ML-Based Classification of Early Modern and Medieval Libraries

Abubakr Usman, Kritik Jain, Cindy Liu

July 2025

Abstract

The ability to access and process what is left of historical academia is invaluable to the study of bygone societies. However, manually reading through large volumes of historical catalogs written in multiple languages requires not only language proficiency, but an impractical amount of dedication and time. In this paper, we propose an ML-based approach to extract structured genre-specific data from such catalogs. This approach extracts names of libraries, the works they hold, and the likely primary literature category in their collections based on pre-processed text. Using BERT and GPT-4o, our model scores an overall accuracy of 87% for identifying genres of books mentioned in texts. This model can replace manual record-keeping with an automated version.

1 Introduction

One good source historians have for the European past is knowledge of the contents of library collections what books were held by various institutions where people came to study. These collections, from monasteries, cathedral schools, universities, and government bureaucracies, reflect the scholarship of their time. The books held by a large research institution, which attracted eager students to it and where they were trained by reading the books held there, can tell us what they eventually learned. This project aims to use cutting-edge AI and Machine Learning techniques to describe, in broad terms, the contents of several of these collections based on AI "reading" many hundreds of titles.

Given a dataset of catalogs in Latin, German, French, and English, uploaded in various file types (PDF, TIFF, and XLSX), our task was to use AI to identify historically relevant information about 14th to 18th century European libraries. This meant processing each file to extract data concerning individual libraries and the works they housed, including the timeline of the libraries, the main categories of works in their collections, and basic information about each work (title, author, year collected, year lost, and genre).

2 Related Works

LLMs have been dependable for projects extracting information from large quantities of data. The introduction of self-attention mechanisms made LLMs effective at recognizing long-range dependencies, ideal for large bodies of text. [Vaswani *et al.*(2017)] Recently, open source Llama 2 models were used to find English text-based historical records of orphaned wells with an accuracy of 100%. [Ma *et al.*(2024)]

Though not dissimilar to other existing text analysis projects in concept or in structure, our work is tailored toward identification of books and analysis of libraries with minimal cost. This project is intended to use machine learning to create a simple solution to a very specific problem; it aims for convenience, volume, and accuracy when used within this niche.

3 Methodology

3.1 Overview

The proposed pipeline begins with the extraction and preprocessing of raw data sourced from PDFs, spreadsheets, and image scans. This raw text is then cleaned and normalized to correct OCR errors, archaic spellings, and formatting noise. A multilingual LLM (GPT-4o) is used in a few-shot prompting setup to extract structured fields from these entries including book title, author, catalog number, and time period. This resulted in the conversion of unstructured text into structured tabular form. This structured data is passed to a fine-tuned BERT model trained to classify the likely genre of each book entry based on contextual clues in the other fields. The architecture of this methodology is depicted in Figure 1.

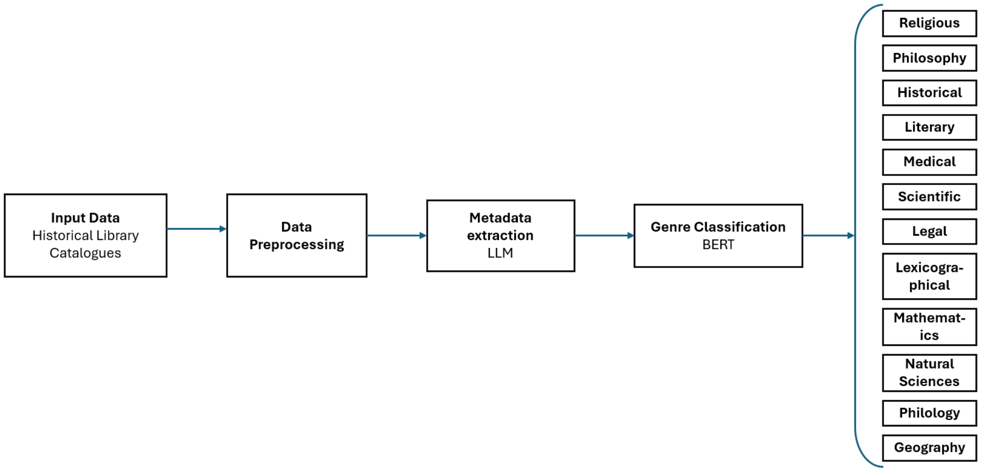


Figure 1: Architecture Diagram.

