Analysis of Optical Character Recognition Methods for Handwritten Archival Text

Arthur De Los Santos and **Brayden Reece Leggett** and **Yasmeen Shabazz**
MIT
Quant. Methods for NLP; Fall 2024

Abstract

Transcribing archival texts is an important but laborious task. Optical Character Regocnition (OCR) could aid in this process. We test and analyze two transformer models for Optical Character Recognition, TrOCR and DTrOCR, on synthetic data with transformations to mimic different aspects of archival text. We compare their character and word error rates, as well as comparing performances between the data transformations. We also analyze the structure of the highest performing version of the DTrOCR model. Future work includes fine-tuning and testing with larger amounts of data and optimization for future use without ground truths.

1 Introduction

1.1 Archival Texts

Historic texts form the basis for modern human civilization's understanding of the past. Preserving these historical texts is an essential but difficult task. Many of these records are brittle and in danger of deteriorating further the more they are used. Creating electronic copies of these records helps to preserve them for future use, avoiding the manual sorting and searching required of maintaining records physically. Extracting the text from within these documents allows for much easier search and retrieval from large collections. However, transcription of historical texts often faces issues like unusual ligatures, faded ink, and varying font sizes or styles. The current method of manual transcription can be slow and laborious. This is where Optical Character Recognition (OCR) could be able to assist in speeding up the process.

1.2 Optical Character Recocgnition

Optical Character Recognition (OCR), or extracting text characters from images, is one of the earliest problems that pattern recognition researchers looked to solve (Mori et al., 1992). Throughout the years, state of the art machine learning models have been used for this task, from Decision Trees to Support Vector Machines. Deep learning methods such as Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN), greatly improved performance and capacity (Memon et al., 2020). Despite the improvements, these approaches fall short when dealing with complex historical documents, due to factors including varied handwriting styles, page layouts, or degradation due to age and digitizing processes. These models primarily struggle with handling contextual ambiguity, which is crucial for correctly interpreting historical texts. To overcome these issues, Transformer-based models, with their ability to capture long-range dependencies and understand context, offer a more sophisticated solution to these limitations. Some such models include a encoder-decoder transformer model, TrOCR (Li et al., 2022), and a decoder-only transformer model, DTrOCR (Fujitake, 2023). This technology could revolutionize access to historical documents, improve cultural heritage preservation, or assist in new historical research.

1.3 OCR for Archival Texts

We believe that transformer-based OCR models show promise in transcribing historical texts. We focus our efforts on handwritten texts, as they are less likely to already be transcribed than print texts.

Since manual transcription of historical data is an extremely resource-intensive project, existing datasets of historical handwriting that have already been transcribed are minimal. Indeed, lack of data for OCR in general has led to the creation and utilization of synthetic image-text data alongside real-world datasets. We aim to use these existing image-text generation algorithms, namely TextRecognitionDataGenerator (Belval, 2020), to generate synthetic data that is visually similar to archival texts. We will use this synthetic data to fine-tune and

test our models. We will also test the models on the Washington IAM Historical Document Dataset, one existing dataset of historical handwriting, to ensure model generalizability to real world data.

Transformer methods show promise in OCR as a whole. TrOCR (Transformer-based Optical Character Recognition) (Li et al., 2022) uses image transformer encoders and text transformer decoders. DTrOCR (Decoder-only Transformer for Optical Character Recognition) (Fujitake, 2023) uses GPT-2 to predict the word, with image embeddings as input. We hypothesize that DTrOCR will be the most accurate in transcribing our data, as it claims better performance on existing handwritten data using the metric of character error rate. We will also analyze the structure of DTrOCR as we fine-tune it on our synthetic data, and which model features and data combinations yield the best overall performance.

