

# KGSAR: A Knowledge Graph-Based Tool for Managing Spanish Colonial Notary Records

Shivika Prasanna<sup>1</sup>, Nouf Alrasheed<sup>2</sup>, Parshad Suthar<sup>1</sup>, Pooja Purushatma<sup>1</sup>, Praveen Rao<sup>1</sup> and Viviana Grieco<sup>2</sup>

<sup>1</sup>University of Missouri-Columbia, Columbia, Missouri, USA

<sup>2</sup>University of Missouri-Kansas City, Kansas City, Missouri, USA

## Abstract

Notary records contain abundant information relevant to historical inquiry but are in physical form and hence, searching for information in these documents could be painstaking. In this demo paper, we present a document retrieval system that allows users to search for a keyword in digitized copies of physical records. The system uses cleaned and denoised images to search a keyword using optical character recognition (OCR) models re-trained on labeled data provided by experts. The word predictions and bounding boxes are stored as a knowledge graph (KG). A keyword query is then mapped to a graph query on the KG. The results are ranked based on text matching. An intuitive user interface (UI) allows a user to search, correct, delete or draw more annotations that are used for retraining of the OCR models.

## Keywords

Knowledge graphs, information retrieval, optical character recognition, historical manuscripts

## 1. Introduction

Historical manuscripts such as 17<sup>th</sup> century Spanish-American notarial scripts, carry a plethora of information that is highly useful to historians in understanding the social, economical, cultural, and political developments during different time periods. Manually searching through the scripts is time consuming and limits the scope of a historian's research findings. With advances in deep learning, OCR models have become more accurate and efficient. However, the lack of high quality training data on specific handwritten collections restricts applicability of pretrained OCR models. Furthermore, efficient and accurate document retrieval is required as the collections can contain millions of handwritten words.

In this paper, we present a new document retrieval system called KGSAR (KG for Spanish American Notary Records) for a set of hand-written Spanish notary documents from the National Archives of Argentina. KGSAR synergistically combines retrained OCR models and the concept of KG to address the challenges in accessing, reading, and searching within the documents. A KG can provide numerous benefits for information/document retrieval [8]. It can enable semantic search and better understanding of users' queries and documents as well as provide explanations

---

*International Semantic Web Conference, 23-27 October 2022, Hangzhou, China*

✉ spn8y@umsystem.edu (S. Prasanna); nalrasheed@mail.umkc.edu (N. Alrasheed); phs2dm@missouri.edu (P. Suthar); ppbh4@missouri.edu (P. Purushatma); praveen.rao@missouri.edu (P. Rao); griecov@umkc.edu (V. Grieco)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

for matched entities and their relationships [8]. In KGSAR, the Resource Description Framework (RDF) and SPARQL are used for efficient representation, indexing, and query processing of data extracted from the documents (e.g., predicted words) via OCR. The KG contains additional facts about the notaries, and is stored and queried using a fast graph database. The UI allows a user to provide additional training data for retraining the OCR models. The design of KGSAR is generic and can be easily adapted to other historical scripts.

## 2. Related Work

Deep learning techniques achieve high accuracy when large, labeled datasets are available [3]. They have also enabled high quality OCR on handwritten documents. To use existing OCR models for specialized collections, we require high quality labeled data from experts.

Alrasheed et.al. [2] showed that after retraining on the Spanish-American notary records, Keras-OCR and YOLO-OCR achieved a better performance as compared to Kraken, Tesseract, and Calamari-OCR. When tested on our collection, the latter systems (which are based on pretrained models for the English language) were only able to detect lines over words and could not recognize any of the characters present in those lines. [12, 13, 14]. For an image containing 670 manually annotated words, Keras-OCR [11] and YOLO-OCR [9, 10] were able to recognize 306 and 146 words respectively, while Kraken, Tesseract and Calamari-OCR were not able to recognize any words in the detected lines.

