specific stage of archival data transformation while remaining interoperable with the other layers.

The first layer, Ingestion and Preservation, was designed to support format identification, fixity checking, preservation packaging, and provenance initiation.

The second layer, Recognition and Extraction, was designed to convert images and handwritten or printed records into machine-readable text with layout and confidence metadata.

The third layer, Semantic Enrichment, was designed to support named entity recognition, entity linking, temporal normalization, and annotation.

The fourth layer, Knowledge Representation, was designed to convert enriched archival data into formal graph-based knowledge structures using standards such as CIDOC-CRM and PROV-O.

The fifth layer, Orchestration and Delivery, was designed to coordinate workflow execution and expose the transformed data through interfaces such as SPARQL, REST/GraphQL, OAI-PMH and IIIF (Harenslak & de Ruiter, 2021).

In addition to the layered architecture, five reusable design patterns were derived from recurring implementation needs observed across the reviewed literature and operational systems. These design patterns are Ingestion Gateway, Adaptive Recognition, Semantic Enrichment Chain, Graph Federation, and Orchestrated Provenance. Each pattern was formulated to describe a common architectural problem, the context in which the problem occurs, the forces shaping the design decision, the proposed solution, implementation considerations, consequences, and applicability.

Evaluation protocol formulation

The evaluation protocol was formulated by translating the functional requirements of the proposed framework into measurable quality indicators. The aim was to provide a structured basis for assessing future implementations of the framework rather than to report results from a deployed system. Evaluation dimensions were derived from established assessment practices in document image analysis, historical NLP, metadata quality assessment, provenance modeling, interoperability testing, and data engineering reliability. The resulting dimensions include recognition quality, semantic enrichment accuracy, metadata completeness, provenance completeness, pipeline throughput, latency, human review burden, graph query performance, interoperability coverage, and system reliability (Constum et al., 2022).

The indicative target values presented in the evaluation protocol are therefore proposed as benchmarks for future empirical validation. They do not represent achieved results from a current implementation. Instead, they serve as hypotheses and practical reference points that future studies may test using standardized archival datasets, prototype systems, and multi-institutional case studies.

Methodological scope and limitations

The methodology is conceptual and design-oriented. It does not involve primary data collection from human participants, experimental deployment, or statistical testing of system performance. Its contribution lies in the structured development of a systems-level framework that integrates existing knowledge across archival science, AI/ML systems, data engineering, and semantic technologies. The main limitation of this methodological approach is that the proposed framework requires future empirical validation. A prototype implementation should be tested using archival collections that vary by document type, language, historical period, material condition, and institutional context. Such validation would enable assessment of the practical performance, scalability, reliability, and generalizability of the proposed framework.

Proposed system architecture

Architectural overview

The proposed framework for AI-driven archival data transformation includes a five-layer reference architecture. The architecture is conceptually illustrated in Figure 1, with detailed descriptions provided in following sections.

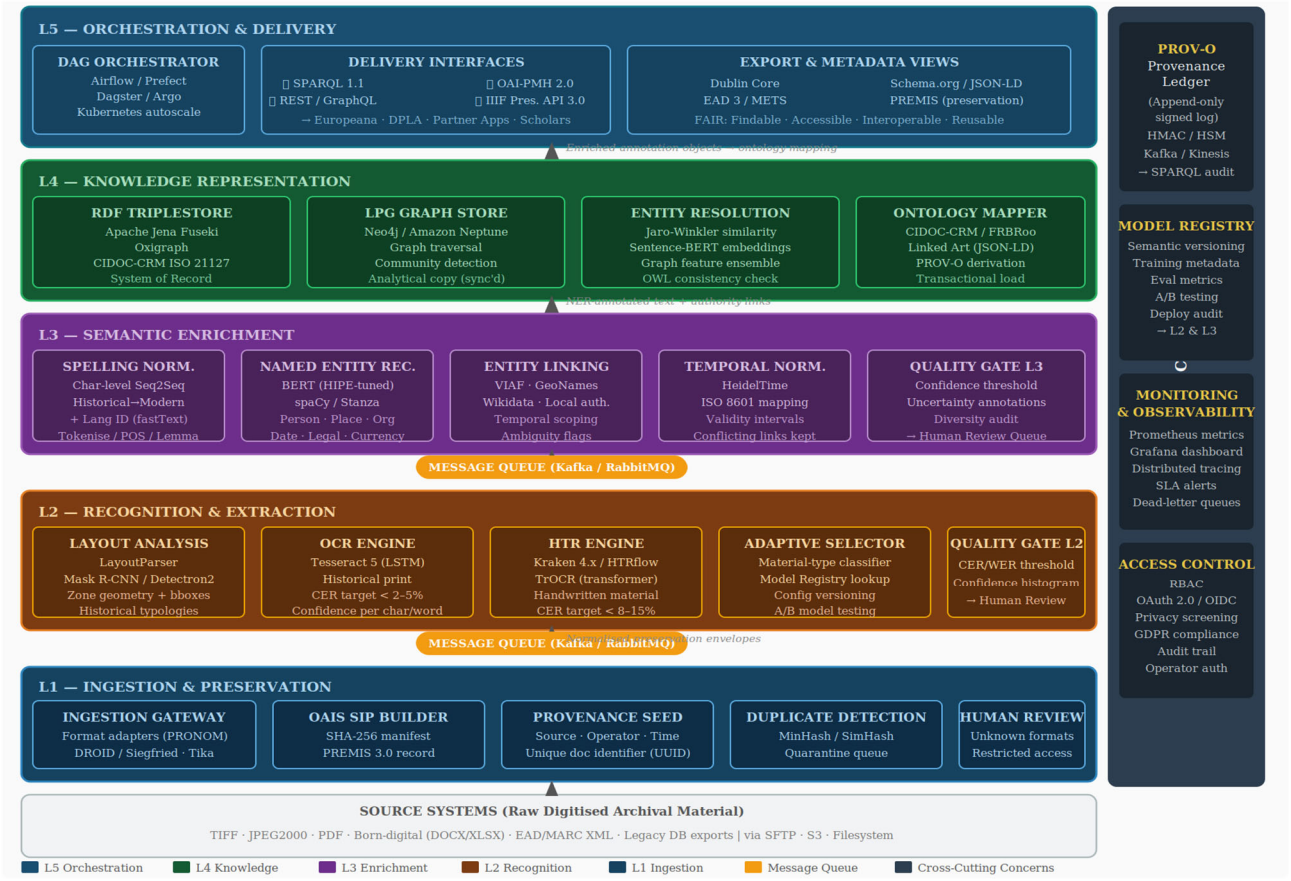

Figure 1 Description (for diagram conversion)