2 Related Works

In past OCR research, RNNs such as Long Short-Term Memory (LSTM) networks have been utilized to address the challenges of recognizing text in blurred, camera-captured documents. Traditional OCR systems like ABBYY Fine Reader and Tesseract rely heavily on pre-processing steps like binarization, which often falter on blurred images. Asad et al. (2016) directly applied LSTMs to grayscale images, particularly for handling motion and out-of-focus blur commonly found in smartphone-captured documents. Using a bidirectional LSTM architecture, their approach achieved a significantly lower character error rate on the SmartDoc-QA dataset compared to traditional OCR systems, demonstrating the LSTM's ability to manage degraded text without the need for binarization or deblurring.

To enhance OCR for historical documents, researchers have explored preprocessing methods, including page segmentation, deskewing, and retraining on domain-specific data. Gruber et al. (2021) proposed a pipeline integrating Faster R-CNN for page segmentation, an improved deskew algorithm, and OCR using a retrained Tesseract model fine-tuned on synthetic and real data. This method particularly addressed challenges in multi-language documents with variable quality. By tailoring preprocessing techniques and data generation, this approach improved OCR accuracy on complex historical archives .

Hybrid CNN-LSTM networks have proven effective for OCR tasks involving complex historical text. Brandt Skelbye and Dannélls (2021) applied OCR methods to Swedish newspapers from the 19th century. They used CNN layers to handle feature extraction from image data, while LSTM layers captured sequential dependencies in character recognition, creating a robust OCR framework for historical documents. Using the open-source OCR engine Calamari, the study demonstrated that these CNN-LSTM hybrids significantly outperform traditional systems on Swedish newspaper archives. Their approach, which combined deep neural networks with voting mechanisms for model outputs, highlighted the potential of hybrid models in improving OCR accuracy across varied historical text formats.

In recent OCR research, transformer-based models such as TrOCR have emerged as significant advancements over traditional CNN-RNN architectures. These models eliminate the reliance on convolutional layers by employing transformers for both image encoding and text generation. TrOCR, as proposed by Li et al. (2022), leverages pre-trained vision and language transformers to decode text without complex pre- and post-processing steps, achieving state-of-the-art results across printed, handwritten, and scene text datasets. This transformer-driven approach has shown promising accuracy in handling diverse text images while reducing computational demands, making it ideal for real-world applications, especially those involving complex and varied text forms.

The DTrOCR (Decoder-only Transformer for Optical Character Recognition) model introduces an innovative approach to text recognition by using a decoder-only transformer architecture, unlike traditional encoder-decoder models. This model by Fujitake (2023) bypasses complex feature extraction by feeding image patches directly into the decoder, leveraging a generative language model trained on vast NLP datasets. Its simplified architecture enables DTrOCR to efficiently recognize printed, handwritten, and scene text in both English and Chinese. Compared to other OCR models that rely on intricate pre- and post-processing steps, DTrOCR achieves state-of-the-art accuracy on benchmarks with fewer parameters and without a vision-specific encoder, making it highly adaptable for various OCR applications.

3 Methods

We analyze two transformer models for Optical Character Recognition, TrOCR and DTrOCR, whose structures are expanded upon in 3.2. As implemented in their respective papers, DTrOCR achived lower character error rate than TrOCR. However, a publicly available version of TrOCR exists as trained by Microsoft, whereas a similar version of DTrOCR does not exist. Therefore, we use the skeleton code provided by Rajan (2024) in *A Pytorch Implementation of DTrOCR* to pretrain and fine-tune the DTrOCR model on data detailed in 3.1. We analyze the structure of the fine-tuned DTrOCR model. We then compare the character error rate between models based on their predictions and the real text, using synthetic data generated to mimic different aspects of archival text. We also compare performances between different types of text transformations. Finally, we compare performance of both final models on real world archival text from the Washington IAM Historical Document Dataset.

3.1 Datasets

We focus our analysis on handwritten archival texts because they are more likely than printed text to be the only existing physical copy of the writing. They are also extremely likely to not have an existing transcription. Due to the lack of transcribed data, synthetic data is often used to train and test OCR models, including our own.

Existing Synthetic Data: HW-Synth (IIIT-HWS) is a corpus of 9M synthetic handwritten word images created by Krishnan and Jawahar (2016). This data is used to pretrain our implementation of the DTrOCR model.