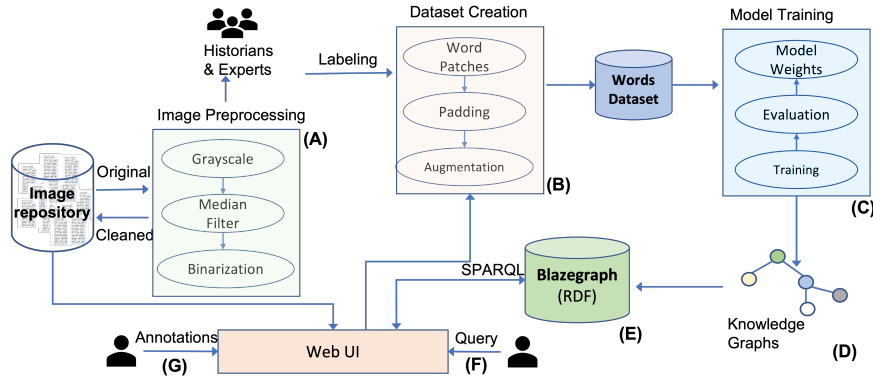
Shaw et.al. [5] proposed a system for converting handwritten medical prescriptions digitally using electronic writing pad and utilized OCR techniques for character recognition in the digital prescriptions, instead of whole words. Sugarawara et.al. [7] proposed a method for retrieving Japanese keywords using a text query where they first generated an image of the query text using Generative semi-supervised model, and then retrieved regions in documents similar to the generated image by feature matching. Preliminary works such as of Kim et.al. [6] presented an end-to-end system that combined word recognition using segmentation with a matching technique designed to handle the large dimensional feature vectors that represented shape description of characters in a word.

Unlike most prior work that focus on text recognition, KGSAR aims to synergistically combine OCR and knowledge management techniques to facilitate efficient and accurate retrieval of 17<sup>th</sup> century Spanish-American notarial scripts.

## 3. Architecture of KGSAR

Seventeenth-century Spanish American notarial scripts include multiple handwritings due to a high turn-over rate in the notary office. Interim notaries did not receive extensive training, thus, handwriting in the documents consists of highly irregular scripts. The current implementation of KGSAR stores 20,000 out of 200,000 images that comprise the entire digitized collection.

KGSAR's architecture is illustrated in Figure 1. Component (A) transforms the document scans into grayscale, applies a median filter to soften backgrounds and removes background noise, and applies image binarization to convert the images to black and white as scanned document images contained noise that affected feature extraction and classification [1]. Component (B)



**Figure 1:** System Architecture

contains 83 cleaned images (166 manuscript pages) that were labeled by Spanish-proficient labelers. This yielded a dataset containing 26,482 words for retraining the OCR models. This dataset is from the hand of *Baldibia y Brisuela*, who by 1650, acted as an interim notary in Buenos Aires, Argentina.

Pretrained Keras-OCR and YOLO-OCR models failed to identify handwritten text as they have been trained on printed English characters. Component (C) represents OCR model training where Keras-OCR recognizer was trained on 21,185 labeled words of 77 images and pretrained detector was used as it was able to accurately draw bounding boxes around the words. YOLO-OCR was trained in a novel way where YOLO was trained as a word localizer to predict only the bounding box coordinates, and convolutional recurrent neural network (CRNN) was trained as a recognizer to identify the text in the bounding boxes.

The retrained models were used to predict on about 20,000 unlabeled images. Component (D) denotes a KG representation built using the predictions. Entities such as the predicted words, bounding box coordinates, image containing the predictions, and the OCR model type that was used were stored as nodes in the KG. These nodes were connected using their respective relations, and serialized into N-triples format. The KG was stored in Blazegraph [4], a popular graph database, as denoted by Component (E). Bulk data loader was used to load all the N-Triples files as an atomic transaction.

Component (F) denotes an intuitive Web UI for a user to pose a keyword query. The word and its n-grams (for word length > 3) are used to construct a SPARQL query, which is executed by Blazegraph. We utilized Blazegraph's FullTextSearch feature to perform exact and partial word matching. Each search result was scored using the cosine distance to the query, allowing words with exact matches and higher match probabilities to be ranked higher. The matching scans were ranked to show the most relevant results. Component (G) denotes the annotation feature where a user can correct the results, delete or annotate more words after a query, to retrain the OCR models with better labeled data.

The UI was developed using HTML5 and AngularJS, and the backend code was developed using Python 3.8. We packaged the entire tool, Blazegraph journal and JAR file into a Docker image to facilitate quick testing and experimentation.

## 4. Demonstration Scenarios

During the demo, a user can interact with KGSAR by posing queries and correcting the bounding boxes as well as labeling new words. We highlight the primary features of KGSAR.



