

# From Press to Pixels: Evolving Urdu Text Recognition

Samee Arif\*, Sualeha Farid\*

University of Michigan  
asamee@umich.edu, sualeha@umich.edu

## Abstract

This paper presents a comparative analysis of Large Language Models (LLMs) and traditional Optical Character Recognition (OCR) systems on Urdu newspapers, addressing challenges posed by complex multi-column layouts, low-resolution scans, and the stylistic variability of the Nastaliq script. To handle these challenges, we fine-tune YOLOv11x models for article- and column-level text block extraction and train a SwinIR-based super-resolution module that enhances image quality for downstream text recognition, improving accuracy by an average of 50%. We further introduce the Urdu Newspaper Benchmark (UNB), a manually annotated dataset for Urdu OCR comprising 829 paragraph images with a total of 9,982 sentences. Using UNB and the OpenITI corpus, we conduct a systematic comparison between traditional CNN+RNN-based OCR systems and modern LLMs, presenting detailed insertion, deletion, and substitution error analyses alongside character-level confusion patterns. We find that Gemini-2.5-Pro achieves the best performance on UNB (WER 0.133), while fine-tuning GPT-4o on just 500 in-domain samples yields a 6.13% absolute WER improvement, demonstrating the adaptability of LLMs to low-resource, morphologically complex scripts like Urdu. The UNB dataset and fine-tuned models are publicly available at <https://github.com/sameearif/urdu-newspaper-benchmark>.

**Keywords:** text recognition, low-resource languages, llms, optical character recognition, super-resolution, segmentation

## 1. Introduction

Optical Character Recognition (OCR) plays a critical role in the digitization of printed material, enabling large-scale archival, search, and accessibility applications across languages and domains (Adenekan, 2024). In particular, newspaper OCR is essential for preserving records, supporting digital journalism, and making large volumes of textual data machine-readable. OCR also facilitates broader accessibility when paired with systems such as Text-to-Speech models, it can make printed content available to visually impaired users or those with reading difficulties. However, OCR for newspapers presents unique challenges due to complex layouts, variable font styles, and poor print quality.

These challenges are further compounded in low-resource languages like Urdu, where script is inherently cursive, with context-sensitive letter forms, frequent ligatures, and baseline variations that make segmentation and recognition difficult (Zaki et al., 2019). Urdu newspapers introduce an additional layer of complexity with multi-article, multicolumncluttered layouts, and stylized fonts, all of which further degrade the performance of conventional OCR pipelines. A particularly important factor in Urdu OCR is the typographic distinction between the Naskh and Nastaliq scripts. Although most OCR systems, including Tesseract (Smith, 2007), EasyOCR (Al, 2020), Kraken (Kießling, 2020), and UTRNet (Rahman et al., 2023), achieve reasonable performance on Naskh, they struggle

در این خاک در این خاک در این مزرعه پاک  
بجز مهر بجز عشق دگر تخم نکاریم  
— مولانا جلال الدین محمد بلخی

در این خاک در این خاک در این مزرعه پاک  
بجز مهر بجز عشق دگر تخم نکاریم  
— مولانا جلال الدین محمد بلخی

Figure 1: Example of Naskh (top) and Nastaliq (bottom).

significantly with the Nastaliq script. The models trained on Urdu book dataset may perform adequately on data from the same domain but fail to generalize to other real-world settings like newspapers. Building on these challenges, we address three key research questions:

1. **RQ1:** Can multimodal LLMs outperform traditional CNN/RNN-based OCR systems on the complex Urdu scripts and generalize across domains such as books and newspapers?
2. **RQ2:** How much can layout-aware preprocessing and image enhancement via fine-tuned YOLOv11x segmentation and SwinIR-based super-resolution improve downstream text recognition accuracy?
3. **RQ3:** To what extent can LLMs be efficiently adapted to low-resource languages through limited in-domain fine-tuning?

To investigate these questions, we introduce the Urdu Newspaper Benchmark (UNB). Using UNB and the OpenITI corpus, we compare traditional OCR engines with modern LLMs, analyze their error patterns, and demonstrate that fine-tuning LLMs for text recognition on a small sample size can yield substantial performance gains. Our analysis provides detailed insights into where OCR models struggle, such as ligature segmentation, diacritic handling, and character-level substitutions, offering diagnostic evidence for future Urdu and low-resource script OCR research.