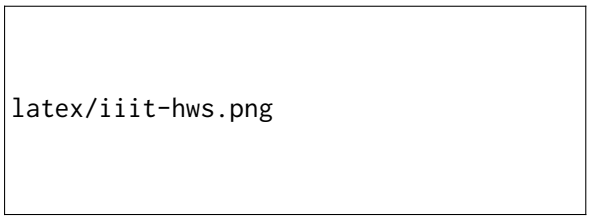

Figure 1: IIIT-HWS Data

Newly Generated Data: We leveraged TextRecognitionDataGenerator by Belval (2020) to generate our own synthetic text images for fine-tuning our DTrOCR model. 20 handwriting fonts were used to generate 90k lines of text, as well as 90k single word images. These images had various transformations applied to them to mimic archival text. Three different backgrounds were tested: white, sepia, and random gaussian noise. For each of the backgrounds, clear, blurry, and distorted versions of the texts were generated. To improve model capability and better capture the performance of the original DTrOCR model, 180k of printed text was also generated with the same specifications. An additional 1k of each data type was generated for testing both models, 18k total lines and 18k total words.

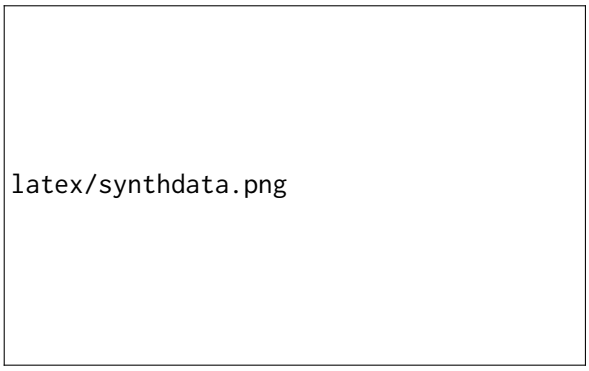

Figure 2: Generated Data, in categories

Real World Data: Among the existing datasets of archival texts is the IAM-Histstorical Document Database (Fischer et al., 2012), which contains handwritten historical manuscript images and their corresponding texts. A subset of this data, the Washington database, was created from the George Washington Papers at the Library of Congress and contains 656 text lines and 4,894 word instances. We use this dataset to test the models' performances on real world data.

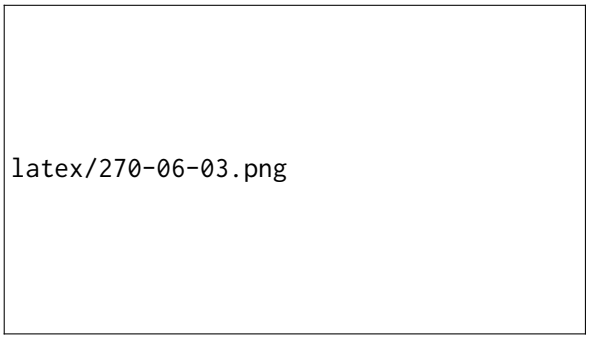

Figure 3: Washington IAM Handwriting Data

3.2 Models

We use two transformer based models and evaluate their performance on our data.

TrOCR (Li et al., 2022): This Transformer-based OCR model is an encoder-decoder utilizing cross attention. There are several versions available for use on HuggingFace; we used the 558M parameter "trocr-large-handwritten". It was initialized from pretrained image encoders and text decoders and subsequently pre-trained on a dataset of hundreds of millions of synthetically generated textlines. This version in particular was fine tuned on the IAM dataset.

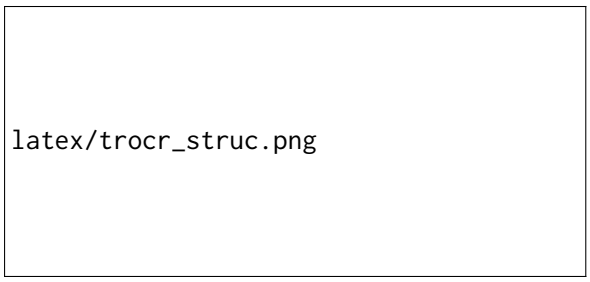

Figure 4: Structure of TrOCR model, Li et al. (2022)

DTrOCR Fujitake (2023): This transformer based model is hypothesized to perform better on archival text data compared to the TrOCR model based on the DTrOCR paper. This model uses patch embeddings for input images and twelve GPT2 transformer blocks for decoding.