**Figure 2:** Search UI: results for keyword ‘poder’

Figure 2 shows a screenshot of KGSAR after searching for the word *poder*<sup>1</sup>. The user will see the bounding boxes of the words matched in the images and can navigate through the images.

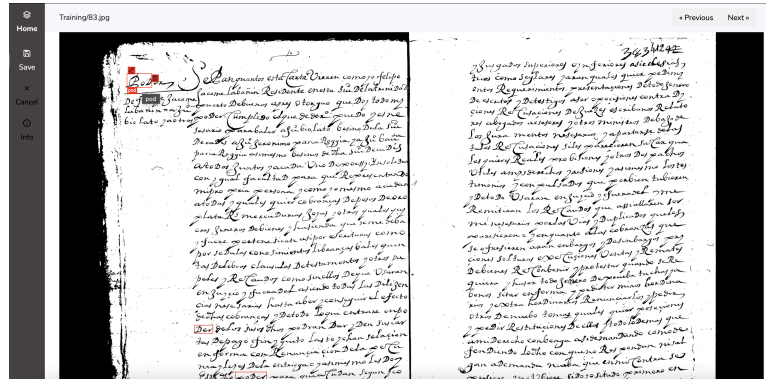
Figure 3 shows the annotation feature for the same word. Here, the user can see edit and delete options for the word, as well as the predicted value for the bounding box. The code for KGSAR is available on GitHub at <https://github.com/MU-Data-Science/KGSAR>.

**Acknowledgments:** This work was supported by a National Endowment for the Humanities (NEH) Digital Humanities Advancement Grant (HAA-271747-20) and a Research and Creative Works Strategic Investment Tier 3 Award from the University of Missouri System. We would like to thank Ryan Rowland and Adam Sisk for labeling a subset of the notary records.

## References

- [1] Alrasheed, N., Rao, P. and Grieco, V., Character Recognition Of Seventeenth-Century Spanish American Notary Records Using Deep Learning. *Digital Humanities Quarterly*

<sup>1</sup>*poder* refers to a power of attorney, a document that, to be valid, required notarial endorsement.



**Figure 3:** Annotate UI: user can edit, delete or view the prediction value

- 15(4) (2021).
- [2] N. Alrasheed, S. Prasanna, R. Rowland, P. Rao, V. Grieco, and M. Wasserman, October. Evaluation of Deep Learning Techniques for Content Extraction in Spanish Colonial Notary Records. In Proceedings of the 3rd Workshop on Structuring and Understanding of Multimedia heritAge Contents, 23-30 (2021).
  - [3] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, ImageNet: A large-scale hierarchical image database, In 2009 IEEE conference on computer vision and pattern recognition. IEEE, 248–255 (2009).
  - [4] Blazegraph, <https://blazegraph.com>. Last accessed June 2022.
  - [5] U. Shaw, R. Mamgai, and I. Malhotra, Medical Handwritten Prescription Recognition and Information Retrieval using Neural Network, In 2021 6th International Conference on Signal Processing, Computing and Control (ISPCC), pp. 46-50 (2021).
  - [6] G. Kim, V. Govindaraju, and S.N. Srihari, An architecture for handwritten text recognition systems, International Journal on Document Analysis and Recognition, 2(1), 37-44 (1999).
  - [7] C. Sugawara, T. Miyazaki, Y. Sugaya, and S. Omachi, Text Retrieval for Japanese Historical Documents by Image Generation, In Proceedings of the 4th International Workshop on Historical Document Imaging and Processing, 19-24 (2017).
  - [8] R. Reinanda, E. Meij, and M. de Rijke, Knowledge graphs: An Information Retrieval Perspective, Foundations and Trends in Information Retrieval, 14(4), 289-444 (2020).
  - [9] J. Redmon, and A. Farhadi, YOLO9000: better, faster, stronger, In Proceedings of the IEEE conference on computer vision and pattern recognition, 7263-7271, 2017.
  - [10] J. Redmon, and A. Farhadi, Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767, 2018.
  - [11] Keras, <https://keras.io>. Last accessed March 2021.
  - [12] Tesseract, [https://en.wikipedia.org/wiki/Tesseract\\_\(software\)](https://en.wikipedia.org/wiki/Tesseract_(software)). Last accessed July 2021.
  - [13] Kraken, <http://kraken.re>. Last accessed July 2021.
  - [14] Calamari OCR, <https://calamari-ocr.readthedocs.io/en/latest/>. Last accessed July 2021.