CODEX COGNITIO: INTELLIGENT READING OF RENAISSANCE TEXTS

Haripriya Tolety
MehtA+

Sammy Lei
MehtA+

Kathleen Lin
MehtA+

Anirudh Bandaru
MehtA+

July 31, 2025

ABSTRACT

This study uses several different models, including Tesseract, Transkribus, and LLMs for transcribing handwritten texts. The given manuscripts date back to the Renaissance in 15th-century Italy. We preprocessed the images to improve clarity and utilized some of them to train existing models. Then, more manuscripts were tested on the models mentioned above. Then, we used a preprocessing method, and the transcription accuracy was compared to that of the original.

1 Introduction

In this study, we were provided with 123 manuscripts from the University of Pennsylvania Schoenberg Institute for Manuscript Studies. The goal was to use machine learning to transcribe the manuscripts. These documents were specifically from 15th-century Italy, an era known as the Italian Renaissance. The Renaissance marked a major cultural shift in society. During that time, people began prioritizing the humanities, art, and religion, and numerous advancements were made in various fields, including literature, science, and philosophy. Understanding what was written back then gives us a better understanding of how people's mindsets were changing and the discoveries they made. Instead of going in manually and translating each word, we used machine learning to transcribe the handwritten text for us quickly and efficiently. This study employed several models to transcribe the data, including Tesseract (an OCR), Transkribus, and LLMs. Furthermore, a preprocessing method was used to augment the accuracy of using AI for transcription.

2 Related Work

Handwritten text recognition for ancient texts is an ever-expanding field, connecting modern machine learning techniques with historical linguistics. Researchers at the University of Pennsylvania recently delved deeper and created a deep learning model on South Asian texts. Utilizing Optical Character Recognition (OCR) and Handwritten Text Recognition (HTR) tools, they successfully parsed through various South Asian languages. Their initiative aimed to expand cultural diversity in HTR, addressing the complexities of cultural heritage that conventional technical methods cannot capture.

Through our project, we will add to the field of Handwritten Text Recognition in the Latin language via Transkribus, LLMs, Tesseract, and a method to increase transcription accuracy. We also present our fine-tuned Transkribus model to identify handwritten text in ancient Latin manuscripts, with a final accuracy of approximately 79.42%. Unfortunately, our Tesseract model didn't turn out as well as hoped, although it can still print out transcribed texts.

3 Methodology

3.1 Dataset

For our dataset, we were provided with 123 manuscripts by the University of Pennsylvania's Schoenberg Institute for Manuscript Studies. Each of the documents contained approximately 150-250 pages. That alone is around 24,600 pages of handwritten text. Our team decided to focus on the single column text with straight blocky letters for this project because it is more legible. We also chose only to use 40-50 pages from multiple documents for our dataset. This decision significantly reduces the memory used and fits the given time constraint. When discussing the machine learning aspect of the project, the data we received was largely unlabeled. On top of that, these manuscripts do not have manually transcribed versions that the machine learning model can use to train on.

3.2 Preprocessing

Preprocessing is vital because it ensures the data is formatted so the model can understand and process it. Preprocessing data for OCR involves transforming the image to make it as clear as possible so the model can accurately transcribe the text from the image to text. The preprocessing used in the study began with a function that resized and scaled the images. This function removed excess margins and resized the image to a more convenient size. This function also changed the pixels per inch to over 300, which is an optimal value for the model. Next, the second part of preprocessing consists of two more functions. The first function removed the patches of high intensity within the image, which smoothed out the image and made it easier for the model to work with. The second function changed all the words to black and the background to white. Great contrast between the text and the non-text makes it easier for the model to read and interpret the writing. In summary, we preprocessed the data by transforming it into a distinct format, which helped us throughout the rest of the experiment.