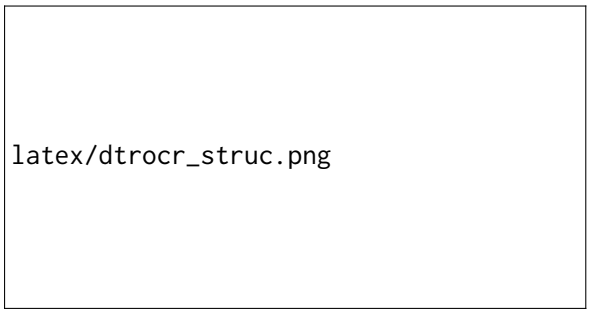

Figure 5: Structure of DTrOCR model, Fujitake (2023)

3.3 DTrOCR Model Process and Analysis

The original DTrOCR paper did not have a pre-trained model available. Therefore, we trained a base model developed by (Rajan, 2024) using the IIIT-HWS single-word handwriting database with 2.75 million images for training and 550 thousand images for validation. Although we achieved a validation accuracy of 99.8% with 7 epochs, testing on our single word synthetic data yielded high character error rates.

To improve the model after pretraining, we tried a number of methods. First, we considered retraining a model only on our synthetic data without using the pretrained model as a base, to ensure that starting with the pretrained model would be the best initialization for our fine-tuning. Then, we proposed an architecture change that would change the maximum positional embedding from 512 to 256 tokens, since this would decrease the training time and provide more regularization for our model's training. We also considered freezing the first half of the weights (the first 5 layers of the transformer layers) to continue using our pretrained model with a decreased fine-tuning time and improved generalization of the original single words data to work for multiple words. Finally, we proposed using the pretrained weights without freezing layers and fine-tuning on our synthetic data. Because our fine-tuning synthetic data consists of lines and words of printed and handwritten text with about 290 thousand images for training and 70 thousand images for validation, we expected that one of these models would be able to generalize well for unseen characters, words, and lines of words.

Figure 6: DTrOCR Training Results

DTrOCR (before fine-tuning)			
Words	White	Sepia	Noise
Clear	0.5541	0.6087	0.5532
Blur	0.4217	0.4364	0.4111
Distort	0.6644	0.7262	0.6631

Table 1: DTrOCR CER on synthetic handwriting

We chose fine-tuning the pretrained model without frozen weights because it yielded a much higher accuracy compared to the other models. Due to unforeseen circumstances, the training was halted early, achieving a validation accuracy of 76.8% after 9 additional epochs. The final fine-tuned model, compared to the pre-trained model, had an equal percent change in the magnitude of the weights throughout the layers, with a slightly higher change in the first layer, further motivating why freezing the first layers yielded poorer performance.

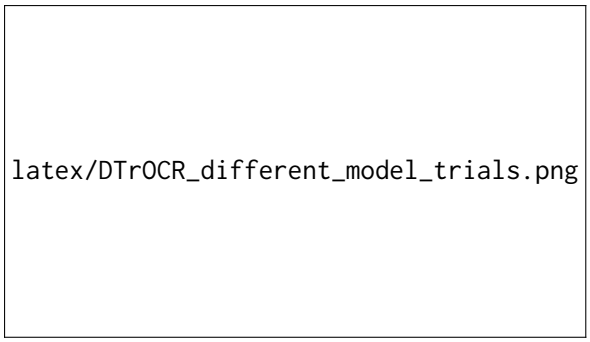

Figure 7: DTrOCR Model Testing

Figure 8: DTrOCR Final fine-tuning Results

4 Results

The main metric used to analyze our models' performance is Character Error Rate (CER).

$$\text{CER} = \frac{\text{errors}}{\text{characters}} = \frac{\text{substitutions + insertions + deletions}}{\text{total characters}}$$

4.1 Synthetic Testing Data

We tested both models on generated synthetic handwriting, comparing their Character Error Rates on different image transformations for both lines and single words. Overall, TrOCR performed better on lines and DTrOCR performed better on words.