3.2 Preprocessing Pipeline for Historical Catalog Texts

The first stage of this proposed pipeline involved constructing a robust preprocessing pipeline to handle the digitized catalogs of early modern and medieval European libraries, which were in multiple languages including Latin, German, English and French. The source files comprised of scanned PDF documents and spreadsheets, marked by low-quality scans and archaic print formats. These documents include catalog entries ranging from short book titles to embedded references within text, making the extraction of information a non-trivial task.

3.2.1 Text Extraction

As illustrated in Figure 2, the pipeline begins with a text extraction phase, wherein raw PDF files are processed using optical character recognition (OCR) and parsing tools (e.g. Tesseract). For spreadsheets, text was extracted using Pandas with regex-based column filtering. This step produces unstructured raw text data that is frequently marred by OCR artifacts, inconsistent line breaks, overlapping headers/footers, and obsolete character glyphs. The output is a stream of unstructured strings that may contain errors due to archaic typefaces, marginal notes, and degraded page quality.

3.2.2 Text Cleaning

Text cleaning is an essential phase in any NLP task, which aims to modify data in a format that is much easier for the algorithm to analyze or predict. In this phase, the raw text is then processed through a modular text cleaning pipeline, designed to standardize the corpus and mitigate common noise factors. This pipeline consists of three main components:

Artifact Removal: This module detects and strips repetitive structural metadata such as page numbers, section headers, catchwords, and column labels. These elements are commonly introduced during OCR and can disrupt downstream text segmentation and linguistic analysis.

Character Encoding Correction: This step resolves encoding mismatches and normalizes special or non-ASCII characters often introduced by OCR misreads.

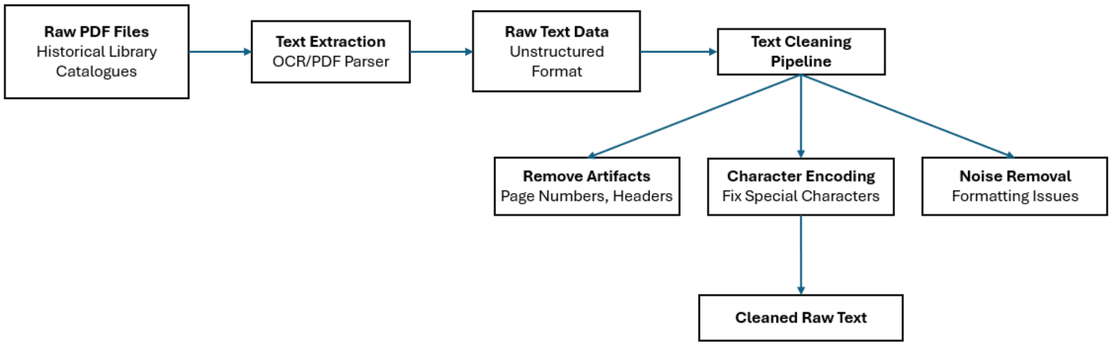


Figure 2: Preprocessing Books Data

Noise and Formatting Cleanup: This module targets line-break inconsistencies, excessive whitespace, and layout artifacts. In catalogs with paragraph-like prose (e.g. narrative cataloging), sentence segmentation is heuristically restored to enable further natural language processing (NLP).

The result of this pipeline is a clean and minimally structured text corpus suitable for computational linguistic analysis.

3.2.3 Structured Information Extraction

To extract semantically meaningful information from short and often noisy data, we employed a multilingual Large Language Model (LLM), specifically GPT-4o[1], using a few shot prompt-based extraction framework. The stage resulted in structured data records under standardized fields as shown in Figure 3.