The figure shows a 5-layer horizontal stack design (L1–L5) and represents data flows from raw archival objects at the base to semantic knowledge services at the top. A vertical band titled “Cross-Cutting Concerns” that runs the height of the five layers is divided into four persistent components: Provenance Ledger (PROV-O), Model Registry (versioned AI models), Monitoring and Observability (Prometheus/Grafana) and Access Control (RBAC). The data flow moves from the base to the top. Raw digitized files enter Layer 1 (L1) at the Ingestion Gateway. Preservation envelopes transit to Layer 2 (L2) over a durable message queue (Kafka/RabbitMQ). Annotated text DOMs transit to Layer 3 (L3) over a second queue. Further processed layered annotation objects and transits to L4 for ontology and graph population. The L5 layer exposes knowledge services through four delivery interfaces: SPARQL, REST/GraphQL, OAI-PMH, and IIIF. The Human Review Queue is a lateral branch of both L2 and L3, and the reviewed outputs are reintegrated into their respective layers. The Model Registry is a shared service that connects to L2 and L3, allowing for versioned models and the ability to log audits for each inference (Table 1).

Table 1. Five-Layer Reference Architecture for AI-Driven Archival Data Processing

Layer	Name	Core Technologies	Key Outputs	Quality-Control Mechanism
L1	Ingestion & Preservation	DROID/Siegfried, SHA-256, OAIS SIP, Kafka	Normalized preservation envelope + provenance seed	Format validation, checksum, duplicate detection
L2	Recognition & Extraction	Kraken/HTRflow, Tesseract, LayoutParser	Confidence-annotated text + zone geometry DOM	CER/WER gates, confidence histogram, human review queue

L3	Semantic Enrichment	spaCy/Stanza, BERT NER, entity linker, HeidelTime	Named entity annotations with VIAF/Wikidata links	Per-entity confidence, temporal scoping and ambiguity flags
L4	Knowledge Representation	Apache Jena/Oxigraph (RDF), Neo4j (LPG), CIDOC-CRM mapper	CIDOC-CRM RDF graph + LPG analytical copy	OWL consistency, entity resolution metrics and graph validation
L5	Orchestration & Delivery	Apache Airflow, Kubernetes, SPARQL, OAI-PMH, IIIF	Structured knowledge services + audit trail	SLA monitoring, health dashboard, dead-letter queues

Layer 1: Ingestion and preservation

Purpose. To accept digitized archival materials from diverse source systems, determine bitstream integrity, normalize to a standard preservation format, and document a provenance seed record prior to any destructive or probabilistic alterations.

Inputs. TIFF, JPEG2000, PDF, born-digital office documents, EAD/MARC export files, legacy database records; delivered via SFTP, S3-compatible object storage, or direct filesystem mount.

Outputs. An OAIS-compliant Submission Information Package (SIP) that contains the normalized preservation bitstream, a PREMIS 3.0 metadata record, a SHA-256 manifest, a source provenance record, and a creation routing manifest with the processing configuration for the asset type.

Core technologies. Identify formats with DROID or Siegfried (PRONOM registry). Convert formats with Tika and LibreOffice. Use Kafka or RabbitMQ for decoupling data and processing streams.

Communicate format and encoding issues, including failure modes. Checksum mismatches trigger requests for retransfers. Manual review alerts occur when formats cannot be identified. All unidentified formats are queued in quarantine. A duplicate registry detects and routes re-ingested materials to a deduplication workflow.

Human-in-the-loop checkpoint. Assets are sent to a staffed review queue for further processing when flagged as having unspecifiable formatting, suspected duplicate entries, or restricted-access markers.

Provenance and security requirements. Every ingest event is documented on an immutable PROV-O provenance ledger. Role-based access control (RBAC) restricts ingest permissions to approved operators.

Layer 2: recognition and extraction

Purpose. To convert image assets to a structured text format with spatial organization and confidence levels at the character level.

Inputs and Outputs. Normalized images from level one. Routing manifest. Outputs include a Document Object Model (DOM) containing text recognition zones with bounding boxes for each zone, confidence levels for each word and character, and a recognition provenance history that states the model version and the parameters used for inference.

Core Technologies. Layout analysis can use either LayoutParser (Shen et al., 2021) or Detectron2 with a segmentation model w/ fine-tuning for historical documents. OCR can use Tesseract 5 (with LSTM) for historical prints. Handwritten text recognition (HTR) can use Kraken 4.x with HTR flow for handwritten documents, or TrOCR (Li et al., 2021) for HTR using transformer models.

Technical Challenges. The recognition layer handles documents in various conditions, from good to very bad. Using data augmentation with noise injection has been proposed to handle this variability (Wigington et al., 2017). Dealing with complex layouts of historical documents, including multiple columns with marginal notes and interleaved tables, will require training models on different layouts from that historical period.

Human-in-the-loop Checkpoint. Tools such as Transkribus and eScriptorium for transcription review enable a continuous loop of improvement, as errors are corrected and fed back into the Model Registry as ground truth.

Model Versioning. Every model used for recognition is uploaded to the shared Model Registry with semantic versioning, metadata for the training corpus, evaluation metrics, and the release date, allowing the recognition process to be repeated to the same standards.

Layer 3: semantic enrichment

Purpose. Integrate a multi-step Natural Language Processing (NLP) arrangement that converts unorganized shouted text to a semi-structured digital record with named entities, time references, and journal citation links.

Core technologies. Language Classification, fastText. Standardized spelling, character-level Seq2Seq (Bollmann, 2019). Tokenization and Part-Of-Speech tagging, spaCy or Stanza. Named Entity Recognition (NER), BERT HIPE benchmark (Ehrmann et al., 2022), fine-tuned model. Linkage, REL or wikimapper via CITAF and Wikidata. Temporal normalisation, HeidelTime (Strötgen & Gertz, 2013).