TrOCR performed the best on images with a noisy background and no other transformation, and the worst on distorted text with a sepia background. Notably, TrOCR had a much higher performance on lines than on single words. These trends are likely due to the training data, which was likely lines of text that had some noise in the background.

DTrOCR performed the best on blurry text with a white background. This format is the most similar to the training data, as seen in 1

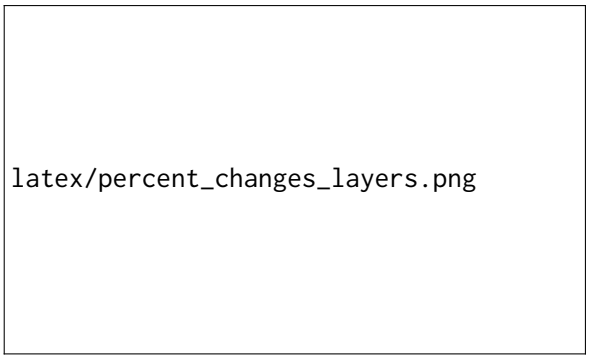

Figure 9: DTrOCR weights' percent change

TrOCR			
Lines	White	Sepia	Noise
Clear	0.0909	0.1013	**0.0799**
Blur	0.0999	0.1306	0.0933
Distort	0.1196	0.1584	0.1183
Words	White	Sepia	Noise
Clear	0.1867	0.2359	**0.1805**
Blur	0.1986	0.2037	0.2078
Distort	0.3794	0.4423	0.3625

Table 2: TrOCR CER on synthetic handwriting

DTrOCR			
Lines	White	Sepia	Noise
Clear	0.3769	0.4298	0.4038
Blur	**0.2317**	0.2997	0.2525
Distort	0.2972	0.3384	0.2987
Words	White	Sepia	Noise
Clear	0.0096	0.0108	0.0121
Blur	**0.0084**	0.0132	0.0090
Distort	0.0116	0.0141	0.0198

Table 3: DTrOCR CER on synthetic handwriting

4.2 Real world Testing data

Model Performance on real world data dropped for both models. In this case, TrOCR had a lower CER, but it was still high at 0.6642. To further analyze the model performance, we broke down the CER into the types of errors: Insertions, Deletions, and Substitutions.

DTrOCR's most common error type was substitution, at almost three quarters of its errors (73.89%), whereas TrOCR made the most insertions (54.64%). Both models' least common error was deletions; 11.1% for DTrOCR and only 1.6%

Figure 10: Example CER breakdown

for TrOCR. One common specific error for TrOCR was to add a space and period to the end of its prediction, 30% of it's predictions on this data ended with this sequence.

We also tested the model performances on lines in the Washington dataset; while TrOCR did much better, with a CER of .1549 CER, DTrOCR did much worse, getting almost nothing correct with a CER of .9107. This underscores the fact that our implementation of DTrOCR only performs well on data it has seen; it was pre-trained solely on words and was fine-tuned on a comparatively small amount of lines.

	TrOCR	DTrOCR
Character Error Rate	0.6642	0.7389
Insertion Rate	0.5465	0.1552
Deletion Rate	0.0158	0.1106
Substitution Rate	0.4377	0.7342
Total Character Errors	**14812**	**16230**

Table 4: Performance on real single word data

5 Discussion/Future Work

Our results show that both models are highly susceptible to their training and fine-tuning data. TrOCR benefited from training on a significantly more diverse synthetic dataset and subsequent fine-tuning, leading to a better performance on the real world data. However, DTrOCR performed extremely well on data that it was familiar with, suggesting a capacity for better overall performance if the correct data is used to train the model. Even after fine-tuning on the synthetic data, while distortion remains the most error-prone transformation there is much less of a distinction between it and the other transformations than before the fine-tuning.

