# InkSight: Towards AI-Aided Historical Manuscript Analysis

**Andrey Sakhovskiy**[1,5*], **Ivan Ulitin**[1,3*], **Emilia Bojarskaja**[1,4*],
**Vladimir Kokh**[1], **Ruslan Murtazin**[1], **Maxim Novopoltsev**[1], **Semen Budenny**[1,2]
[1]Sber AI [2]AIRI [3]Perm State University [4]AI Talent Hub [5]Skoltech
**Correspondence:** andrey.sakhovskiy@gmail.com

## Abstract

Large-scale scientific research on historical documents — particularly medieval Arabic manuscripts — remains challenging due to the need for advanced paleographic and linguistic training, the large volume of hand-written materials, and the absence of assisting software. In this paper, we propose **InkSight**, the first end-to-end Arabic manuscript analysis tool for manuscript-based analytics and research hypothesis testing. *InkSight* integrates three key components: (i) an Optical Character Recognition (OCR) module utilizing a Large Visual Language Model (LVLM); (ii) a lightweight document indexing and information retrieval module that enables query-based evidence retrieval from book-length manuscripts; and (iii) a flexible Large Language Model (LLM) prompting interface factually grounded to the given manuscript via Retrieval-Augmented Generation (RAG). Empirical evaluation on the existing KITAB OCR benchmark and our in-house dataset of ancient Arabic manuscripts has revealed that historical research can be effectively supported using smaller fine-tuned LVLMs without relying on larger proprietary models. The live web demo for InkSight is available freely at: https://inksight.ru and the source code for *InkSight* is publicly available at Github[1].

## 1 Introduction

Recent advances in Optical Character Recognition (OCR) (JaidedAI, 2020; Heakl et al., 2025b) and Natural Language Processing (NLP), particularly with Retrieval-Augmented Generation (RAG) (Lewis et al., 2020b) approach, now enable systems to answer complex questions by retrieving and analyzing supporting evidence passages from long documents. Despite these technological advances, large-scale scholarly analysis of historical documents — particularly medieval Arabic manuscripts — continues to face significant practical barriers. Many archival collections remain underexplored due to a critical shortage of specialists capable of accurately transcribing historical scripts. Scholars often spend excessive time on mechanical transcription tasks, with a complex manuscript page requiring up to three days of work even for experienced researchers. In our work, we address this challenge by developing an integrated *InkSight* tool that overcomes the OCR bottleneck while providing historians with an efficient LLM framework for book-length manuscript analysis, hypothesis testing, and evidence-based historical interpretation.

The past few decades have witnessed a notable improvement in handwritten text recognition (HTR) due to OCR methods that adopt either Convolutional Neural Networks (CNN) (Lecun et al., 1998; Wigington et al., 2018; Fasha et al., 2020; JaidedAI, 2020; Bhunia et al., 2021) or task-specific Transformer models (Li et al., 2023). However, these models typically require task-specific fine-tuning and exhibit limited generalization abilities to new domains and fonts. Additionally, historical Arabic manuscripts exhibit high script variability. In contrast to prior OCR approaches, Large Vision-Language Models (LVLMs) (Bai et al., 2023) are capable of surpassing task-specific OCR models in accuracy and generalization without extensive fine-tuning (Heakl et al., 2025b).

Recently, Large Language Models (LLMs) have excelled at fine-grained analysis of long-form texts. Specifically, LLMs enhanced with RAG were shown to have good fact checking capabilities not only for English texts (Min et al., 2023; Liu et al., 2025), but also for Arabic data (Kim et al., 2024; Shafayat et al., 2024). For question answering, current state-of-the-art LLMs exhibit near-human performance (Abdelali et al., 2024) on numerous Arabic datasets (Mozannar et al., 2019; Lewis et al.,

*These authors contributed equally to this work.
[1]https://github.com/ds-hub-sochi/InkSight-tool



*Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics*
*Volume 3: System Demonstrations*, pages 271–281
March 24-29, 2026 

2020a; Artetxe et al., 2020; Clark et al., 2020).

In this paper, we present *InkSight*, the first end-to-end Arabic manuscript analysis tool that adopts Large Visual Language Models (LVLMs) for image-to-text document transcription, hypothesis testing, and evidence retrieval. As seen from Figure 1, InkSight integrates three key components: (i) an Optical Character Recognition (OCR) module utilizing a LVLM; (ii) a Retrieval-Augmented Generation (RAG) module designed to efficiently index and retrieve information from book-length manuscripts; and (iii) a flexible prompting interface, allowing domain experts to formulate custom analytical queries and hypotheses.

The contributions of our paper are as follows:

- We present *InkSight*, an open-access web tool that accepts a handwritten Arabic book, performs *OCR* and document indexing, and supports arbitrary comprehension queries. User questions are answered by an *LLM* enhanced with *RAG* module, which retrieves relevant passages from the book and returns them explicitly as supporting evidence.

- Experimental evaluation on modern KITAB OCR benchmark (Heakl et al., 2025b) and our in-house MAS corpus of historical manuscripts has revealed that large proprietary LLMs, namely `GPT-4o`, `GPT-5` and `Gemini-2.0 Flash`, have limited zero-shot generalization to ancient Arabic texts.

- *InkSight*'s modular pipeline enables easy adaptation to historical manuscripts in other languages. We make the source code for our tool publicly available: `https://github.com/ds-hub-sochi/InkSight-tool`.