Temporal entity scoping. Each entity-authority linkage has a validity period, determined by local date expressions. It is based on historical fluctuations in names and boundaries, with conflicting connections retained as alternatives for review.

Failure modes and quality control. Low confidence NER outputs are marked as uncertain. A diversity audit component is included to measure and review the output of entity recognition across languages and time frames, examining potential bias in the training data.

Layer 4: knowledge representation

Purpose. The goal is to map enriched annotation objects to a formal semantic model, determine the identities of entities within and across documents, and deposit organized knowledge in a graph store that can be discovered and analyzed.

Core technologies. The main triplestore: Fuseki, Apache Jena or Oxigraph. Analytical LPG: Neo4j or Amazon Neptune. Ontology: CIDOC-CRM (ISO 21127) with the FRBRoo extension and the Linked Art JSON-LD profile. Provenance: PROV-O (Lebo et al., 2013). Entity resolution: Jaro-Winkler string similarity and Sentence-BERT embedding similarity.

Dual-graph architecture. The system contains both an RDF triplestore (the system of record and the publication of linked data) and an LPG store (optimized for graph traversal and community detection). A synchronization service copies all writes from the RDF store to the LPG store in a bidirectional way, relying on event-driven delta computation (Frey et al., 2017).

Quality control. OWL consistency checking is used to find ontological contradictions before loading the graph. Entity resolution metrics are evaluated periodically to identify model drift. The graph's population is transactional, and failed loads are retried from a persistent log.

Layer 5: Orchestration and Delivery

Purpose. To facilitate the scheduling and tracking of pipeline execution across the four processing layers and the management of computational resources to meet service-level objectives. Additionally, the purpose is to expose archival knowledge using standard-conformant delivery interfaces.

Orchestration Model. Processing pipelines are modeled as DAGs in Apache Airflow and its equivalents (such as Prefect, Dagster, and Argo Workflows). Each node in the DAG corresponds to a processing stage in the pipeline, and is configurable with I/O contracts, retry

policies, timeout policies, and resource specifications. Deployment is on Kubernetes, enabling horizontal pod autoscaling for processing stages based on the queue size.

Delivery Interfaces. (i) SPARQL 1.1 Endpoint for RDF Query; a standards-compliant RDF query; (ii) a REST/GraphQL API for access at the application level; (iii) OAI-PMH 2.0 for data harvester access; (iv) IIIF Presentation API 3.0 for delivering images and annotations.

Monitoring and Observability. A Prometheus-compatible metrics endpoint exposes throughput, recognition accuracy distribution, entity-linking hit rates, graph population latency, and queue heights. Grafana is used to visualize operational health. Structured logging with a distributed trace identifier propagated at each stage allows for end-to-end traceability.

Canonical design patterns

This section describes five foundational design patterns that encapsulate reusable architectural decisions for deploying archival AI.

These patterns employ the formal structuring of Gamma et al., (1994) which we adapted for data engineering and AI systems architecture (Fehling et al., 2014).

Pattern 1: Ingestion Gateway

Problem. Archival materials come from various digitized documents, repositories, deposits, transfers, and migrations, creating outdated databases. Each of these has its own file formats, metadata schemas, transfer protocols, and quality profiles.

Context. A large consortium of sources or a consortium of sources with different spaces and different IT structures and different institutions.

Forces. The processing pipeline requires a stable and uniform input contract. Variability in source formats manifests as fragility in downstream components. Provenance capture must be atomic with ingest to prevent a processed record from existing without a traceable origin. For format-appropriate validation, checksum computation must follow format identification.

Solution. Establish a single logical ingestion gateway that standardizes all incoming data into a preservation envelope. The gateway uses contract-based ingestion and format-based adapters. The gateway identifies and confirms formats, creates preservation checksums, secures origin traceability, assigns a unique document ID, and places the envelope on a durable message queue.

Implementation. In a format registry keyed on PRONOM PUID identifiers, format adapters are registered. The message queue (Kafka, RabbitMQ) guarantees that a consumer's offset is tracked and that messages are delivered at least once. The ingest contract is versioned independently of the pipeline code, allowing onboarding systems to be added without redeploying the pipeline.

Consequences. Systems that handle variability in data sources can develop components downstream of processes. All material will contain a provenance record and a checksum. The gateway creates a bottleneck in processing; the disruption must be addressed to meet peak digitization output rate estimates.

Applicability. Required for all deployments that deal with material from multiple source systems. Can be reduced to one adapter for highly homogeneous, single-source deployments.

Pattern 2: Adaptive Recognition

Problem. No single recognition model can achieve an acceptable level of performance on all types of material, spanning multiple periods and languages in historical corpora.

Context. Though a sub-collection requires different OCR or HTR configurations, a collection can span multiple centuries or languages.

Forces. The performance of a model dedicated to a sub-collection is diluted by training on all materials. Having different pipelines for each material type increases operational complexity. Model choices must be reproducible and auditable, and new model versions must be added to the system without disrupting ongoing processing.

Solution. Create an adaptive recognition engine that selects, combines and sets parameters for different models based on a material-type classifier. The classifier assigns the documents to a recognition configuration, which is maintained in a Model Registry. The configuration and models can be updated without changing the pipeline code.

Implementation. The classifier is based on image features and metadata. A/B testing in the Model Registry allows for new model versions to be evaluated against the production traffic.

Consequences. Maximum quality of recognition per material type is attained. Changes to models are auditable and reversible. Adding the classification and registry lookup for each recognition incurs latency that is not noticeable compared to the time recognition takes to infer.

Applicability. This solution is applicable wherever a collection spans more than a century and multiple script traditions. For very homogeneous collections, a fixed configuration may be sufficient.

Pattern 3: Semantic Enrichment Chain

Problem. Enriching the semantics of historical texts involves an NLP pipeline in which each component is uncertain, resulting in compounding uncertainty.

Context. An NLP pipeline processing texts from different languages, eras, and domains should allow for component failures without terminating the entire system.