## 2. Related Work

### 2.1. Open-source OCR Engines

Tesseract is one of the most widely used OCR engines. It follows a traditional CNN+LSTM architecture and supports Urdu in its default language pack. EasyOCR uses a CRNN-based pipeline with CTC loss, providing a general-purpose OCR framework for over 80 languages, including Urdu. Kraken is a flexible OCR engine based on OCRopus and designed to support right-to-left and complex scripts. It allows custom model training and has been successfully applied to Arabic-script languages. TrOCR (Li et al., 2022) is a Transformer-based OCR model that uses pre-trained vision and language models in an encoder–decoder architecture. It achieves state-of-the-art results on printed, handwritten, and scene text recognition tasks.

### 2.2. Prior Work in Urdu OCR

Early work in Urdu OCR focused on creating segmentation-based approaches such as those by Sabbour and Shafait (2013) who reported an accuracy of 91% on clean Urdu text, and Tabassam et al. (2009) who developed an OCR system for Urdu Naskh script utilizing pattern matching techniques, achieving an accuracy of 89%. However, these methods often struggled with the cursive and context-sensitive nature of the Urdu script.

Segmentation-free approaches for Urdu OCR are also explored. Shabbir and Siddiqi (2016) proposed a system that recognizes Urdu words in the Nastaliq script using ligatures as units of recognition, employing Hidden Markov Models (HMMs) for classification. Similarly, Khattak (2017a) presented an OCR system for printed Urdu text, also using HMMs and focusing on ligature recognition. These approaches demonstrated improved handling of the complex ligature structures of the Urdu script. UTRNet is a recent CNN+RNN-based model specifically designed for Urdu text recognition which uses a deep convolutional encoder combined with recurrent and attention-based decoders. Trained on a synthetically generated dataset and

the UPTI corpus, UTRNet achieves state-of-the-art accuracy (92.97%) on its test set.

### 2.3. Recent Advances in Multilingual OCR

In recent years, there has been significant advancement in multilingual OCR systems with transformer-based architectures. For example, the MIT-10M dataset by Khattak (2017b) contains 10 million plus image-text pairs in 14 languages. In another work by Wang et al., the authors proposed the use of LLMs for OCR pipelines and discussed ways to mitigate hallucinations to enhance reliability in multilingual image-to-text pipelines (Wang et al., 2024). However, Urdu remains underrepresented in both large-scale datasets and employing current LLM based methods for this task.

### 2.4. Urdu Text Recognition Datasets

Several datasets have been introduced to support OCR research for the Urdu Nastaliq script. Naz et al. (2016) introduced the Urdu Printed Text Image (UPTI) dataset, containing over 10,000 synthetic printed text-line images for benchmarking printed Nastaliq recognition. Ahmed et al. (2017) presented the Urdu Nastaliq Handwritten Dataset (UNHD), comprising samples from 500 writers with more than 312,000 text-line images, making it the first large-scale handwritten Nastaliq corpus. Rahman et al. (2023) introduced UTRSet-Real (11k real annotated lines) and UTRSet-Synth (20k synthetic lines) for printed OCR, along with the Urdu-Doc dataset of full-page document images for layout analysis. Chandio et al. (2020) released three datasets covering isolated characters, cropped words, and 8,312 end-to-end text-line images for detection and recognition.

## 3. Methodology

### 3.1. Data Collection and Preparation

Table 1 summarizes the dataset sizes used for each stage of our pipeline.

Dataset	Train Size	Test Size
Article Segmentation	26,830	2,975
Column Segmentation	3,969	440
Super-Resolution	38,512	4,279
UNB (Nastaliq)	-	829
OpenITI (Nastaliq)	-	250
OpenITI (Naskh)	-	250

Table 1: Dataset sizes used for article segmentation, image super-resolution, and text recognition.



Figure 2: OCR pipeline with article and column segmentation, super-resolution, and LLM-based text recognition.

**Text Recognition Datasets.** As part of this work, we introduce Urdu Newspaper Benchmark (UNB), a new manually annotated OCR dataset of Urdu newspaper scans written in the Nastaliq script. It comprises 829 samples, each consisting of full text blocks (rather than isolated lines or sentences), resulting in 9,982 sentences and 9,758 unique words. Each sample is available in both low-resolution and high-resolution ( $4\times$ ) variants. The annotation task involved transcribing Urdu text from scanned images, a task where correctness is largely objective. To ensure quality, we employed a two-pass annotation process: the first annotator transcribed each image, and a second independently reviewed and corrected any errors using a custom Streamlit interface displaying the scan and text side-by-side.