3.2.4 Prompt Design and Metadata Schema

We provided GPT with a system prompt that included examples (few-shot learning) of how to extract information under the following format:

```
{
    "BookName": "Title of book, where present or inferable",
    "AuthorName": "Author’s name when stated",
    "BookNo": "Number of books under this heading (if Author or Genre referenced only)",
    "TimePeriod": "Approximate time period, where present or inferable",
    "Library": "Name of library"
    "Location": "Location associated with book, where present."
    "Genre": "Genre of book",
    "Description": "Short sentence describing any other relevant information."
  }
```

3.2.5 Input Chunking Strategy

Given GPT’s token limit and the fragmented nature of the texts, input was processed in batches of six at a time. For each block we extracted from the clean raw text, GPT was prompted using custom instruction designed to identify and return available fields only. Each output was parsed into a structured JSON record and appended to the cumulative dataset. Missing fields were permitted in the results to allow partially useful outputs.

[1] We tested Llama 3.2 8Bn parameter model for text extraction, but the quality of the result was not optimum for downstream classification task

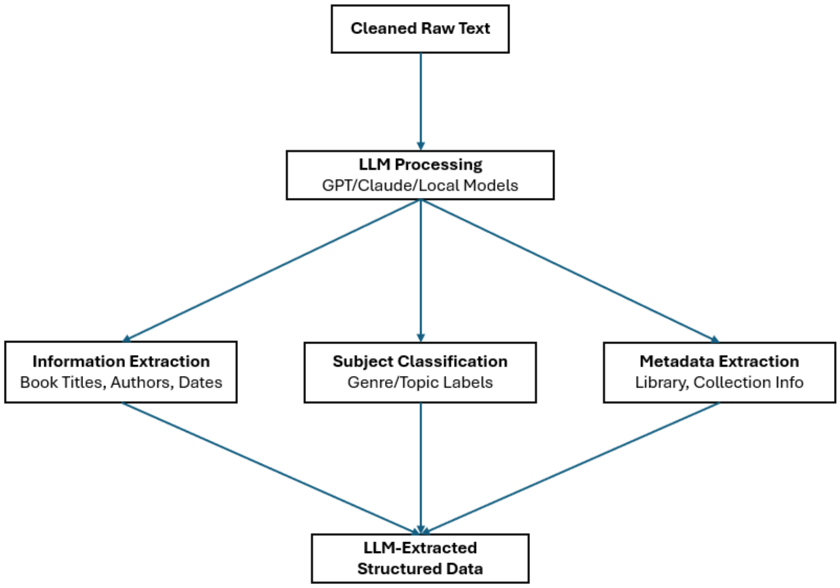


Figure 3: Extraction of Structured Dataset

3.2.6 Output Format

The final output of this stage was a structured dataset in tabular form, with one row per catalog entry and columns matching the above format.

3.3 Genre Classification using BERT

After extracting the structured information using LLM, the resulting tabular dataset was used as input to a supervised transformer-based classifier to predict the Genre for entries where it was missing or ambiguous.

We finetuned pre-trained BERT-base-cased-multilingual model to address the book genre classification task. BERT-multilingual is a variant of the BERT-base model that supports over 100 languages, making it well-suited for our multilingual book corpus, which includes Latin, German, English, and French texts from diverse library collections. BERT-multilingual is a lightweight enough to be trained with modest resources while retaining strong multilingual contextual understanding (Figure 4). This makes it ideal for classifying books into genre categories using combined textual features (book names, library names, and descriptions) across diverse linguistic sources.