Forces. The stages of an NLP pipeline are interdependent: for example, spelling normalization affects named entities, and named entities affect entity resolution. It may be necessary to implement distinct components for different material types, but the stages themselves may remain constant.

Solution. Construct a series of operations (stages) where each step (component) constructs a typed annotation schema and is expressed as a dependency on an adjacent step. Each component interprets the typed schema and is internally opaque to other steps.

Implementation. Typed schemas are outlined using either JSON Schema or Apache Avro, both of which allow for iteration without compatibility loss. Each step is implemented as a microservice, and independent steps or stages may be implemented in parallel (horizontal scaling).

Consequences. The robustness of the system is enhanced because a single step (component) may fail without invalidating the entire pipeline. The ability to independently deploy each step (component) allows for the replacement of language-specific models. The dependency on a shared annotation may create an I/O bottleneck, and will be supplemented with asynchronous storage.

Applicability. This approach is adaptable to all multi-stage NLP tasks, especially those involving multiple languages or eras.

Pattern 4: Graph Federation

Problem. Relevant data is found across multiple institutional records, and each data bank contains incompatible entity identifiers and ontological issues with its knowledge graphs.

Context. A challenge that requires multi-institutional collaboration to progress, where cross-institutional discovery is necessary and cannot be resolved with a single knowledge graph.

Forces. Data centralization is politically and technically unfeasible. Local SPARQL road endpoints host institutional data, but cross-graph co-reference is not supported. Links to authority files provide a basis for alignment, but their application is inconsistent.

Solution. Develop a federation layer above the institutional knowledge graphs that implements a query-planning and rewriting engine to decompose cross-repository SPARQL queries. The engine will execute sub-queries in parallel and merge the results to implement

entity co-reference resolution. The shared authority mapping table will support a locally maintained URI to a shared authority (VIAF, Wikidata) identifier.

Implementation. Federation query planning uses a cost-based optimizer to estimate sub-query cardinality and statistics from a cached endpoint. The authority mapping table will be a persistent key-value index and will be updated asynchronously while Entity Linking is running. Participating endpoints must agree to and cooperate under a defined data-sharing agreement.

Consequences. The federation layer enables cross-institutional discovery without centralized control. The firmware layer adds query latency in direct proportion to the number of endpoints participating. The authority mapping co-reference resolution is directly dependent on the quality of the authority mapping.

Applicability. This can be used in situations that require cross-institutional discovery. The case is not recommended when query latency would be affected by federation overhead.

Pattern 5: Orchestrated Provenance

Problem. In archiving, every record must be traceable to its full processing history for appropriate citations, legal compliance, and audit accountability.

Context. Any archival AI designed to process records used in scholarship, the law, or any form of institutional accountability.

Forces. Provenance must be collected at every single processing step. The record must be structurally secure and tamper-evident. Traceability must not significantly increase the processing time. Records must be retrievable using the same frameworks as the archival contents.

Solution. The orchestration layer communicates a document-level provenance record throughout the processing pipeline. Every stage appends a processing event that includes the stage name, the component version, input and output checksums, time taken, confidence Level, and the reviewer (for reviewer processing steps). This record is then stored along with a PROV-O derivation chain, timestamped by a trusted timestamp service, and added to the knowledge graph.

Implementation. The provenance ledger is implemented as a compacted, append-only distributed log (e.g., Apache Kafka or AWS Kinesis) for the knowledge graph. Records are HMAC-signed with a key held in an HSM. The provenance graph, along with SPARQL, allows us to run audit queries for, say, “show all records processed by model version X.Y.Z for the time period of A to B".

Consequences. Every record contains the full history of its processing in a verifiable manner. At a large scale, the provenance records add a non-trivial amount of storage, approximately two to five kilobytes, and must be accounted for.

Applicability. Mandatory for any deployment targeting scholarly, legal, or institutional accountability. Optional for any deployment of a completely unbounded and exploratory nature.

Metadata standards and interoperability

Interoperability is a key element of any efficient archival ecosystem, allowing for collaboration with partner repositories, aggregators, and discovery services.

This proposed framework utilizes a layered metadata strategy to achieve interoperability while keeping deeper, complex, internal structures and views that comply with metadata export standards.

This framework adopts the FAIR principles (Wilkinson et al., 2016) to meet interoperability requirements at different levels.

Specifically, Findability of data is achieved through DOI and VIAF persistent identifiers and Schema.org metadata; Accessibility is ensured through delivery interfaces (SPARQL, OAI-PMH, IIIF) where the default license is the CC BY 4.0; Interoperability is achieved through

alignment of CIDOC-CRM with CIDOC-CRM and JSON-LD; and Reusability is achieved through versioned datasets and provenance records. Standards Integration is presented in Table 2.

Table 2. Metadata and Interoperability Standards in the Proposed Framework

Standard	Function in the Framework	Architecture Layer	Interoperability Benefit
OAIS (ISO 14721)	Defines SIP/AIP/DIP preservation package model	L1, L5	Universal digital preservation standard; cross-system package exchange
EAD 3	Encodes archival finding aid hierarchical description	L1, L5	Finding aid exchange with national and international aggregators
METS	Wraps structural metadata, file inventory, and behavior specs	L1, L5	Interoperability with digital library management systems
PREMIS 3.0	Records preservation metadata: format, rights, provenance, fixity	L1	Long-term preservation management and format migration audit
Dublin Core	Lightweight descriptive metadata for aggregator harvesting	L5	Supported by Europeana, DPLA, and OAI-PMH harvesters
CIDOC-CRM (ISO 21127)	Primary ontological backbone for a semantic knowledge graph	L4	Event semantics: alignment with international museum and archive data
PROV-O (W3C)	Formal representation of processing derivation chains	L1–L5	Data lineage tools; scholarly citation of provenance
RDF / SPARQL	Serialization and query language for the knowledge graph	L4, L5	Linked Data publication; federated query across endpoints
JSON-LD	Lightweight linked data serialization for web APIs	L4, L5	Developer-accessible linked data; Linked Art profile
OAI-PMH 2.0	Protocol for metadata harvesting by aggregators	L5	Europeana, DPLA, national aggregator integration
IIIF Presentation API 3.0	Image delivery, manifest structure, annotation protocol	L5	Cross-institutional image comparison and scholarly annotation
Schema.org	Structured metadata for web discovery and search indexing	L5	Findability via Google Dataset Search and web indexing

Evaluation Protocol for Archival AI Systems

Overview

This section outlines a framework for evaluating implementations of the framework. It includes an evaluation dimension and target metrics. The values in Table 3 are examples of what future empirical validation may yield, and the framework design keeps in mind that no implementation has yet to achieve these metrics.