To assess generalization across font styles and domains, we also evaluate the models on the paragraph-level images from OpenITI corpus,<sup>1</sup> which contains printed Urdu text in both Naskh and Nastaliq scripts from books. This dual-source evaluation enables us to benchmark both traditional OCR models and LLMs in terms of their robustness to script variation and domain shift from books to newspaper content.

**Segmentation Datasets.** For article and column segmentation, we constructed training datasets using publicly available Urdu newspaper scans sourced from online archives. For the image super-resolution dataset, we generated low-quality inputs by downscaling high-resolution Urdu newspaper scans by a factor of 4 and reducing the quality by 30% using JPEG compression.

### 3.2. OCR Architecture

Our end-to-end OCR pipeline consists of four main stages: article segmentation, image super-resolution, column segmentation, and text recognition using LLMs. Figure 2 illustrates the complete workflow.

In the first stage of our pipeline, we use a fine-tuned YOLOv11x (Lisa and Bot, 2017) model for

article-level segmentation, which isolates individual articles from cluttered, multi-article newspaper layouts. This step is essential because performing column segmentation directly would make it impossible to preserve article boundaries because of the cluttered newspaper layout. In the second stage, we apply a super-resolution model, SwinIR (Liang et al., 2021) to the cropped article images. We perform this step before column segmentation because enhancing the visual quality produces sharper edges and better-defined text regions, which in turn improves the accuracy column segmentation. In the third stage, we perform column segmentation within each article using a separate YOLOv11x model. This further simplifies the layout by breaking complex, multi-column articles into manageable single-column text blocks, which improves OCR accuracy across models. Finally, in the fourth stage, we pass the high-quality column segments to an LLM for text recognition.

### 3.3. Experimental Setup

Our experimental pipeline is divided into four main components: segmentation, super-resolution, text recognition, and error analysis. Below, we detail the setup for each stage and the metrics used.

**Image Segmentation.** We fine-tune and evaluate the YOLOv11x model for both article- and column-level segmentation using standard object detection metrics: precision, recall, mAP@50, and mAP@50:95.

**Image Super-Resolution.** The SwinIR model is trained and evaluated using Peak Signal-to-Noise Ratio (PSNR) to measure image reconstruction quality. To assess the impact of super-resolution on OCR performance, we run each LLM on both the original low-resolution images and their enhanced high-resolution counterparts generated by SwinIR. We then compare the Word Error Rate (WER) and Character Error Rate (CER) across these two conditions to quantify the benefit of super-resolution. Additionally, we run our column segmentation model on both versions of the im-

<sup>1</sup>[https://github.com/OpenITI/arabic\\_print\\_data](https://github.com/OpenITI/arabic_print_data)

ages and measure performance using mAP@50. This allows us to quantify how super-resolution improves the reliability of layout detection when applied prior to column segmentation.

**Text Recognition.** We evaluate text recognition performance using standard OCR metrics, WER and CER across two categories, traditional OCR models and LLMs.

1. **Baseline OCR Models.** We benchmark Tesseract, EasyOCR, Kraken, UTRNet, and TrOCR on both the OpenITI corpus (in Naskh and Nastaliq scripts) and our UNB dataset.
2. **Large Language Models.** We further evaluate Gemini-2.5-Pro (gem, 2024), GPT-4.1, GPT-4o (OpenAI, 2024), Claude-3.7-Sonne (Anthropic), Llama-4-Maverick, and Llama-4-Scout (Meta) on the same datasets to assess their zero-shot OCR capabilities. To test adaptability, GPT was fine-tuned on a 500-image subset of UNB and evaluated on the remaining 329 test images. All evaluations were performed at temperature 0 using a fixed system prompt and user template (given below) for reproducibility.

#### System Prompt

You are an OCR system. Your job is to transcribe image text exactly as shown, without interpretation, paraphrasing, translation, summarization, or hallucination.

#### User Message

Extract the exact text from this image. Preserve sentence structure NOT spacing. If anything is unreadable, write '[UNREADABLE]'.

## 4. Results and Discussion

### 4.1. Lexical Diversity Analysis of UNB

To demonstrate the lexical richness of UNB, we evaluate several established measures of lexical diversity:

1. **Moving-Average Type-Token Ratio (MATTR) (RDRR):** This metric measures the proportion of unique words within moving windows of text, providing a stable estimate of vocabulary variety independent of text length. For UNB, MATTR@50 is 0.803 and MATTR@100 is 0.694, indicating that even across longer spans, the proportion of novel words remains high and repetition is limited.