Both models struggled with distorted handwriting images, suggesting that fine-tuning on distortion-specific data could be a promising avenue for improving CER across models. While we were able to compare the DTrOCR and TrOCR on single word images, limited time and resources kept us from adequately training the DTrOCR on text lines. Future work will investigate the effects of fine-tuning on distorted data to assess whether such an approach enhances general robustness and performance across all noise types and conditions.

Acknowledgments

Quant. Methods for NLP Fall 2024 Staff

TrOCR (Li et al., 2022)

Pytorch Implementation of DTrOCR (Rajan, 2024)

TextRecognitionDataGenerator (Belval, 2020)

IIIT-HWS dataset (Krishnan and Jawahar, 2016)

The authors acknowledge the MIT SuperCloud and Lincoln Laboratory Supercomputing Center for providing (HPC, database, consultation) resources that have contributed to the research results reported within this paper/report.

References

F. Asad, A. Ul-Hasan, F. Shafait, and A. Dengel. 2016. High performance ocr for camera-captured blurred documents with lstm networks. In *2016 12th IAPR Workshop on Document Analysis Systems (DAS)*, pages 7–12.

Belval. 2020. Textrecognitiondatagenerator.

M. Brandt Skelbye and Dana Dannélls. 2021. Ocr processing of swedish historical newspapers using deep hybrid cnn–lstm networks. *ACL Anthology*.

A. Fischer, A. Keller, V. Frinken, and H. Bunke. 2012. Lexicon-free handwritten word spotting using character hmms. *Pattern Recognition Letters*, 33(7):142642–142668.

Masato Fujitake. 2023. Dtrocr: Decoder-only transformer for optical character recognition. *Preprint*, arXiv:2308.15996.

I. Gruber, M. Hruž, P. Ircing, P. Neduchal, T. Zítka, M. Hlaváč, Z. Zajíc, J. Švec, and M. Bulín. 2021. Ocr improvements for images of multi-page historical documents. In *Speech and Computer. SPECOM 2021. Lecture Notes in Computer Science*, volume 12997. Springer, Cham.

Praveen Krishnan and C. V. Jawahar. 2016. Matching handwritten document images. *CoRR*, abs/1605.05923.

Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei Florencio, Cha Zhang, Zhoujun Li, and Furu Wei. 2022. Trocr: Transformer-based optical character recognition with pre-trained models. *Preprint*, arXiv:2109.10282.

Jamshed Memon, Maira Sami, Rizwan Ahmed Khan, and Mueen Uddin. 2020. Handwritten optical character recognition (ocr): A comprehensive systematic literature review (slr). *IEEE Access*, 8:142642–142668.

S. Mori, C.Y. Suen, and K. Yamamoto. 1992. Historical review of ocr research and development. *Proceedings of the IEEE*, 80(7):1029–1058.

A. Rajan. 2024. A pytorch implementation of dtrocr: Decoder-only transformer for optical character recognition [computer software].

A Impact Statement

Our research contributes to advancing Optical Character Recognition (OCR) technology for historical texts, potentially transforming the way archivists and researchers interact with historical material. This work facilitates digitization efforts, enabling broader accessibility, search-ability, and preservation of historical texts. Applications of this research extend beyond archivists, possibly reaching museums, libraries, academic researchers, and even private entities hoping to preserve personal collections.

While promising, the deployment of these types of models carries societal implications. Over-reliance on OCR models can lead to the overlooking of transcription errors, which might propagate inaccuracies in historical records. Moreover, the potential misuse of OCR tools for selective censorship or misinformation underscores the need for careful oversight in their application.

To mitigate such risks, future research should focus on improving model interpretability and transparency, enabling users to verify and correct outputs efficiently. Additionally, initiatives that combine automated OCR with crowd-sourced validation could improve model accuracy and scalability. Addressing the societal challenges of trust and misuse will ensure that this technology serves as a responsible tool for preserving and understanding our collective history.