Table 3. Indicative Target Values for Evaluation Dimensions (Proposed for Future Empirical Validation)

Evaluation Dimension	Metric	Indicative Target	Notes
OCR 19th-c. print	CER	< 2%	High-quality scan; after post-processing
OCR 18th-c. print	CER	< 5%	Degraded typography; Gothic typefaces
HTR clear hand	CER	< 8%	Well-documented contemporary script
HTR difficult hand	CER	< 15%	Archaic, damaged, or cursive scripts
NER person entities	F1	> 0.80	Post spelling normalization, based on HIPE benchmarks
NER place entities	F1	> 0.78	Historical toponyms are more variable
Entity linking accuracy	% correct links	> 75%	Linked to VIAF/GeoNames/Wikidata
Metadata completeness	% fields populated	> 85%	Mandatory + recommended fields weighted
Provenance completeness	% with full PROV-O chain	> 99%	Non-negotiable for archival authenticity
Pipeline throughput	Pages/hour	400–1,200	Depends on GPU allocation and document complexity
Human review burden	% routed to review	< 15%	Target for mature deployment; higher initially
SPARQL query (simple)	Response time	< 200 ms	Entity lookup; p95 under 50 concurrent users
SPARQL query (multi-hop)	Response time	< 5 s	3–5 hop traversal, with caching
Interoperability coverage	% schema-valid	> 98%	Across all four delivery interfaces
System reliability	Failure rate	< 1%	Manual interventions per 1,000 documents

Note: Indicative target values are proposed for future empirical validation and do not represent results achieved by any deployed system. Actual performance will vary by collection characteristics, infrastructure configuration, and model training data quality.

Evaluation dimensions

Recognition Quality. Character Error Rate and Word Error Rate signal the quality of recognition (Fischer et al., 2012). They are computed for OCR and HTR outputs against transcribed ground truth (GH) at the page and line levels. For GH, stratification by material type, period, and language is required, as it can help in identifying systematic changes in performance.

Semantic Enrichment Accuracy. For HIPE (Ehrmann et al., 2022), this is measured by precision, recall, and F1-score for person, places, organizations, and dates. The accuracy of entity linking is determined by the number of correctly linked entities relative to the annotational benchmark.

Metadata Completeness. The complete combination of required and recommended metadata fields in a processed record is evaluated at the document level (METS/EAD completeness) and at the entity level (linked authority entities), as per Park and Tosaka (2010).

Provenance Completeness. The number of processed records that have a complete PROV-O derivation chain, traceable to the original preservation ingest event, and that pass signature validation against the provenance ledger.

Pipeline Throughput and Latency. The pages processed per hour under normal and peak conditions are evaluated at each stage of the pipeline to determine throughput and latency.

Human Review Burden. For documents reviewed, this includes the average time of an operator to review, as well as the distribution of documents in the L2 and L3 human review queues. This dimension defines the cost of the labor associated with operating the pipeline based on a quality threshold.

Graph Query Performance. Measured response time for specified SPARQL endpoint queries, including basic entity retrievals and multi-hop queries, during concurrent user simulations.

Interoperability Coverage. The percentage of processed records that are available and schema-valid through each delivery interface (SPARQL, REST, OAI-PMH, IIIF) and passing automated conformance tests.

System Reliability. Percent of failed service interruptions, average time between service interruptions for critical components, and average recovery time from critical component interruption.

Discussion

Scalability and infrastructure architecture

The proposed design employs horizontal expansion of stages using a containerized microservice architecture in order to facilitate independent scaling via Kubernetes. Stages that are both compute- and memory-intensive are designed to use GPUs to lower inference time, and GPU-node scaling is triggered when a queue reaches a defined threshold. Each of the processing stages derives its own performance benefits from batching, a practice that invites the grouping of (and inference of) multiple pages queued to the stage.

The trade-offs between on-premises and cloud-native architectures are pronounced for archival institutions. There are significant drops in the costs of managing the system within a cloud-native architecture; however, there are data sovereignty concerns when dealing with cloud providers such as AWS, GCP and Azure. Maintaining on-premises architecture using open-source tools allows data to remain on servers, while institutions lacking server infrastructure sacrifice horizontal expansion and elasticity. Hybrid systems may be the most

rational option for many institutions, in which on-premises footprints for both preservation and recognition are used. At the same time, the remaining semantic enrichment and delivery are cloud-based.

Smaller institutions with limited IT budgets cannot afford the full five-layer architecture. A more rudimentary system that implements layers L1, L2, and L5, providing high-quality OCR with delivery that meets standards, is a practical first step institutions can use to build their systems progressively.

Workflow orchestration and operational concerns

Pipeline failures, crashes in recognition models, network interruptions, and other issues must be handled without loss, duplication, or corruption of data provenance. Using idempotent task implementations that check previous outputs to avoid redundant tasks, along with Apache Airflow's at-least-once task execution, creates a better model to handle the aforementioned failures. Dead-letter queues for documents that exhaust the retry policy ensure that failures become visible to operators, rather than being silently dropped.

Particular attention to operations management for both the model version and registry management is a must. As recognition and NLP models improve, new versions of these models should be deployed without disrupting ongoing human reviews or invalidating what was processed. The hidden technical debt in production ML systems is a noted worry. The Model Registry pattern clearly addresses this concern by associating the processed documents with the model version used, enabling targeted re-processing when the model version improves.

Ethical and archival integrity considerations

The most immediate ethical risk of AI-driven archival processing appears to be the dissemination of recognitional and NLP errors as authoritative knowledge errors, damaging the scholarly record. The framework mitigates this risk with confidence-gated human review, machine-readable annotations that differentiate automatically generated and human-verified notes, and by preserving the original, unaltered copies of the archives alongside their derived representations.