2. **Measure of Textual Lexical Diversity (MTLD) (McCarthy and Jarvis, 2010):** MTLD quantifies how many words can be read before the type-token ratio falls below a threshold, thus reflecting how quickly lexical repetition occurs. UNB achieves an MTLD of 82.15, which is considered strong and suggests a consistently rich vocabulary across extended text segments.
3. **Hapax Legomena (Davis, 2018):** This measure captures the proportion of word types that appear only once in the corpus, emphasizing the presence of rare or distinctive words. In UNB, 39.7% of all word types are hapax legomena, revealing a pronounced long-tail distribution typical of real-world linguistic data.
4. **Heaps' Law Exponent ( $\beta$ ):** Heaps' Law models the rate at which vocabulary size increases with corpus size, where higher  $\beta$  values imply faster lexical growth. For UNB,  $\beta = 0.629$ , showing that the vocabulary continues to expand rather than saturate, confirming substantial lexical diversity as the dataset scales.

### 4.2. Image Segmentation

We evaluate article and column-level segmentation performance using standard object detection metrics as shown in Table 2.

Metric	Segmentation	
	Article	Column
Precision	0.963	0.970
Recall	0.971	0.997
mAP@50	0.975	0.975
mAP@50:95	0.866	0.854

Table 2: Scores for article and column segmentation.

**Article Segmentation.** For article-level segmentation, our fine-tuned YOLOv11x delivers both high accuracy and coverage. It achieves a precision of 0.963, meaning over 96% of its detected regions are true articles, and a recall of 0.971, capturing more than 97% of all actual article blocks. It also scores an mAP@50 of 0.975 and an mAP@50:95 of 0.866, demonstrating robust detection under both lenient and strict IoU thresholds. Together, these metrics confirm that our model reliably isolates article regions from the complex, multi-column layouts of Urdu newspapers.

**Column Segmentation.** Our fine-tuned YOLOv11x demonstrates outstanding performance on column-level segmentation. The precision of 0.970 indicates that 97.0% of all predicted column regions correspond to actual columns, minimizing

false positives. The recall of 0.997 indicates that 99.7% of true column regions are successfully detected, ensuring almost no columns are missed. Under a loose IoU threshold (mAP@50), the model achieves 0.975, reflecting high tolerance for localization variation. Even when evaluated under the stricter mAP@50:95 metric, it maintains a robust score of 0.854, showing consistent accuracy across tighter overlap requirements. These results confirm that the model can reliably identify and localize column segments in complex Urdu newspaper layouts.

We perform column segmentation before text recognition to address challenges posed by the multi-column layout of Urdu newspapers. Without this step, LLMs tend to extract text across columns non-sequentially, resulting in jumbled and incoherent output. This misordering disrupts sentence structure and readability. Figure 3 illustrates a reading path followed by an LLM when extracting text directly from a multi-column article image, highlighting the need for explicit column segmentation.

1	لاہور (این این آئی) پاکستان مسلم لیگ (ن) لاہور
2	افواج، سکیورٹی فورسز اور قانون نافذ کرنے والے
3	کے صدر سیف الملوک کو کھڑے کہا ہے کہ ترقی کرنا
4	ادارے دیکھو دوں کا قلع قمع کرنے تک جین سے
5	پاکستان ہمارے دشمنوں کی آنکھوں میں کھٹکتا ہے،
6	نہیں بیٹھیں گے اور ان کا ہر جگہ چھپا کیا جائے گا۔
7	بھارت کی مالی معاونت سے دہشتگردی کرنے والے
8	ملاقات کے لئے آنے والے کارکنوں سے گفتگو
9	نہ پہلے پاکستانی قوم کے عزم کو جھڑل کر کے نہ
10	آگے اٹھ سکتے ہیں۔ دہشتگردوں اور دہشتگردوں کے خلاف
11	پاکستان جب بھی (باقی صفحہ 4 بجیہ نمبر 3 پر)
12	

Figure 3: The sequence of text extracted by the LLMs from multi-column image.