## 2 InkSight System

### 2.1 Handwritten Text Recognition Pipeline

For HTR, we implement a two-step pipeline that performs *line segmentation* followed by *line-level transcription* using a fine-tuned LVLM. Line level processing minimizes layout complexity and interference while maintaining adequate context for LVLM autoregressive capabilities (Younes and Abdellah, 2015; Chan et al., 2024).

**Task Formulation** The Handwritten Text Recognition (HTR) task aims to transcribe handwritten text into a machine-interpretable format. Formally, given an input image $X \in \mathbb{R}^{H \times W \times C}$ of a handwritten document (with $C = 1$ for grayscale or $C = 3$ for RGB), the objective is to predict the corresponding character sequence $Y = (y_1, y_2, ..., y_m)$ of length m. When implemented with LVLMs, HTR performs conditional generation by jointly using the image $X$ and a textual prompt $P$ that specifies the recognition task, producing the transcription $Y$ as output.

**Line Segmentation** For line segmentation, we adopt the Kraken toolkit[2] (Kiessling, 2019). First, the input image $X_{page} \in \mathbb{R}^{H \times W \times C}$ undergoes binarization to enhance text-background contrast and reduce noise. The rule-based segmenter then processes this binarized image to extract text line regions $L = \{l_1, l_2, ..., l_k\}$, where k is the number of detected lines, providing robust handling of historical document layouts.

**Segmentation Post-Processing** After initial page segmentation with Kraken, we apply a four-step geometry- and content-aware post-processing pipeline to refine text-line extraction for Arabic handwriting, addressing common artifacts such as over-segmentation and redundant detections:

1. **Dominance-based Box Merging** Given an initial set of text line bounding boxes $\mathcal{B} = \{b_i\}_{i=1}^{N}$ extracted by Kraken, we apply padding with ratio $\gamma_{\text{p}}$. For each pair of boxes $(b_i, b_j)$, we compute their padded intersection $I_{\text{p}} = \text{Area}(b_i^p \cap b_j^p)$. If $I_{\text{p}} > 0$ and $\frac{\text{Area}(b_i)}{\text{Area}(b_j)} > \alpha_{\text{dominance}}$, the smaller box is designated as a fragment and merged with the dominant anchor box. This yields a refined set $\mathcal{B}_{\text{merged}} \subseteq \mathcal{B}$.

2. **Overlap-based Filtering** For each box $b_i \in \mathcal{B}_{\text{merged}}$, we compute the total overlap ratio: $r_i = \frac{\sum_{j \neq i} \text{Area}(b_i \cap b_j)}{\text{Area}(b_i)}$. Boxes exceeding the overlap threshold $\tau_{\text{overlap}}$ are discarded, producing $\mathcal{B}_{\text{filtered}} = \{b_i \mid r_i \leq \tau_{\text{overlap}}\}$.

3. **Image-based Content Validation** In the third step, each remaining candidate box is converted to image coordinates and its region of interest is extracted. We then validate the presence of handwriting using multiple image-based criteria, including: ink density $\rho \in [\rho_{\min}, \rho_{\max}]$; minimum contour count $c_{\min}$; minimum contour area $a_{\min}$; minimum

[2] https://kraken.re

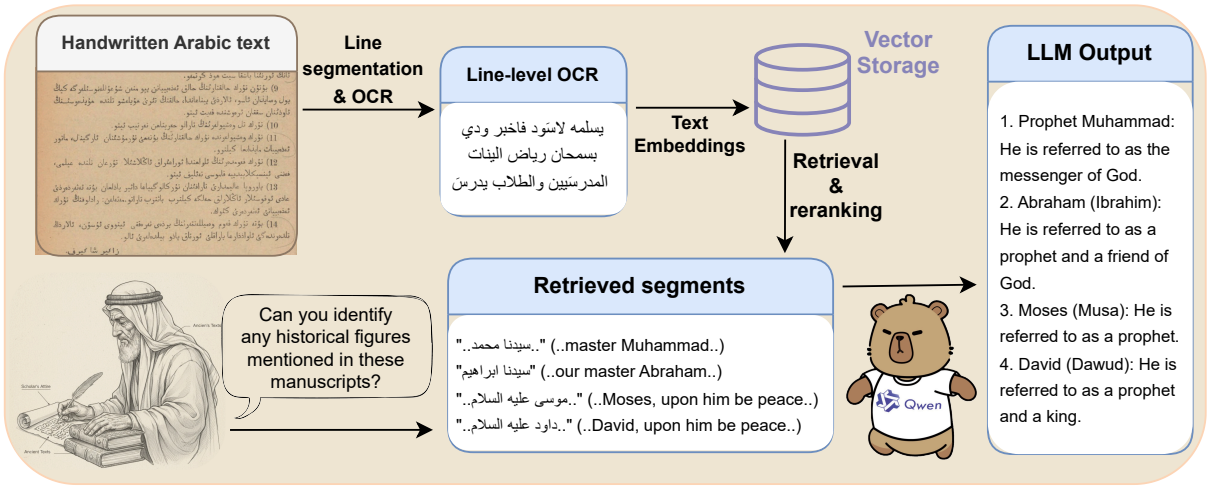


Figure 1: An overview of the InkSight tool for historical Arabic manuscript analysis. The system processes manuscript images through (1) a fine-tuned LVLM-based OCR pipeline with specialized post-processing, (2) a RAG module for semantic indexing of book-length documents, and (3) LLM-based chat interface for queries related to document analysis.

hole ratio $\eta_{\min}$; and minimum average defects per contour $\delta_{\min}$. Only boxes satisfying all criteria are retained.