4 Results and Discussion

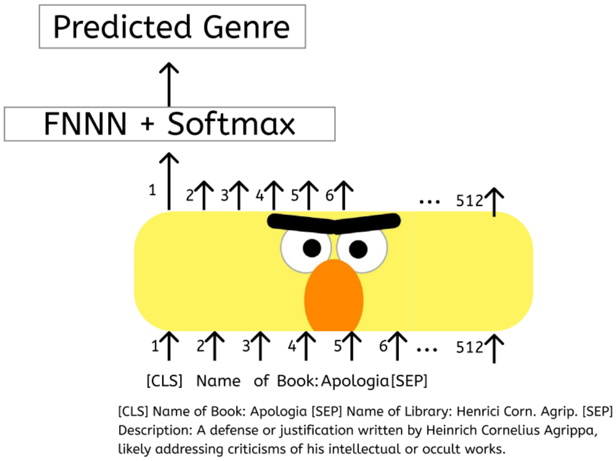
4.1 Dataset

In proposed model, we employed advanced LLM based pipeline to generate a dataset to train a classifier, validate the model and test it. Details about the produced dataset can be seen in Figure 5 and 6.

4.2 Experiments

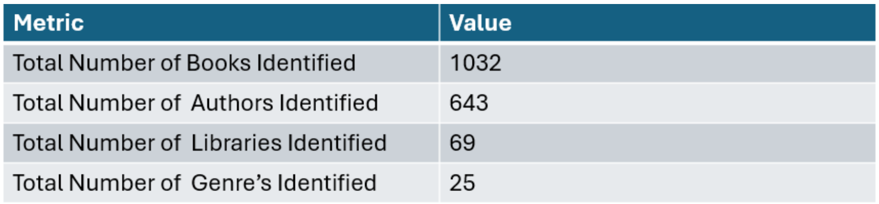
4.2.1 Experimental Setup

We used pre-trained bert-base-multilingual-cased model for the genre classification task. This model consists of 12 transformer layers, each with 12 self attention heads and a hidden size of 768 units, totaling approximately 110 million parameters. The maximum sequence length was set to 512 tokens.



[CLS] Name of Book: Apologia [SEP] Name of Library: Henrici Corn. Agrip. [SEP]
Description: A defense or justification written by Heinrich Cornelius Agrippa, likely addressing criticisms of his intellectual or occult works.

Figure 4: BERT Architecture



Metric	Value
Total Number of Books Identified	1032
Total Number of Authors Identified	643
Total Number of Libraries Identified	69
Total Number of Genre's Identified	25

Figure 5: Table describing Dataset

Training was conducted for 3 epochs using a batch size of 8 for both training and evaluation. We employed the AdamW optimizer with a learning rate of 5e-5, 500 warm-up steps, and a weight decay of 0.01. Early stopping was enabled with a patience of 3 epochs. Training was performed on T4-GPU with 16GB RAM.

Preprocessing involved concatenating the BookName, Library, and Descriptions fields using [SEP] tokens. Missing fields were filled with empty strings, and entries with no contextual evidence were removed. To address class imbalance, we focused on the top 5 most common genres. Text inputs were tokenized and using the BERT multilingual tokenizer with truncation and padding enabled.

The dataset was split into 80% training and 20% testing. Additionally 10% of the training set was used as validation. Stratified sampling was applied to preserve genre distributions with a random seed of 42 for reproducibility.

4.2.2 Evaluation Metrics

Evaluation was performed every 500 steps during training, using accuracy as the primary metric. The best model was selected based on validation accuracy. Final evaluation included a confusion matrix, per-class accuracy and a full classification report.

To further analyze model performance beyond overall accuracy, we incorporated precision, recall, and F1-score into our final evaluation. These metrics provide a more nuanced understanding of how well the model performs for each genre, especially in the presence of class imbalance.

Precision helps assess how many predicted instances of a genre were actually correct, while recall measures how well the model retrieves all actual instances of a genre. The F1-score balances the two, offering a single value that reflects both precision and recall.