It is difficult for AI to remain unbiased. AI models that are trained on digitized historical collections reflect the values of those collections. Historically, those collections have included the values of the majority social groups while neglecting those of minority social groups, including marginalized communities and languages (Bender et al., 2021; D'Ignazio & Klein, 2020). The framework mitigates this risk with diversity audits of the training data and NER performances across demographics, as well as through community stakeholder mechanisms for reporting misrepresentations.

When reviewing archival collections containing information about living individuals, the framework considers the privacy concerns arising from the GDPR and similar regulations. The framework integrates a privacy screening Layer 3 with suppression/pseudonymization rules. These are configured based on the entity's recency, the document's sensitivity, and rules defined within the jurisdiction. Long-term sustainability is specific to archival contexts where preservation obligations extend over decades. Technology-agnostic metadata standards are included in the framework to reduce dependence on specific software implementations, avoid explicit format migration at Layer 1, and separate preservation copies from derivative representations.

Limitations

Some constraints should be considered. First, and most importantly, the framework has not been empirically verified against a standardized corpus of archival records. Evaluation protocols describe target values taken from the literature. However, real-world performance may differ greatly in deployment, especially for low-resourced languages and more complex materials. The main outstanding work is an empirical validation of the framework against a multi-institutional benchmark corpus. Second, the graph federation pattern assumes a degree of collaboration and willingness to share data among institutions. Smaller repositories may

lack the technical capabilities to share their data, and even if they have data-sharing agreements, the legal and governance negotiations may be lengthy. The framework cannot address these non-technical challenges, which are the predominant barriers to the development of federated archival knowledge graphs. Third, the semantic enrichment pipeline performs poorly compared to inflected modern languages with a high degree of Natural Language Processing (NLP) resources. The framework is most applicable to the vernacular European languages for which annotated training data and pre-existing resources are available. Fourth, the framework still has gaps for low-resourced institutions, as the costs and requirements of the full five-layer architecture can be significant. The framework does not fully address the infrastructure and cost gaps. Fifth, the framework does not account for processual records, which have value because of the processes that involve their creation and use over time. These also present unique challenges for AI-based processing that go beyond those of document image analysis.

Future work

The most urgent future work consists of empirical validation against a standardized multi-institutional archival corpus stratified across material types (print, manuscript, photograph, map, born-digital), time periods (pre-1700, 1700–1850, 1850–1950, post-1950), languages (English, French, German, Latin, one non-European language), and institutional contexts.

A prototype implementation should be deployed and evaluated against the protocol, and the results should be presented in a companion empirical study. Research on low-resource historical language models for NLP subtasks in underrepresented languages is lacking and urgently needed. However, transfer learning approaches based on multilingual models (Conneau et al., 2020) are currently the most viable option.

The integration of LLM-assisted archival search and question-answering L5 is highly valuable for non-specialist users, though hallucination, source attribution, and confidence indication are needed to fulfill archival requirements (Lewis et al., 2020). LLMs grounded in a verified knowledge graph RAG are a promising direction. The ability to extend the graph federation pattern to cross-institutional, privacy-preserving federated learning (McMahan et al., 2017; Yang et al., 2019) would strategically address the dual challenges of resource scarcity and low-quality materials, as well as underdeveloped models for small institutions.

Framework application scenarios

The proposed framework can support large-scale census digitization initiatives by integrating document ingestion, OCR/HTR processing, semantic enrichment, and scalable publication workflows. The layered architecture enables efficient processing of complex census records while preserving provenance and metadata quality. Historical newspapers present challenges related to degraded scans, layout variability, and entity extraction. The framework supports these use cases through adaptive recognition pipelines, semantic enrichment components, and knowledge graph-based information representation. Large-scale city directories and genealogical collections require accurate entity resolution, metadata extraction, and search infrastructure. The proposed architecture provides a scalable approach for transforming such collections into searchable knowledge resources.

Conclusions

This paper develops a framework grounded in design science research, bringing a systems-level perspective to the AI-enabled transformation of historical archival data. The framework addresses a gap in the systems-level engineering required to deploy AI capabilities at a large institutional scale, even as AI advances at the component level.

The primary contribution of this study is the development of a systems-level reference architecture and associated design patterns for AI-driven transformation of historical data at

scale. The framework integrates preservation, recognition, semantic enrichment, knowledge representation, and delivery mechanisms into a unified architectural model suitable for modern archival ecosystems.

A five-layer reference architecture, Ingestion and Preservation, Recognition and Extraction, Semantic Enrichment, Knowledge Representation, and Orchestration and Delivery, delineates the processing challenge. It describes the inputs and outputs, the core technologies, failure modes, quality control, and human-in-the-loop checkpoints and provenance at each layer. The five design patterns Ingestion Gateway, Adaptive Recognition, Semantic Enrichment Chain, Graph Federation, and Orchestrated Provenance are representative of architectural solutions for archival challenges, which provide a basis for practitioners to design and implement solutions for a variety of problems.

The structured evaluation protocol defines nine criteria with delineated target values. This provides a baseline for future research and informs practitioners of the extent to which their designs can be realized. The metadata standards integration framework provides an architecture upon which systems may join the international digital cultural heritage infrastructures.

The intersection of AI and archival needs provides an unprecedented opportunity to unlock records that were, until now, inaccessible to scholarly, civic and cultural value for decades and even centuries. This framework lays the foundation in system engineering for realizing this opportunity.