4. **Horizontal Row Aggregation** Bounding boxes are iteratively merged into horizontal rows when three conditions are satisfied: their horizontal projections overlap, their heights differ by less than a threshold $\zeta_{\text{height}}$, and their vertical overlap ratio exceeds a minimum value $\kappa_{\text{overlap}}$.

The key post-processing parameters are presented in Table 4 of Appx. C.

## 2.2 OCR Model

After line segmentation, each segmented line image is processed using our pretrained Qwen3-VL-8B (Team, 2025) model queried with a task-specific prompt (see Appendix A), which directs the model to produce an Arabic transcription of line content.

## 2.3 Training Datasets

To train our OCR model, we adopt annotated data from three Arabic OCR datasets: (i) Muharaf (Saeed et al., 2024a), (ii) SARD (Nacar et al., 2025), and (iii) *MAS*, our in-house corpus of medieval Arabic manuscripts. The overview of the three training datasets is presented in Table 1.

MUHARAF (Saeed et al., 2024b) (Manuscripts of Handwritten Arabic) is a dataset of manuscripts spanning from the early 19th to the 21st century. It comprises 1,644 authentic document page images containing 36,311 annotated text lines in total. The corpus covers various document types, including personal correspondence, legal records, and literary fragments.

| Feature | Muharaf | SARD | MAS |
|---|---|---|---|
| Data Type | Historical manuscripts | Synthetic | Historical manuscripts |
| # Pages | 1,644 | 843,622 | 1,023 |
| # lines/words | 36,311 lines | 690M words | 11,841 lines |
| Fonts/Scripts | Predominantly Ruq'ah script | 10 fonts | 3 scripts |
| Centuries | 19th–21st | 21st | 12th–19th |

Table 1: Comparison of Muharaf, SARD, MAS Datasets.

SARD (Nacar et al., 2025) (Synthetic Arabic Recognition Dataset) is a large-scale corpus of synthetically generated document pages designed to simulate book-style Arabic documents. In total, SARD has 843,622 document images with approximately 690 million words, rendered in ten distinct Arabic fonts to ensure wide typographic diversity.

MAS (**M**edieval **A**rabic **S**cript) is our in-house collection of manuscripts sourced from the archives of the Abu Rayhan Biruni Institute of Oriental Studies in Uzbekistan. This corpus includes 1,023 pages containing 11,841 annotated text lines from original manuscripts and rewritten copies of the authentic documents spanning the period from 12th to 19th centuries. MAS documents represent Arabic calligraphy in various styles including Naskh, Nastaliq, and Taliq, preserving the natural varia-

| Dataset | Train | Test | Total |
|---|---|---|---|
| MUHARAF | 24,495 | | 24,495 |
| SARD | 100,000 | | 100,000 |
| Amiri font | 20,000 | | |
| Arial font | 20,000 | | |
| Calibri font | 20,000 | | |
| Sakkal Majalla font | 20,000 | | |
| Scheherazade new font | 20,000 | | |
| MAS | 10,658 | 1,183 | 11,841 |
| Naskh script | 1,726 | 192 | 1,918 |
| Taliq script | 3,785 | 420 | 4,205 |
| Nastaliq script | 5,147 | 571 | 5,718 |
| KITAB benchmark | | 4,138 | 4,138 |

Table 2: OCR data statistics in terms of line count.

tions, imperfections, and layout complexities found in historical manuscripts. The dataset covers diverse domains, including waqf documents (testamentary acts of property donation to religious institutions), yarliqs (khan decrees and firmans), arizas (petitions), cheks (legal receipts and certificates), shajars (genealogical tables), and announcements (public proclamations and charters). Due to calligraphy variability and the linguistic gap between Medieval and Modern Arabic, the annotation of the manuscripts is labour-intensive. Each page is annotated by a single expert in Arabic calligraphy only. Notably, the archives of the Abu Rayhan Biruni Institute comprises over 26,000 manuscript volumes dating from the 9th to the 20th centuries with the majority of the documents not yet digitized.

**Data Statistics** Although SARD provides over 800k documents, we subsampled 100k for training as preliminary experiments did not show further error rates decrease from training on more synthetic data. Specifically, we took 20k for each of the 5 most common fonts. Annotated lines from the MAS dataset are randomly split into train and test sets, respectively (see Table 2). Each split preserves the proportions the three calligraphy styles, namely Naskh, Nastaliq, Taliq, ensuring a balanced representation across all subsets. The summarized statistics for the OCR training and evaluation data are shown in Table 2.

### 2.4 Training Details

**Training Setup** We applied domain-specific adaptations to the `Qwen2.5-VL-7B-Instruct`[3] and `Qwen3-VL-8B-Instruct`[4] models for the Arabic language. For this stage, we employed LoRA adapters (Hu et al., 2022) (r=8) for parameter-efficient fine-tuning with optional bf16 quantization to reduce memory usage.

### 2.5 Document Search Index

Efficient document retrieval is the core component of the's RAG- and LLM-based analytical module based on RAG and LLM InkSight. The pages recognized by the HTR module serve as evidence for answering a historician's query.

**Text Chunking** Due to variability of page lengths recognized by OCR, each full page text is segmented into fixed-size textual chunks (passages) prior to embedding. For segmentation, we employ a deterministic sliding-window procedure with a chunk length up to $L = 1000$ characters and a window size and chunk overlap of $O = 200$. Pages shorter than $L$ produce a single chunk. To avoid mid-sentence fragmentation, chunk boundaries are adjusted to the nearest separator (e.g., sentence punctuation, paragraph breaks, or word spaces). This ensures chunks to align with linguistic units while maintaining consistent size.