### 4.3. Image Super-Resolution

After 100 epochs of training article-segmented data, SwinIR achieves a 32.71 dB PSNR on the test set. This level of image reconstruction is noteworthy given the challenges posed by Urdu script. A PSNR above 30 dB is generally considered strong for image restoration tasks (Kim et al., 2025; Anandhi and Mahalingam, 2025). In the context of Urdu OCR, it shows that the textual feature such as stroke continuity and glyph boundaries are effectively preserved which is essential for accurate downstream text recognition. Figure 4 presents visual examples of low- and high-resolution Urdu newspaper images, demonstrating the improvements achieved by our model.

We observe a significant improvement in column segmentation performance due to super-resolution. Specifically, mAP@50 for column segmentation improves from 0.928 to 0.975, representing a gain of approximately 5.1%.

The super-resolution model significantly boosts text recognition across all LLMs, as shown in Table 4. Gemini-2.5-Pro improves by 24.9%, GPT-4.1 by 62.8%, GPT-4o by 58.0%, and Llama-4-Maverick by 70.6% in WER after super-resolution.



(a) Input



(b) Output



(c) Input



(d) Output

Figure 4: Side-by-side comparisons of input and output of super-resolution model.

Claude-3.7-Sonnet and Llama-4-Scout fail to process most of the low-resolution images, often returning messages such as "Unfortunately, I am unable to extract text from the image...", "I can't directly extract text...when the image quality does not allow for clear text recognition.", and "The image contains text in what appears to be Urdu script..." instead of performing any transcription. However, when evaluated on high-resolution images, Claude achieves a WER of 0.249, while Llama-4-Scout records a WER of 0.347.

### 4.4. Text Recognition

**Baseline Models.** Table 3 presents WER and CER for five OCR models across three datasets. TrOCR delivers the best overall performance, achieving the lowest errors on OpenITI-Nastaliq (WER 0.451, CER 0.177) and UNB (WER 0.422, CER 0.159), and the best WER on OpenITI-Naskh (0.205). Kraken is strongest among the CNN/RNN systems and attains the lowest CER on OpenITI-Naskh (0.069), but degrades substantially on UNB (WER 0.558, CER 0.221), indicating limited cross-domain robustness. Tesseract is weakest overall (e.g., UNB WER 2.401, CER 2.580). UTRNet and EasyOCR show similar brittleness across script and domain shifts. Overall, all models perform markedly better on Naskh than on Nastaliq confirming the challenges posed by Nastaliq's complex ligatures and non-linear structure. Furthermore, when transitioning from the OpenITI book dataset

Model	OpenITI-Naskh		OpenITI-Nastaliq		UNB (High-Resolution)	
	WER	CER	WER	CER	WER	CER
<b>Tesseract</b>	0.902	0.955	1.567	1.420	2.401	2.580
<b>EasyOCR</b>	0.532	0.177	0.904	0.392	0.802	0.246
<b>Kraken</b>	0.249	0.069	0.626	0.305	0.558	0.221
<b>UTRNet</b>	0.989	0.741	0.862	0.638	0.602	0.306
<b>TrOCR</b>	0.205	0.128	0.451	0.177	0.422	0.159

Table 3: CER and WER of baseline models on the three datasets.

to the out-of-domain newspaper scans in UNB, all models exhibit significant performance degradation, underscoring their limited ability to generalize across domains and highlighting the need for more robust, domain-adaptive OCR solutions.

**Modern LLMs.** Table 4 presents WER and CER for six large language models across the same datasets. **Gemini-2.5-Pro** delivers the best overall performance, achieving a WER of 0.303 on OpenITI-Nastaliq and 0.133 on UNB representing a 50% improvement compared to the best baseline model (TrOCR, WER 0.422). GPT-4.1 and GPT-4o follow with WERs of 0.443 and 0.327, respectively, while Claude-3.7-Sonnet and both Llama-4 variants perform strongly on Naskh but degrade on Nastaliq, reflecting the script’s inherent complexity. Across datasets, LLMs consistently maintain lower error rates and narrower domain gaps: where CNN/RNN models’ WER often exceeds 0.6 on Nastaliq, LLMs remain below 0.45. These results highlight the superior adaptability of transformer-based large scale multimodal architectures, which capture both visual and linguistic structure. By leveraging large-scale pretraining, models like Gemini and GPT not only reduce transcription errors but also generalize robustly across scripts, domains, and font variations, marking a substantial leap beyond conventional OCR systems.