1. **Precision (Positive Predictive Value)**

$$\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{1}$$

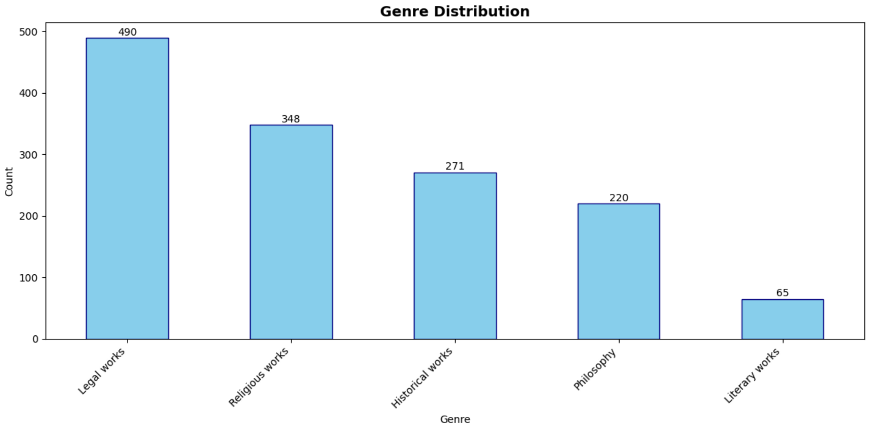


Figure 6: Genre Distribution of Dataset

2. **Recall (Sensitivity / True Positive Rate)**

$$\text{Recall}_i = \frac{TP_i}{TP_i + FN_i} \tag{2}$$

3. **F1-Score (Harmonic Mean of Precision and Recall)**

$$F1_i = 2 \times \frac{\text{Precision}_i \times \text{Recall}_i}{\text{Precision}_i + \text{Recall}_i} \tag{3}$$

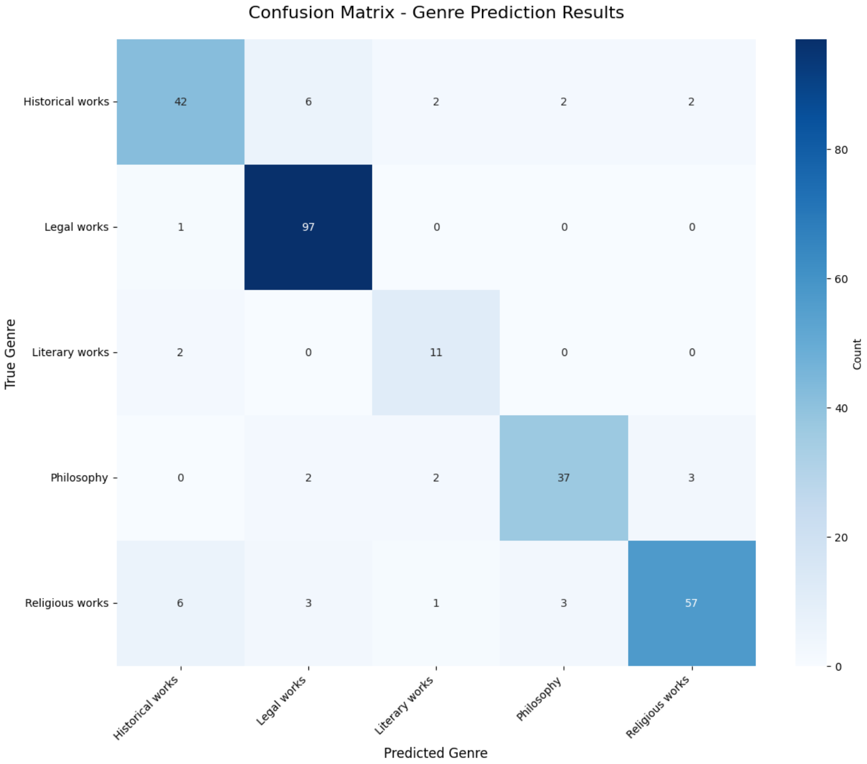
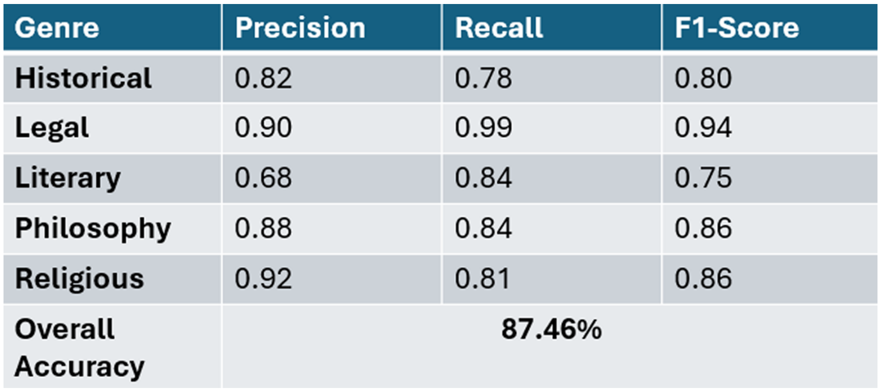