**Passage Embedding** Each chunk is encoded into a dense semantic vector using the multilingual BERT-based encoder[5] (Devlin et al., 2019) trained on the MS Marco dataset (Nguyen et al., 2016). Although there exist models with better retrieval quality for Arabic (Al-Rasheed et al., 2025), we selected the model for its balanced trade-off between representation quality and computational efficiency. All embeddings are stored in a ChromaDB[6] vector index, configured for cosine similarity search. The indexing pipeline is implemented using LangChain[7], enabling easy adaptation to other domains and languages thanks to its modular pipeline. Our source code is publicly available (Sec. 1), allowing users to change the retrieval model by changing a single line in the configuration file.

### 2.6 Document Analysis

**Passage Retrieval** The retrieval is performed in two stages: dense vector retrieval followed by optional cross-encoder reranking. Given a natural language query $q$, the system first encodes it into

[3] hf.co/Qwen/Qwen2.5-VL-7B-Instruct
[4] hf.co/Qwen/Qwen3-VL-8B-Instruct
[5] hf.co/amberoad/bert-multilingual-passage-reranking-msmarco
[6] https://github.com/chroma-core/chroma
[7] https://www.langchain.com/

a dense embedding using the same used for indexing. Then the cosine similarity between $q$ and each indexed chunk $c$ is computed as:

$$\text{sim}(q, c) = \frac{\langle f(q), g(c) \rangle}{|f(q)||g(c_i)|} \quad (1)$$

The top-k most similar chunks (with $k = 4$ by default) are retrieved as evidence supporting the input query. Only chunks exceeding a minimum similarity threshold are considered, ensuring low-confidence matches are discarded early. For retrieval and reranking, we adopt the same encoder model.

The final set of retrieved passages is passed to the generative reasoning component, implemented as the `Qwen3.5-397B-A17B` LLM[8] accessed via OpenRouter[9]. The LLM is prompted with a user query concatenated with the retrieval textual passages. Thus, the RAG module functions as the evidence retrieval mechanism that ensures generated LLM responses to be grounded to and factually aligned with the studied manuscript. For OCR and manuscript analysis prompts, please see Appx. A.

Overall, the InkSight's architecture provides a robust and reproducible framework for manuscript-based in historical research. All modules may be updated with more advanced models, e.g., LVLM, LLM, and retriever, allowing seamless adaptation to manuscripts in other languages as well.

## 3 Evaluation

### 3.1 Evaluation Data

To assess the quality of InkSightOCR, we performed evaluation on (i) OCR part of the KITAB benchmark and (ii) 589 lines from the MAS's test part. While KITAB covers more modern texts (starting from the 19th century), MAS includes calligraphic documents from 12th-19th centuries. Thus, our evaluation explores how well modern LVLMs generalize to Arabic language and visual stylistic variations.

*KITAB* OCR benchmark (Heakl et al., 2025b) is a collection of 4,138 samples across multiple document types, including historical manuscripts, handwritten texts, and printed documents for Arabic text recognition. It integrates data from established Arabic OCR datasets such as KHATT (Mahmoud et al., 2014), ADAB (Boubaker et al., 2021), Muharaf (Saeed et al., 2024a), and EvAREST (Hassan et al., 2021), covering diverse writing styles, fonts, and document complexities that present authentic challenges for Arabic script recognition.

| | KITAB | | MAS | |
|---|---|---|---|---|
| **Model** | **CER** | **WER** | **CER** | **WER** |
| **Closed-Source LVLMs** | | | | |
| GPT-5 | — | — | .52 | .91 |
| GPT-4o | .31 | .55 | .51 | .84 |
| Gemini-2.0 Flash | **.13** | .32 | .55 | .80 |
| **Task-Specific Models** | | | | |
| Tesseract | .54 | .84 | — | — |
| EasyOCR | .58 | .89 | — | — |
| Surya | 4.95 | 5.91 | — | — |
| **Open-Source LVLMs** | | | | |
| Qwen2-VL-7b | 1.48 | 1.55 | 4.35 | 3.55 |
| Qwen2.5-VL-7b | 1.2 | 1.4 | .89 | 1.14 |
| Qwen2.5-VL-7b +sft | .43 | .67 | .34 | .73 |
| Qwen-3VL-8b | .67 | .95 | .94 | 1.23 |
| Qwen-3VL-8b +sft | .54 | .80 | **.24** | .60 |
| AIN-7b | .20 | **.28** | .34 | **.53** |

Table 3: Evaluation results on Arabic KITAB OCR benchmark and our in-house MAS corpus of Arabic manuscripts. Smaller values are better. The best results for each metric are highlighted in **bold**.

### 3.2 Evaluation Setup

**Baselines** Following prior research on Arabic OCR on the KITAB benchmark (Heakl et al., 2025b), we evaluate our *InkSight* model against both LVLMs and prior task-specific OCR approaches. For LVLM evaluation, we compare against open-source models, namely `Qwen2-VL-7B` (Wang et al., 2024), `Qwen2.5-VL-7B` (Team, 2025), `Qwen3-VL-8B`, and the `AIN-7B` (Heakl et al., 2025a) as well as proprietary GPT-4o (OpenAI et al., 2024) and `Gemini-2.0 Flash`[10] (Team et al., 2024) models. Among task-specific OCR models, we adopted `Tesseract` (Smith, 2007), `EasyOCR` (JaidedAI, 2020), and `Surya` (Paruchuri and Team, 2025).