**GPT Fine-tuning.** To evaluate the impact of model adaptation, we fine-tuned GPT-4o on a small subset of 500 manually annotated images from the UNB dataset and evaluated it on the remaining 329 images. Despite the limited data, this experiment yielded a 6.13% relative improvement in WER, dropping from 0.330 to 0.269, demonstrating the clear potential of even light fine-tuning for Urdu OCR tasks. These results suggest that with additional annotated data, further improvements are achievable.

#### 4.5. Error Analysis

**Insertion, Deletion and Substitution.** Table 5 presents a breakdown of insertion, deletion, and substitution errors for each LLM on the UNB

dataset. Across all models, we observe that deletion errors dominate, indicating that LLMs frequently omit characters or words during transcription. This trend suggests a conservative decoding bias, models may avoid making uncertain predictions in the presence of degraded or ambiguous visual features. The cursive and ligature-rich structure of Urdu, especially in Nastaliq script, likely contributes to this behavior by making character boundaries harder to resolve.

Among the models, GPT-4o and Llama-4-Scout exhibit the highest deletion counts (47,204 and 42,970, respectively). Llama-4-Maverick also shows a high number of deletions (27,668) alongside a substantial number of substitutions and insertions, indicating a more aggressive decoding strategy that often results in errors across all categories. In contrast, Gemini-2.5-Pro achieves the lowest error counts across all three categories, only 4,947 insertions, 9,341 deletions, and 2,650 substitutions, highlighting its robustness in maintaining both precision and recall. Its ability to more accurately extract the text suggests better visual-language alignment, potentially due to stronger image encoding. Claude-3.7-Sonnet shows high deletion errors (33,805), pointing to underlying limitations in handling densely written cursive text.

Model	Error		
	Ins.	Del.	Sub.
<b>GPT-4.1</b>	13,008	23,627	17,990
<b>GPT-4o</b>	19,490	47,204	28,343
<b>Gemini-2.5-Pro</b>	4,947	9,341	2,650
<b>Claude-3.7-Sonnet</b>	8,902	33,805	16,79
<b>Llama-4-Maverick</b>	20,730	27,668	19,076
<b>Llama-4-Scout</b>	25,001	42,970	24,065

Table 5: Insertion, deletion, and substitution error for the LLMs on UNB dataset.

**Character-Level Analysis.** To further investigate recognition behavior, we conducted a character-level error analysis across all evaluated models. The top five most frequently misrecognized characters were extracted for each model.

Across nearly all models, YEH (ﻱ) and ALEF (ا) consistently rank among the most error-prone

LLM	OpenTI-Naskh		OpenTI-Nastaliq		UNB (Low-Resolution)		UNB (High-Resolution)	
	WER	CER	WER	CER	WER	CER	WER	CER
<b>GPT-4.1</b>	0.286	0.089	0.443	0.203	0.682	0.471	0.254	0.096
<b>GPT-4o</b>	0.418	0.252	0.628	0.432	0.779	0.559	0.327	0.154
<b>Gemini-2.5-Pro</b>	0.228	0.066	0.303	0.125	0.177	0.046	0.133	0.032
<b>Claude-3.7-Sonnet</b>	0.329	0.127	0.616	0.386	Fail	Fail	0.249	0.100
<b>Llama-4-Maverick</b>	0.302	0.104	0.765	0.561	1.036	0.837	0.305	0.128
<b>Llama-4-Scout</b>	0.331	0.131	0.876	0.735	Fail	Fail	0.342	0.151

Table 4: WER before and after passing images through super-resolution model.

characters. These are among the most frequently used letters in Urdu and are especially vulnerable to misrecognition due to their minimal strokes and visual similarity to other letters with simple vertical or curved shapes, such as REH (ر), WAW (و), and even fragments of compound ligatures. For example, ALEF (ا), which often appears as a lone vertical stroke, can be easily confused with REH (ر) in initial forms or with parts of medial ligatures. Similarly, YEH (ي), especially in its final form, closely resembles NOON (ن) and HEH (ه) due to shared looped or tail-like endings. These similarities become even more pronounced in Nastaliq, where slanted positioning, and vertical stacking blur the boundaries between glyphs. As a result, these characters are often either substituted for similar shapes or omitted entirely when the model lacks strong visual disambiguation capabilities.