References

1. *Bender, E. M., Gebru, T., McMillan-Major, A., & Shmitchell, S. (2021). On the dangers of stochastic parrots: Can language models be too big? In Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency (pp. 610–623). ACM.* https://doi.org/10.1145/3442188.3445922
2. *Bollmann, M. (2019). A large-scale comparison of historical text normalization systems. In Proceedings of NAACL-HLT (pp. 3885–3898). Association for Computational Linguistics.* https://doi.org/10.18653/v1/N19-1389
3. *Burns, B., Grant, J., Oppenheimer, D., Brewer, E., & Wilkes, J. (2022). Borg, Omega, and Kubernetes. Communications of the ACM, 59(5), 50–57.* https://doi.org/10.1145/2898442
4. *Clausner, C., Antonacopoulos, A., & Pletschacher, S. (2019). ICDAR2019 competition on recognition of documents with complex layouts. In Proceedings of ICDAR 2019 (pp. 1521–1526). IEEE.* https://doi.org/10.1109/ICDAR.2019.00245
5. *Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., Grave, E., Ott, M., Zettlemoyer, L., & Stoyanov, V. (2020). Unsupervised cross-lingual representation learning at scale. In Proceedings of ACL 2020 (pp. 8440–8451). Association for Computational Linguistics.* https://doi.org/10.18653/v1/2020.acl-main.747
6. *Constum, T., Kempf, N., Paquet, T., Tranouez, P., Chatelain, C., Brée, S., & Merveille, F. (2022). Recognition and information extraction in historical handwritten tables: Toward understanding early 20th-century Paris census. Lecture Notes in Computer Science, 13237, 143–157.* https://doi.org/10.1007/978-3-031-06555-2_10
7. *Cramer, T. (2011). The International Image Interoperability Framework (IIIF): Laying the foundation for common services, integrated resources and a marketplace of tools for scholars worldwide. In Proceedings of the Charleston Library Conference.* https://doi.org/10.5703/1288284315901
8. *D'Ignazio, C., & Klein, L. F. (2020). Data feminism. MIT Press.* https://doi.org/10.7551/mitpress/11805.001.0001

9. *Doerr, M. (2003). The CIDOC conceptual reference module: An ontological approach to semantic interoperability of metadata. AI Magazine, 24(3), 75–92.* https://doi.org/10.1609/aimag.v24i3.1720
10. *Doerr, M., & Hagedorn-Saupe, M. (2020). Harmonizing heritage collections using CIDOC CRM. In M. Lamber & B. Sanderhoff (Eds.), Sharing is caring: Openness and sharing in the cultural heritage sector. National Gallery of Denmark.*
11. *Ehrmann, M., Romanello, M., Flückiger, A., & Clematide, S. (2020). Extended overview of CLEF HIPE 2020: Named entity processing on historical newspapers. In CLEF 2020 Working Notes. CEUR-WS.* https://doi.org/10.5281/zenodo.4117566
12. *Ehrmann, M., Romanello, M., Bircher, S., & Clematide, S. (2022). HIPE-2022: Named entity recognition and linking in multilingual historical documents. In Proceedings of LREC 2022 (pp. 5756–5768). European Language Resources Association.*
13. *Fehling, C., Leymann, F., Retter, R., Schupeck, W., & Arbitter, P. (2014). Cloud computing patterns: Fundamentals to design, build, and manage cloud applications. Springer.* https://doi.org/10.1007/978-3-7091-1568-8
14. *Fischer, A., Kaden, M., Märgner, V., & Bunke, H. (2012). Ground truth creation for historical handwritten documents. In Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (pp. 351–355). IEEE.* https://doi.org/10.1109/DAS.2012.90
15. *Frey, J., Müller-Birn, C., Leinberger, M., Lohmann, S., & Doehring, L. (2017). RDF delta: Change sets and patches for RDF datasets. In Proceedings of the 16th International Semantic Web Conference Workshops. ISWC 2017.*
16. *Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1994). Design patterns: Elements of reusable object-oriented software. Addison-Wesley.*
17. *Graves, A., & Schmidhuber, J. (2009). Offline handwriting recognition with multidimensional recurrent neural networks. In Advances in Neural Information Processing Systems 21 (NIPS 2008) (pp. 545–552). Curran Associates.*
18. *Grover, C., Tobin, R., Byrne, K., Woollard, M., Reid, J., Dunn, S., & Ball, J. (2012). Use of the Edinburgh geoparser for georeferencing digitized historical collections. Philosophical Transactions of the Royal Society A, 368(1925), 3875–3889.* https://doi.org/10.1098/rsta.2010.0149
19. *Harenslak, B., & de Ruiter, J. (2021). Data pipelines with Apache Airflow. Manning Publications.*
20. *Harrower, N., Maryl, M., Biro, T., & Immenhauser, B. (Eds.). (2020). Scholarly primitives revisited: Towards a practical typology of digital humanities research activities and objects. DARIAH-EU.*
21. *Honnibal, M., Montani, I., Van Landeghem, S., & Boyd, A. (2020). spaCy: Industrial-strength natural language processing in Python. Zenodo.* https://doi.org/10.5281/zenodo.1212303
22. *Hyvönen, E. (2012). Publishing and using cultural heritage linked data on the semantic web. Morgan & Claypool.* https://doi.org/10.2200/S00452ED1V01Y201210WBE003
23. *Hyvönen, E. (2020). Using the semantic web in digital humanities: Shift from data publishing to data-analysis and serendipitous knowledge discovery. Semantic Web, 11(1), 187–193.* https://doi.org/10.3233/SW-190386
24. *Isaac, A., & Haslhofer, B. (2013). Europeana linked open data: Data.europeana.eu. Semantic Web, 4(3), 291–297.* https://doi.org/10.3233/SW-120092
25. *Jaillant, L., & Zhao, L. (2025). Introduction: When data turns into archives — Making digital records more accessible with AI. AI & Society, 40, 5787–5791.* https://doi.org/10.1007/s00146-025-02374-y
26. *Kang, L., Riba, P., Rusiñol, M., Fornés, A., & Villegas, M. (2022). Pay attention to what you read: Non-recurrent handwritten text-line recognition. Pattern Recognition, 129, 108766.* https://doi.org/10.1016/j.patcog.2022.108766