**OCR finetuning Details** We configured Qwen2.5-VL with temperature $= 1 \times 10^{-6}$ to obtain stable, low-variance transcription

[8] http://hf.co/Qwen/Qwen3.5-397B-A17

[9] `https://openrouter.ai`

[10] cloud.google.com/vertex-ai/generative-ai/docs/models/gemini/2-0-flash

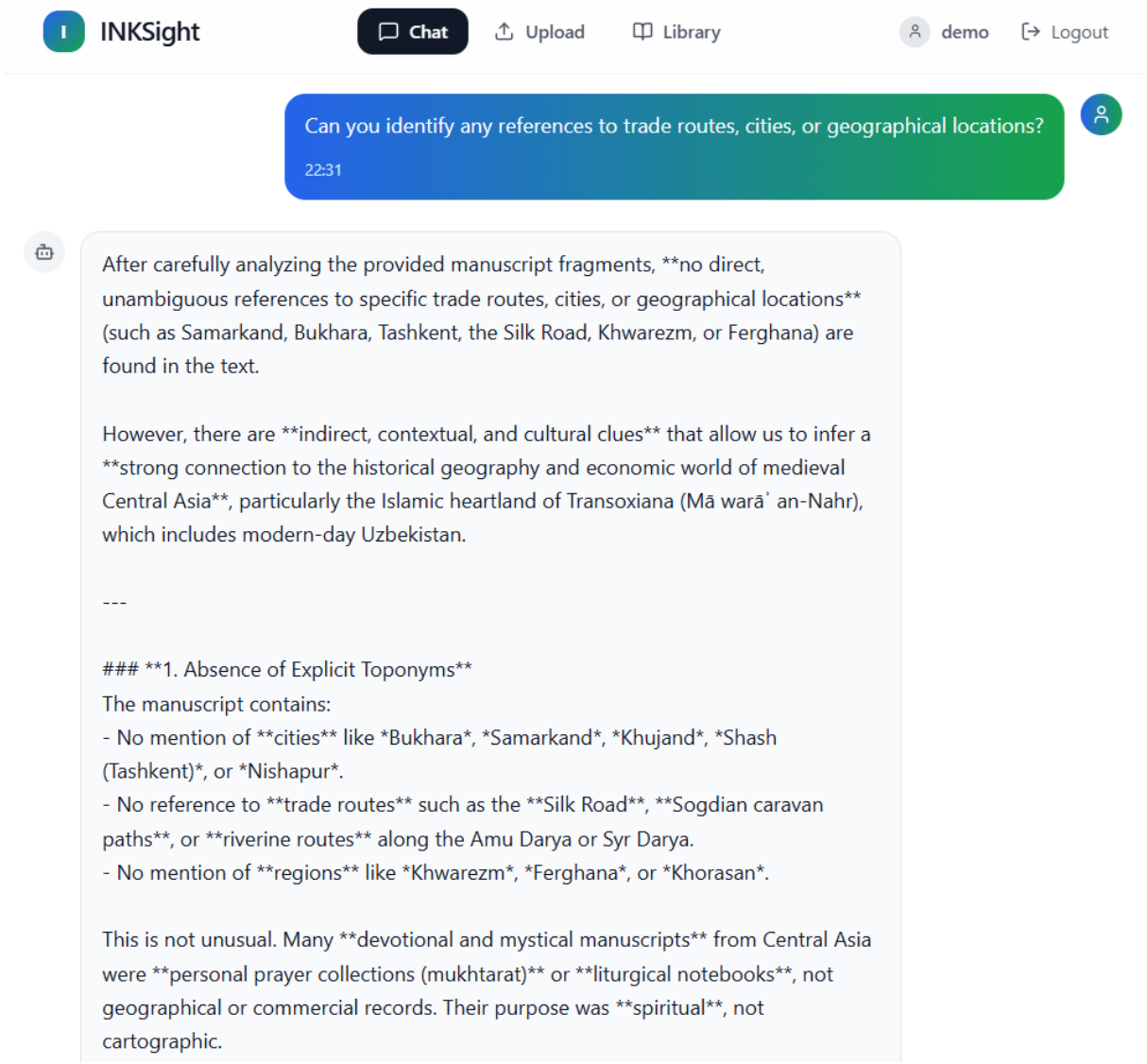


Figure 2: InkSight's output example for the query using the evidence from handwritten *Dala'il al-Khayrat* book.

predictions. Fine-tuning was conducted for 4 epochs with a batch size of 8, utilizing the Adam optimizer ($\beta_1 = 0.9, \beta_2 = 0.9999$), a learning rate of $5 \times 10^{-4}$, a weight decay of 0.02, and gradient clipping with a maximum norm of 0.5 to enhance stability. During training, we employed a cosine learning rate schedule with a warmup ratio of 0.05, linearly increasing the learning rate over the first 5% of steps before transitioning to cosine decay.

**Metrics** We evaluate OCR quality using two standard metrics widely used in OCR research (Saeed et al., 2024a; Dolek and Kurt, 2025): Character Error Rate (CER) and Word Error Rate (WER). Both rely on the Levenshtein edit distance (Levenshtein, 1966) and quantify the number of substitutions (S), insertions (I), and deletions (D) required to transform the predicted text into the ground truth.

$$\text{CER} = \frac{S + I + D}{N_{\text{char}}}; \tag{2}$$

$$\text{WER} = \frac{S + I + D}{N_{\text{word}}} \tag{3}$$

The combination of CER and WER therefore offers a balanced assessment of both character-level recognition quality and semantic correctness.