This trend is most pronounced in GPT-4o, which misrecognized YEH (ي) over 1,911 times and ALEF (ا) 1,763 times, making them the top two sources of error. Llama-4-Scout follows a similar pattern, with 1,793 and 1,468 errors for YEH (ي) and ALEF (ا), respectively. These high counts indicate a recurring inability to resolve minimal-stroke characters, particularly in cluttered ligatures or degraded print. Llama-4-Maverick and Claude-3.7-Sonnet also demonstrate this vulnerability, each showing over 1,000 errors on these same characters, further underscoring the systemic nature of this challenge across transformer-based models. In contrast, Gemini-2.5-Pro exhibits a distinctly different error profile, with far fewer total mistakes. It reports only 306 errors on HEH (ه), followed by WAW (و) at 154, and JEEM (ج) and TTE (ت) just over 100. Interestingly, Gemini also reports far fewer errors on high-frequency characters like YEH (ي) and ALEF (ا).

Additionally, the presence of blank or missing characters, manifesting as empty tokens in the error logs, is particularly prominent in GPT-4o, and Llama-4-Scout. GPT-4o logs over 1,500 such omissions, making it the third most common error character in its output. This correlates strongly with the deletion-heavy behavior observed in earlier analyses. Overall, this character-level breakdown

reveals both script-level complexities (e.g., overlapping glyph shapes, cursive ambiguity) and model-specific failure mode. These findings reinforce the value of Urdu-specific modeling strategies and suggest that meaningful gains in accuracy may depend on enhancing character-level discrimination in both image encoding and token decoding stages.

## 5. Conclusion

Our contributions in this work are threefold, we develop a robust, modular OCR pipeline that integrates YOLOv11 which we fine-tuned on our datasets for article and column segmentation, a SwinIR-based image super-resolution model trained on our high-quality Urdu newspaper scans, and LLM-based text recognition. This end-to-end design effectively transforms noisy, complex Urdu newspaper layouts into clean, machine-readable text while addressing the structural challenges of newspapers. We introduce the UNB dataset, a manually annotated dataset of Urdu newspaper text, providing a new benchmark for OCR. We conduct a comprehensive comparative analysis between traditional OCR systems and modern LLMs, showing that pre-trained large-scale transformer-based models such as, GPT-4, and Gemini-2.5-Pro outperform CNN- and RNN-based baselines across both Naskh and Nastaliq scripts. Our error analysis expose systemic weaknesses in Urdu character modeling, and a small-scale fine-tuning experiment on GPT-4o (using just 500 samples) yields a 6.13% WER reduction, highlighting the potential of LLM adaptation even in low-resource settings. Together, these contributions advance Urdu OCR and highlight the potential of multimodal LLMs for complex, low-resource writing systems.

## 6. Future Work

Looking ahead, we aim to advance our pipeline and expand its applicability in several key directions: (i) evaluate LLMs in few-shot settings to directly compare their performance with zero-shot baselines; (ii) significantly expand our Urdu OCR dataset and fine-tune open-source LLMs such as Gemma-3-4B

and Gemma-3-12B (Gemma, 2025) to build more lightweight and accessible models for downstream deployment; and (iii) extend our framework to additional regional scripts such as Punjabi, Sindhi, and Balochi to assess the generalization capabilities of LLMs across low-resource languages.

## 7. Limitations

Despite the effectiveness of our approach, several limitations remain. First, our evaluation dataset consists of only 829 manually transcribed samples of Urdu newspaper articles. While carefully curated, this relatively small dataset limits the statistical robustness and linguistic diversity of our benchmarks. The labor-intensive nature of manual annotation makes it difficult to scale up to the volume required for training or evaluating. Second, our experiments rely on off-the-shelf LLMs operating in a zero-shot setting. While we demonstrate that even minimal fine-tuning on a small subset of 500 samples can lead to noticeable improvements, broader and more consistent gains would require access to significantly larger volumes of annotated in-domain OCR data. Such data is currently lacking for Urdu, especially in real-world formats like newspapers, highlighting the need for future dataset development efforts. Third, the end-to-end pipeline introduces considerable computational overhead. Each stage (YOLOv11x-based article and column segmentation, SwinIR-based super-resolution, and LLM-based transcription, requires substantial GPU resources and memory. Among these, the inference time of LLMs poses the greatest bottleneck, with high latency and slow throughput making real-time or large-scale batch processing infeasible in low-resource environments

## 8. Bibliographical References

2024. Gemini.

Tobiloba Adenekan. 2024. Advancing text digitization: A comprehensive system and method for optical character recognition.