27. *Kleppmann, M. (2017). Designing data-intensive applications: The big ideas behind reliable, scalable, and maintainable systems. O'Reilly Media.*
28. *Lebo, T., Sahoo, S., & McGuinness, D. (Eds.). (2013). PROV-O: The PROV ontology (W3C Recommendation). World Wide Web Consortium.* https://www.w3.org/TR/prov-o/
29. *Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In Advances in Neural Information Processing Systems 33 (NeurIPS 2020) (pp. 9459–9474). Curran Associates.*
30. *Li, M., Lv, T., Chen, J., Cui, L., Lu, Y., Florencio, D., Zhang, C., Li, Z., & Wei, F. (2021). TrOCR: Transformer-based optical character recognition with pre-trained models. arXiv preprint arXiv:2109.10282.* https://doi.org/10.48550/arXiv.2109.10282
31. *Lombardi, F., & Marinai, S. (2020). Deep learning for historical document analysis and recognition – A survey. Journal of Imaging, 6(10), 110.* https://doi.org/10.3390/jimaging6100110
32. *McMahan, B., Moore, E., Ramage, D., Hampson, S., & Aguera y Arcas, B. (2017). Communication-efficient learning of deep networks from decentralized data. In Proceedings of AISTATS 2017 (pp. 1273–1282). PMLR.*
33. *Meng, K., Bau, D., Andonian, A., & Belinkov, Y. (2022). Locating and editing factual associations in GPT. In Advances in Neural Information Processing Systems 35 (NeurIPS 2022) (pp. 17359–17372). Curran Associates.*
34. *Momtaz, Y., Laccetti, L., & Russo, G. (2025). Modular pipeline for text recognition in early printed books using Kraken and ByT5. Electronics, 14(15), 3083.* https://doi.org/10.3390/electronics14153083
35. *Mons, B., Neylon, C., Velterop, J., Dumontier, M., da Silva Santos, L. O. B., & Wilkinson, M. D. (2017). Cloudy, increasingly FAIR; revisiting the FAIR data guiding principles for the European Open Science Cloud. Information Services & Use, 37(1), 49–56. https://doi.org/10.3233/ISU-170824*
36. *Muehlberger, G., Seaward, L., Terras, M., Ares Oliveira, S., Bosch, V., Bryan, M., Colutto, S., Déjean, H., Diem, M., Fiel, S., Gatos, B., Greinoecker, A., Grüning, T., Hackl, G., Haukkovaara, V., Heyer, G., Hirvonen, L., Hodel, T., Jokinen, M., … Zagoris, K. (2019). Transforming scholarship in the archives through handwritten text recognition: Transkribus as a case study. Journal of Documentation, 75(5), 954–976.* https://doi.org/10.1108/JD-07-2018-0114
37. *Noy, N., Gao, Y., Jain, A., Narayanan, A., Patterson, A., & Taylor, J. (2019). Industry-scale knowledge graphs: Lessons and challenges. Queue, 17(2), 48–75.* https://doi.org/10.1145/3329781.3332266
38. *Opgenhaffen, L. (2022). Archives in action: The impact of digital technology on archaeological recording strategies and ensuing open research archives. Digital Applications in Archaeology and Cultural Heritage, 27, e00231.* https://doi.org/10.1016/j.daach.2022.e00231
39. *Park, J.-R., & Tosaka, Y. (2010). Metadata quality control in digital repositories and collections: Criteria, semantics, and mechanisms. Cataloging & Classification Quarterly, 48(8), 696–715. https://doi.org/10.1080/01639374.2010.508711*
40. *Puigcerver, J. (2017). Are multidimensional recurrent layers really necessary for handwritten text recognition? In Proceedings of ICDAR 2017 (pp. 67–72). IEEE.* https://doi.org/10.1109/ICDAR.2017.20
41. *Qi, P., Zhang, Y., Zhang, Y., Bolton, J., & Manning, C. D. (2020). Stanza: A Python natural language processing toolkit for many human languages. In Proceedings of ACL 2020: System Demonstrations (pp. 101–108). Association for Computational Linguistics.* https://doi.org/10.18653/v1/2020.acl-demos.14
42. *Reis, J., & Housley, M. (2022). Fundamentals of data engineering: Plan and build robust data systems. O'Reilly Media.*

43. *Sculley, D., Holt, G., Golovin, D., Davydov, E., Phillips, T., Ebner, D., Chaudhary, V., Young, M., Crespo, J.-F., & Dennison, D. (2015). Hidden technical debt in machine learning systems. In Advances in Neural Information Processing Systems 28 (NIPS 2015) (pp. 2503–2511). Curran Associates.*
44. *Seuret, M., Nicolaou, A., Stutzmann, D., Marti, U.-V., & Liwicki, M. (2021). ICDAR 2021 competition on historical document classification. In Proceedings of ICDAR 2021 (pp. 618–634). Springer.* https://doi.org/10.1007/978-3-030-86337-1_41
45. *Shen, Z., Zhang, R., Dell, M., Lee, B. C. G., Carlson, J., & Li, W. (2021). LayoutParser: A unified toolkit for deep learning based document image analysis. In Proceedings of ICDAR 2021 (pp. 131–146). Springer.* https://doi.org/10.1007/978-3-030-86549-8_9
46. *Snydman, S., Sanderson, R., & Cramer, T. (2015). The International Image Interoperability Framework (IIIF): A community and technology approach for web-based images. In Proceedings of the Archiving 2015 Conference (pp. 16–21). Society for Imaging Science and Technology.*
47. *Strötgen, J., & Gertz, M. (2013). Multilingual and cross-domain temporal tagging. Language Resources and Evaluation, 47(1), 269–298.* https://doi.org/10.1007/s10579-012-9179-y
48. *Van Hooland, S., De Wilde, M., Verborgh, R., Steiner, T., & Van de Walle, R. (2013). Exploring entity recognition and disambiguation for cultural heritage collections. Digital Scholarship in the Humanities, 30(2), 262–279.* https://doi.org/10.1093/llc/fqt016
49. *Wigington, C., Stewart, S., Newman, B., Price, B., Hampton, S., & Cohen, S. (2017). Data augmentation for recognition of handwritten words and lines using a CNN-LSTM network. In Proceedings of ICDAR 2017 (pp. 639–645). IEEE.* https://doi.org/10.1109/ICDAR.2017.107
50. *Wilkinson, M. D., Dumontier, M., Aalbersberg, I. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., da Silva Santos, L. B., Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., … Mons, B. (2016). The FAIR guiding principles for scientific data management and stewardship. Scientific Data, 3, 160018.* https://doi.org/10.1038/sdata.2016.18
51. *Yang, Q., Liu, Y., Chen, T., & Tong, Y. (2019). Federated machine learning: Concept and applications. ACM Transactions on Intelligent Systems and Technology, 10(2), 1–19.* https://doi.org/10.1145/3298981