## 4 Results

**Proprietary Models Struggle with Manuscripts** OCR evaluation results are presented in Table 3. While strong proprietary `GPT-4o` and `Gemini-2.0 Flash` models show low *CER* and *WER* on KITAB, they fall short of smaller open-source fine-tuned `Qwen` models on ancient manuscripts from MAS corpus. Thus, we use finetuned `Qwen-3VL-8b` in

InkSight as it has the lowest CER on MAS. However, `AIN-7b` and `Qwen2.5-VL-7b+sft` show similar performance and could also be used for OCR on Arabic manuscripts.

**Pretraining is Not Essential for Historical HTR** Despite undergoing full model pretraining on authentic Arabic texts, `AIN-7b` achieves,comparable character error rates to fine-tuned `Qwen2.5VL-7b` `Qwen3VL-8b` with the latter showing even smaller *CER* (0.24 vs 0.34) on MAS. This indicates that language-specific pretraining is unnecessary for historical Arabic HTR when leveraging parameter-efficient adaptation of multilingual LVLMs. Our results indicate that synthetic data paired with lightweight fine-tuning can enable historical manuscript digitization for low-resource non-English languages without costly pretraining.

**Case Study: Dala'il al-Khayrat** To demonstrate InkSight's capabilities in real-world historical research, we conducted an analysis on a XVII century copy of *Dala'il al-Khayrat* (Guidelines to Goodness) manuscript, a seminal Islamic devotional text composed by the Moroccan scholar Muhammad al-Jazuli (who died in 1465). The 186 pages of the manuscript were segmented into 3,720 lines by the InkSight's OCR component. The InkSight output for an example query on trade routes mention is shown in Figure 2. From the example, InkSight allows a researcher to test a hypothesis (e.g., the given book to mention any trade routes) in seconds.

## 5 Conclusion

In this work, we presented InkSight, the first end-to-end AI-aided system for historical Arabic manuscript analysis that integrates LVLM-based OCR, semantic search via RAG, and an expert-oriented prompting interface. Our evaluation demonstrates that appropriately fine-tuned open-source LVLMs can outperform larger proprietary models like GPT-4o, GPT-5, and Gemini-2.0 Flash on historical document analysis tasks. The system directly addresses critical bottlenecks in historical research workflows by automating transcription and indexing processes, enabling scholars to focus on higher-value semantic and historical analysis rather than mechanical transcription. The proposed system design can be adopted to automate historical research in other domains and languages.

## Acknowledgments

The authors thank Alexei Rastorguev for his invaluable assistance with the deployment and maintenance of the web demo.

## References

Ahmed Abdelali, Hamdy Mubarak, Shammur Chowdhury, Maram Hasanain, Basel Mousi, Sabri Boughorbel, Samir Abdaljalil, Yassine El Kheir, Daniel Izham, Fahim Dalvi, Majd Hawasly, Nizi Nazar, Youssef Elshahawy, Ahmed Ali, Nadir Durrani, Natasa Milic-Frayling, and Firoj Alam. 2024. LAraBench: Benchmarking Arabic AI with large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 487–520, St. Julian's, Malta. Association for Computational Linguistics.

Raghad Al-Rasheed, Abdullah Al Muaddi, Hawra Aljasim, Rawan Al-Matham, Muneera Alhoshan, Asma Al Wazrah, and Abdulrahman AlOsaimy. 2025. Evaluating RAG pipelines for Arabic lexical information retrieval: A comparative study of embedding and generation models. In *Proceedings of the 1st Workshop on NLP for Languages Using Arabic Script*, pages 155–164, Abu Dhabi, UAE. Association for Computational Linguistics.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. On the cross-lingual transferability of monolingual representations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4623–4637, Online. Association for Computational Linguistics.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A versatile vision-language model for understanding, localization, text reading, and beyond. *Preprint*, arXiv:2308.12966.

Ayan Kumar Bhunia, Shuvozit Ghose, Amandeep Kumar, Pinaki Nath Chowdhury, Aneeshan Sain, and Yi-Zhe Song. 2021. Metahtr: Towards writer-adaptive handwritten text recognition. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15825–15834.

Houcine Boubaker, Abdelkarim Elbaati, Najiba Tagougui, Haikal El Abed, Monji Kherallah, Volker Märgner, and Adel M. Alimi. 2021. Adab database.

Adrian Chan, Anupam Mijar, Mehreen Saeed, Chau-Wai Wong, and Akram Khater. 2024. Hatformer: Historic handwritten arabic text recognition with transformers. *arXiv preprint arXiv:2410.02179*.

Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. TyDi QA: A benchmark

for information-seeking question answering in typologically diverse languages. *Transactions of the Association for Computational Linguistics*, 8:454–470.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. pages 4171–4186, Minneapolis, Minnesota.

Ishak Dolek and Atakan Kurt. 2025. Ottoman htr: Recognition of the ottoman riqa font using deep learning models. *2025 Innovations in Intelligent Systems and Applications Conference (ASYU)*, pages 1–6.

Mohammad Fasha, Bassam Hammo, Nadim Obeid, and Jabir Widian. 2020. A hybrid deep learning model for arabic text recognition. *CoRR*, abs/2009.01987.

Heba Hassan, Ahmed El-Mahdy, and Mohamed E. Hussein. 2021. Arabic scene text recognition in the deep learning era: Analysis on a novel dataset. *IEEE Access*, 9:107046–107058.

Ahmed Heakl, Sara Ghaboura, Omkar Thawakar, Fahad Shahbaz Khan, Hisham Cholakkal, Rao Muhammad Anwer, and Salman H. Khan. 2025a. AIN: the arabic inclusive large multimodal model. *CoRR*, abs/2502.00094.