Saad Bin Ahmed, Saeeda Naz, Salahuddin Swati, and Muhammad Imran Razzak. 2017. Hand-written urdu character recognition using 1-dimensional blstm classifier.

Jaided AI. 2020. EasyOCR: Ready-to-use ocr with 80+ supported languages and growing. <https://github.com/JaidedAI/EasyOCR>. Accessed: [Date you accessed it, e.g., 2025-07-27].

A. Anandhi and Jaiganesh Mahalingam. 2025. An enhanced image restoration using deep learning and transformer based contextual optimization algorithm. *Scientific Reports*, 15.

Anthropic. Claude 3.7 sonnet and claude code.

Asghar Ali Chandio, Md. Asikuzzaman, Mark Pickering, and Mehwish Leghari. 2020. Cursive-text: A comprehensive dataset for end-to-end urdu text recognition in natural scene images. *Data in Brief*, 31:105749.

Victor Davis. 2018. Types, tokens, and hapaxes: A new heap's law.

Gemma. 2025. Gemma 3 technical report.

Israr Khattak. 2017a. Segmentation-free optical character recognition for printed urdu text. *EURASIP Journal on Image and Video Processing*, 62:1–18.

Israr Khattak. 2017b. Segmentation-free optical character recognition for printed urdu text. *EURASIP Journal on Image and Video Processing*, 62:1–18.

Benjamin Kiessling. 2020. Kraken ocr. <https://github.com/mittagessen/kraken>. Accessed: 2025-07-27.

Kibaek Kim, Yoon Kim, and Young-Joo Kim. 2025. Hybrid frequency–spatial domain learning for image restoration in under-display camera systems using augmented virtual big data generated by the angular spectrum method. *Applied Sciences*, 15(1).

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.

Jingyun Liang, Jiezhong Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. 2021. Swinir: Image restoration using swin transformer. *arXiv preprint arXiv:2108.10257*.

Mona Lisa and Hew Bot. 2017. My Research Software.

Philip M McCarthy and Scott Jarvis. 2010. MTL, vocd-d, and HD-D: a validation study of sophisticated approaches to lexical diversity assessment. *Behav. Res. Methods*, 42(2):381–392.

Meta. The llama 4 herd: The beginning of a new era of natively multimodal ai innovation.

Saeeda Naz, Arif Iqbal Umar, Riaz Ahmed, Muhammad Imran Razzak, Sheikh Faisal Rashid, and Faisal Shafait. 2016. Urdu nasta'liq text recognition using implicit segmentation based on multi-dimensional long short term memory neural networks. *SpringerPlus*, 5(1):2010.

OpenAI. 2024. [Gpt-4o system card](#).

Abdur Rahman, Arjun Ghosh, and Chetan Arora. 2023. *UTRNet: High-Resolution Urdu Text Recognition in Printed Documents*, page 305–324. Springer Nature Switzerland.

RDRR. MATTR: Lexical diversity: Moving-Average Type-Token Ratio (MATTR) in koRpus: Text Analysis with Emphasis on POS Tagging, Readability, and Lexical Diversity — rdr.io. <https://rdr.io/cran/koRpus/man/MATTR.html#heading-5>. [Accessed 30-09-2025].

Nazly Sabbour and Faisal Shafait. 2013. [A segmentation free approach to arabic and urdu ocr](#). *Proceedings of SPIE - The International Society for Optical Engineering*, 8658.

Safia Shabbir and Imran Siddiqi. 2016. [Optical character recognition system for urdu words in nastaliq font](#). *International Journal of Advanced Computer Science and Applications*, 7(5).

Ray Smith. 2007. [An overview of the tesseract ocr engine](#). In *ICDAR '07: Proceedings of the Ninth International Conference on Document Analysis and Recognition*, pages 629–633, Washington, DC, USA. IEEE Computer Society.

Nawaz Tabassam, Syed Naqvi, Habib Rehman, and Faiz Anoshia. 2009. Optical character recognition system for urdu (naskh font) using pattern matching technique. *International Journal of Image Processing*, 3.

Xintong Wang, Jingheng Pan, Liang Ding, and Chris Biemann. 2024. [Mitigating hallucinations in large vision-language models with instruction contrastive decoding](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15840–15853, Bangkok, Thailand. Association for Computational Linguistics.

Urooba Zaki, Dil Hakro, Khalil-Ur-Rehman Khoubati, Muhammad Zaki, and Maryam Hameed. 2019. Issues & challenges in urdu ocr. 3:43 – 49.