In order to test whether preprocessing increases the accuracy of the transcriptions, we used a singular part of a manuscript to have both an original and preprocessed version, then plugged both into an LLM. After getting the transcriptions, we compared both versions to the original transcription. The differences are highlighted in red:

Correct Transcription:

ETsi ego mi Nicolae prius quoq(ue) uehementer amabam Platonem tuum sic (e)n(im) mihi placet appellare illum pro quo tu aduersus indoctorum turbam omni tempore pugnauisti tn ex quo hunc dialogum in latinum facere incepi tantus mihi beniuolentię cumulus accrescit.´ ut nunc demum amare ip(su)m uidear prius autem solumodo dilexisse. Caue n existimes qcq oio reperiri posse quod aut sapientius scriptum sit.´ aut facundius.´ Quod quidem ego nc magis q antea intelligo. Cum ob conuertendi diligentiam singula eius uiri dicta conteram atq olfaciam necesse est. Maximas igitur colucio patri et preceptori meo gratias ago. qui iniungendo hoc munere tantum mihi beneficium attulit. Prius n duntaxat Platonem uideram nunc et ut mihi uideor cognoui. x‘Cuius ego libros si aliquando absoluero et lati-

Original Transcription:

Diligo mi Nicolae, prius quoque uehementer amabam Platonem tuum hunc, mihi placet appellare illum pro quo tu aduersus indoctorum turbam omni tempore pugnauisti. Tamen ex quo hunc dialogum in latinum facere incepi, tantus mihi beneuolentię cumulus accreuit, ut nunc demum amare ipsum uidear, prius autem plurimum dilexisse. Caue n existimes, quod q oio[1] reperiri posse quod aut sapientius scriptum sit aut facundius. Quod quidem ego nc magis q antea intelligo. Cum ob conuertendi diligentiam singula eius uiri dicta conteram atq olfaciam necesse est. Maximas igitur Colucio[2] patri & preceptori meo gratias ago, qui iniungendo hoc munere tantum mihi beneficium attulit. Prius n duntaxat Platonem uideram, nunc et ut mihi uideor cognoui. Cuius ego libros si aliquando absoluero et lati...

Preprocessed Transcription:

[S][1] Diligo mi Nicolae prius quoque uehementer amabam Platonem tuum sic.n.[2] mihi plac& (placet) appellare illum pro quo tu aduersus indoctorum turbam omni tempore pugnauisti. tn (tamen) ex quo hunc dialogum in latinum facere incepi tantus mihi beneuolentię cumulus accreuit, ut nunc demum amare ipm (ipsum) uidear prius autem plurimodo[3] dilexisse. Caue.n. (enim) existimes qd (quod) q(uia) oio[4] reperiri posse quod aut sapientius scriptum sit. aut facundius. Quod quidem ego nūc (nunc) magis ql (quam) antea intelligo. Cum ob conuertendi diligentiam singula eius uiri dicta conteram atq(ezh tail) (atque) olfaciam necesse est. Maximas igitur colucio patri et preceptori meo gratias ago. qui iniungendo hoc munere tantum mihi beneficium attulit. Prius .n. (enim) duntaxat platonem uideram nunc et ut mihi uideor cognoui. Cuius ego libros si aliquando absoluero & lati...

3.3 Model

LLMs A Large Language Model (LLM) like ChatGPT or Gemini uses zero-shot or few-shot learning. This is useful for our project because we have no labeled dataset. We wanted to test if the LLM would give better results if given preprocessed data versus the data in its original format. ChatGPT was our choice for a Large Language Model. We already had the transcribed version of the document. After testing ChatGPT on the original and preprocessed data, we observed several differences within the text. Both the original data and the preprocessed data were very accurate. But overall, the preprocessed data works better because a few extra words were closer to the correct transcribed version.

While testing this on other manuscripts, we see a similar pattern. However, it is difficult to know for certain because the data is unlabeled.

Transkribus Transkribus is a platform that provides many models that can be used or trained to make a viable handwritten text recognition model. Due to our time and material constraints, one of our methods would be to fine-tune an existing model of Transkribus to a humanistic script. We fed the model Transkribus Print M1 37 pages from the documents Lewis E 171, Lewis E 150, Lewis E 114, and MS 53.

Due to our limited validation data (transcriptions of the original documents themselves), we had to use an LLM to generate transcriptions of a few documents due to our limited knowledge of humanistic script. Then, we used an integrated Transkribus AI to generate regioning and lining, securing the format of our transcription to be equal to the original format of the page. Then, we copied the transcription into Transkribus, which fits uniformly into the formatting.