Ahmed Heakl, Muhammad Abdullah Sohail, Mukul Ranjan, Rania Elbadry, Ghazi Shazan Ahmad, Mohamed El-Geish, Omar Maher, Zhiqiang Shen, Fahad Shahbaz Khan, and Salman Khan. 2025b. KITAB-bench: A comprehensive multi-domain benchmark for Arabic OCR and document understanding. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 22006–22024, Vienna, Austria. Association for Computational Linguistics.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. Lora: Low-rank adaptation of large language models. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net.

JaidedAI. 2020. Easyocr. `https://github.com/JaidedAI/EasyOCR`. GitHub repository.

Benjamin Kiessling. 2019. Kraken - A Universal Text Recognizer for the Humanities. In *Digital Humanities 2019*, Utrecht, Netherlands.

Vu Trong Kim, Michael Krumdick, Varshini Reddy, Franck Dernoncourt, and Viet Dac Lai. 2024. An analysis of multilingual FActScore. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4309–4333, Miami, Florida, USA. Association for Computational Linguistics.

Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10:707.

Patrick Lewis, Barlas Oguz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020a. MLQA: Evaluating cross-lingual extractive question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7315–7330, Online. Association for Computational Linguistics.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020b. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Minghao Li, Tengchao Lv, Jingye Chen, Lei Cui, Yijuan Lu, Dinei A. F. Florêncio, Cha Zhang, Zhoujun Li, and Furu Wei. 2023. Trocr: Transformer-based optical character recognition with pre-trained models. In *Thirty-Seventh AAAI Conference on Artificial Intelligence, AAAI 2023, Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence, IAAI 2023, Thirteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2023, Washington, DC, USA, February 7-14, 2023*, pages 13094–13102. AAAI Press.

Xin Liu, Lechen Zhang, Sheza Munir, Yiyang Gu, and Lu Wang. 2025. VeriFact: Enhancing long-form factuality evaluation with refined fact extraction and reference facts. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 17919–17936, Suzhou, China. Association for Computational Linguistics.

Sabri A. Mahmoud, Irfan Ahmad, Wasfi G. Al-Khatib, Mohammad Alshayeb, Mohammad Tanvir Parvez, Volker Märgner, and Gernot A. Fink. 2014. Khatt: An open arabic offline handwritten text database. *Pattern Recognition*, 47(3):1096–1112. Handwriting Recognition and other PR Applications.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Hussein Mozannar, Elie Maamary, Karl El Hajal, and Hazem Hajj. 2019. Neural Arabic question answering. In *Proceedings of the Fourth Arabic Natural Language Processing Workshop*, pages 108–118, Florence, Italy. Association for Computational Linguistics.

Omer Nacar, Yasser Al-Habashi, Serry Sibaee, Adel Ammar, and Wadii Boulila. 2025. Sard: A large-scale synthetic arabic ocr dataset for book-style text recognition. *Preprint*, arXiv:2505.24600.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

Vikas Paruchuri and Datalab Team. 2025. Surya: A lightweight document ocr and analysis toolkit. https://github.com/VikParuchuri/surya. GitHub repository.

Mehreen Saeed, Adrian Chan, Anupam Mijar, Joseph Moukarzel, Georges Habchi, Carlos Younes, Amin Elias, Chau-Wai Wong, and Akram Khater. 2024a. Muharaf: Manuscripts of handwritten arabic dataset for cursive text recognition. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*.

Mehreen Saeed, Adrian Chan, Anupam Mijar, Joseph Moukarzel, Georges Habchi, Carlos Younes, Amin Elias, Chau-Wai Wong, and Akram Khater. 2024b. Muharaf: Manuscripts of handwritten arabic dataset for cursive text recognition.

Sheikh Shafayat, Eunsu Kim, Juhyun Oh, and Alice Oh. 2024. Multi-fact: Assessing multilingual llms' multi-regional knowledge using factscore. *Preprint*, arXiv:2402.18045.

R. Smith. 2007. An overview of the tesseract OCR engine. In *9th International Conference on Document Analysis and Recognition (ICDAR 2007), 23-26 September, Curitiba, Paraná, Brazil*, pages 629–633. IEEE Computer Society.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, Soroosh Mariooryad, Yifan Ding, Xinyang Geng, Fred Alcober, Roy Frostig, Mark Omernick, Lexi Walker, Cosmin Paduraru, Christina Sorokin, and 1118 others. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *Preprint*, arXiv:2403.05530.

Qwen Team. 2025. Qwen2.5-vl.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Yang Fan, Kai Dang, Mengfei Du, Xuancheng Ren, Rui Men, Dayiheng Liu, Chang Zhou, Jingren Zhou, and Junyang Lin. 2024. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *Preprint*, arXiv:2409.12191.

Curtis Wigington, Chris Tensmeyer, Brian L. Davis, William A. Barrett, Brian L. Price, and Scott Cohen. 2018. Start, follow, read: End-to-end full-page handwriting recognition. In *Computer Vision - ECCV 2018 - 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part VI*, volume 11210 of *Lecture Notes in Computer Science*, pages 372–388. Springer.

Mokhtari Younes and Yousfi Abdellah. 2015. Segmentation of arabic handwritten text to lines. *Procedia Computer Science*, 73:115–121.

## A Prompts

The manuscript analysis prompt, presented in Figure 4, implements a structured reasoning framework that forces LLM-based analysis to be grounded to the provided retrieved evidence from the indexed manuscript. This two-stage protocol — first retrieving relevant manuscript passages via the search_knowledge tool before providing contextualized analysis — directly addresses the hallucination problem common in LLM applications in general.