Figure 7: Confusion Matrix

4.3 Results

The genre classification model demonstrated strong overall performance, particularly for high-support genres. As shown in the confusion matrix (Figure 7) and accuracy table (8, the model achieved the highest per-class accuracy on Legal works, with a remarkable 99.0% accuracy. Other genres also showed promising results, with Literary works at 84.6%, Philosophy at 84.1%, and Religious works at 81.4% accuracy.

The lowest accuracy was observed in Historical works, with 77.8%. Misclassifications in this category were primarily into Legal works and minor confusion with other genres. This is likely due to content overlap between historical and legal documents in the corpus. The confusion matrix also highlights that most misclassifications occurred among semantically adjacent categories (e.g., Philosophy vs. Religious works), indicating that the model is sensitive to genre similarities.



Genre	Precision	Recall	F1-Score
Historical	0.82	0.78	0.80
Legal	0.90	0.99	0.94
Literary	0.68	0.84	0.75
Philosophy	0.88	0.84	0.86
Religious	0.92	0.81	0.86
Overall Accuracy	**87.46%**		

Figure 8: Accuracy Table

Overall, these results suggest that the model is effective at identifying distinct genres when sufficient contextual and lexical signals are present.

5 Conclusion

In this paper, we developed an end-to-end machine learning-based pipeline to classify books from early modern and medieval European library catalogs into primary literary genres. By combining Tesseract to extract text, GPT-4o for metadata extraction, and a fine-tuned multilingual BERT model for genre classification, we transform ancient unstructured catalogs into structured and analyzable data. The model achieved an overall accuracy of 87.46%, with a particularly strong performance on legal, philosophical, and religious works. Our findings thus show that transformer-based architectures can effectively process noisy historical texts and extract meaningful insights. Our method significantly reduces the manual effort required for analyzing historical catalogs.

Future work will focus on increasing representation of underrepresented genres through historical expansion sampling, as well as exploring more advanced LLM usage to further improve classification of rarer entries in the catalogs.

6 Project Breakdown

The work was divided as follows:

- Data Preprocessing: Abubakr Usman, Cindy Liu
- BERT classification model: Abubakr Usman
- GPT-4o model fine-tuning:Abubakr Usman
- Structured data extraction: Abubakr Usman

- Llama model testing: Abubakr Usman, Kritik Jain
- Paper: Abubakr Usman, Cindy Liu, Kritik Jain
- Poster: Abubakr Usman, Cindy Liu, Kritik Jain

Acknowledgments

We are grateful to Professor Colin Wilder for supplying the original data and helping us define our task framework, and Haripriya and Bhagirath Mehta from MehtA+ Tutoring for guiding us through this process.

References

[Ma *et al.*(2024)] Zhiwei Ma, Javier E. Santos, Greg Lackey, Hari Viswanathan, and Daniel O'Malley. "Information Extraction from Historical Well Records Using a Large Language Model." *Scientific Reports*, vol. 14, article 81846, 2024. `https://doi.org/10.1038/s41598-024-81846-5`

[Vaswani *et al.*(2017)] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. "Attention Is All You Need." In *Advances in Neural Information Processing Systems*, vol. 30, 2017. `https://arxiv.org/abs/1706.03762`