In the preprocessing of this model, Transkribus automatically transformed all of the documents to high contrast (black and white), meaning that no pre-processing work was needed. Then, it split the documents into a 9:1 ratio as the training and validation data sets. It ran for around 100 epochs.

In the end, the finished model, named “Dante Alighieri” in tandem with other names on the website, had a character error rate of around 20.58%. The character error rate finds the number of incorrect characters. It measures the counts of the number of substitutions (S), deletions (D), insertions (I), and correct characters (C). The formula used to calculate the character error rate is below:

$$CER = \frac{S + D + I}{S + D + C} \tag{1}$$

The equation calculates the CER as a percentage. However, since the CER is the percentage of inaccuracy, a lower CER indicates a higher accuracy in the model. Therefore, our accuracy of the transcriptions is around 79.42%.

Optical Character Recognition (OCR) Optical character recognition (OCR) is a technique that is frequently employed to recognize and extract text from images. Given that the provided data were images with text, we determined that OCR would be ideal in this application. The specific library we used was Pytesseract. Pytesseract is a Python library that can perform OCR. However, it is limited in its scope and often struggles with handwritten text. When our OCR model was complete, we ran it, only to find out that the model had a poor text transcription. The primary reason for the poor output was likely because the OCR model from Pytesseract was mostly trained on printed Latin and not handwritten Latin text. To combat this problem, we would have to fine-tune the model. However, that would prove difficult given that each manuscript has different handwriting styles for each character. This obstacle makes it extremely tough to tune a machine learning model with greater accuracy.

4 Results

LLMs that were fed our preprocessed data gave the highest accuracy out of all the models tested. As shown in the preprocessing section, using original data resulted in 8 mistakes, while using our preprocessed data resulted in 5 mistakes. However, the Transkribus had an 8% character error rate, which proved to be more accurate than the LLM with a 10% error rate. However, while creating the models, Google released the Gemini 2.5 Pro model for public use. The 2.5 Pro model surpassed the 92% accuracy. Therefore, LLMs still held the highest accuracy.

5 Conclusion and Future Work

In this exploratory study, we explored handwritten text recognition using Large Language Models (LLM), Transkribus, and OCRs. However, we conclude that feeding an LLM with preprocessed data was the best method for transcribing data. The preprocessing smoothed out areas of high intensity and transformed the original images of text so that the text was black and the background was white. With the preprocessed data, the words become more visible with more contrast against their background, making it easier for the LLM to transcribe. Still, LLMs could be further improved, as they had trouble identifying more cursive-like text.

In the future, if we had more resources, we would like to explore further how finer preprocessed data can improve results and fine-tune LLM models. Given different handwriting styles, an LLM model can be fine-tuned to identify various handwritten styles and achieve higher accuracy. For example, some manuscripts had cursive-like handwriting while others had straight font-like writing. Overall, we hope our project provided an opportunity to build and open more research in this area of study.

6 Division of Labor

We divided the work as follows:

- Preprocessing data: Haripriya
- Research paper and poster: Kathleen, Haripriya, Sammy, Anirudh
- Transkribus and escriptorium: Sammy
- OCR model and LLM model: Anirudh

7 Acknowledgements

We would like to acknowledge and thank Ms. Dot Porter, Ms. Haripriya Mehta, Mr. Bhagirath Mehta, Mr. Arrun Sivasubramanian, and the entire Mehta+ team for their unwavering support.

References

[1] Transkribus. READ-COOP SCE., 2025.

[2] Jeroen Ooms. Tesseract. 2025.

[3] Jajwalya Karajgikar. Reflections on south asia studies digital humanities workshop. In *The RDDS Blog*. Penn Libraries, 05 2025.

[4] StackOverflow. Preprocessing image for tesseract ocr with opencv. 2017.

[5] Andrew Morris, Viktoria Maier, and Phil Green. From wer and ril to mer and wil: improved evaluation measures for connected speech recognition. HuggingFace, 01 2004.

[6] Latin humanist - transcription. HMML School.