For OCR benchmark evaluation, we employ two baseline prompts: Figure 5 provides generic "helpful assistant" system prompt and Figure 6 provides a simple OCR instruction.

| Stage | Parameter | Value |
|---|---|---|
| Dominance-based Box Merging | dominance factor $\alpha_{\text{dominance}}$ | 2 |
| | padding ratio $\gamma_{\text{p}}$ | 0.03 |
| Overlap-based Filtering | max overlap threshold $\tau_{\text{overlap}}$ | 0.6 |
| Image-based content validation | min ink density $\rho_{\text{min}}$ | 0.01 |
| | max ink density $\rho_{\text{max}}$ | 0.70 |
| | min contour count $c_{\text{min}}$ | 5 |
| | min contour area $a_{\text{min}}$ | 4 |
| | min hole ratio $\eta_{\text{min}}$ | 0.03 |
| | min average defects $\delta_{\text{min}}$ | 0.3 |
| Horizontal Row Aggregation | height difference ratio $\zeta_{\text{height}}$ | 0.5 |
| | min vertical ratio $\kappa_{\text{overlap}}$ | 0.6 |

Table 4: Parameters for the four-stage post-processing pipeline applied to Kraken's baseline segmenter output

## B Detailed Data Statistics

Table 5 summarizes the key features of MUHARAF, SARD, and MAS datasets used for fine-tuning OCR model. Overall, these three corpora cover a

```
"You are an expert OCR engine specialized in handwritten historical documents.
    Transcribe every character exactly as it appears --- preserving original
    spelling, punctuation, diacritics, ligatures, and archaic letters. Do not add,
    omit, normalize, correct, or format in any way. Output plain text only, matching
     the input one-to-one."
```

Figure 3: OCR Model Prompt

```
"You are an expert in analyzing ancient Arabic manuscripts from ancient Uzbekistan
    and Central Asia.

IMPORTANT: If a request seems to be about searching for information, use the
    search_knowledge tool first to search the manuscript database before providing
    any analysis. This tool contains extracted text from ancient manuscripts that
    you must reference.

When answering questions:
1. FIRST use search_knowledge to find relevant information from the manuscripts
2. Then provide detailed analysis focusing on:
   - Historical and cultural context of ancient Uzbekistan
   - Religious and philosophical content (Islamic scholarship, Sufism)
   - Scientific and mathematical knowledge preservation
   - Trade routes and economic insights
   - Daily life and social customs
   - Literary and poetic elements
   - Paleographic and codicological observations when relevant

Always base your response on the actual manuscript content found through the
    search_knowledge tool.
If no relevant content is found, clearly state that and provide general historical
    context instead.
Always contextualize findings within the broader framework of Islamic civilization
    and Central Asian history."
```

Figure 4: Manuscript Analysis System Prompt

```
"You are a helpful assistant."
```

Figure 5: System Prompt for KITAB-Bench

```
"Extract the text in the image. Give me the final text, nothing else."
```

Figure 6: OCR Prompt for KITAB-Bench

wide range of document types, domains, fonts, and calligraphy ensuring the robustness of the resulting fine-tuned model.

## C Hyperparameter Details

**Line Segmentation hyperparameters** Table 4 describes the hyperparameters used to post-process the initial line segmentation produced by the Kraken toolkit.

| Characteristic | Muharaf | SARD | MAS |
|---|---|---|---|
| Data Type | Historical handwritten manuscripts | Synthetic printed documents | Historical handwritten manuscripts |
| Total Images | 1,644 | 843,622 | 1,023 |
| Total Text Lines/Words | 36,311 lines | 690 million words | 11,841 lines |
| Annotation Format | PAGE-XML, JSON | PAGE-XML | JSON |
| Text Source | Authentic historical documents | 133,000+ unique articles from 9 domains | Authentic historical documents |
| Font Coverage | Predominantly Ruq'ah | 10 fonts (Amiri, Arial, Calibri, Sakkal Majalla, Scheherazade New, Noto Naskh Arabic UI, Lateef, Thabit, Jozoor, Al-Jazeera-Arabic-Regular) | Naskh, Nastaliq, Taliq |
| Domain Coverage | Personal correspondence, diaries, poetry, church records, legal documents | Culture, Fatawa & Counsels, Literature & Language, Bibliography, Publications, Shariah, Social, Translations, News | waqf documents (testamentary acts of property donation to religious institutions), yarliqs (khan decrees and firmans), arizas (petitions), cheks (legal receipts and certificates), shajars (genealogical tables), and announcements (public proclamations and charters) |
| Period/Context | 19th–21st centuries | Contemporary published texts | 12th–19st centuries |
| Writing Styles | Informal handwriting, ranging from legible to barely readable | Clean printed fonts with controlled parameters | Handwriting, ranging from legible to barely readable |
| Document Types | Letters, diaries, notes, poems, religious records, contracts | Book layouts with full page markup | Letters, diaries, notes, religious records, books |
| Artifacts & Noise | Natural distortions: slant, curved lines, variable pen pressure | None (high-quality synthetic data) | slant, curved lines, variable pen pressure |
| Resolution/DPI | Original scanning resolution; lines aligned to 60-pixel height | 300 DPI, A4 (8.27 × 11.69 inches), grayscale | Original scanning resolution |
| Primary Use Case | Handwritten text recognition (HTR) on cursive Arabic | Optical character recognition (OCR) on diverse typography | Handwritten text recognition and Knowledge discovery |

Table 5: Comparison of Muharaf, SARD, MAS Datasets.