1. Introduction

Our project uses Google's Vision and Natural Language APIs to generate text transcriptions and name entity values for a Spanish language corpus from 19th century newspapers from the Latin American Digital Initiative (LADI) collection. The generated text values feed into a select group of elements from an existing MODS-compliant metadata schema.

Project contributors include Itza Carbajal, Vivek Khetan, Emma Whittington as part of a Metadata Generation and Interfaces for Massive Datasets graduate course taught by Unmil Karadkar at the University of Texas School of Information.

1.1 Purpose

The Latin American Digital Initiative (LADI) is a collaboration between LLILAS Benson Latin American Studies and Collections at The University of Texas at Austin, the University of Texas Libraries, and Latin American partner institutions to “preserve and provide access to unique archival documentation from Latin America, with an emphasis on collections documenting human rights, race, ethnicity, and social exclusion in the region.”

LADI hosts digitized collection material —provided to them by various partner sites — on their online repository platform (stored as .tiff images).

1.2 Intended Audience

Decision makers and project managers of public libraries, museums, archives looking to optimize their content heavy digitized materials. Also relevant to culture and information specialists from museums, libraries and archives working with historic newspaper articles.

1.3 Project Scope

Our project uses Google’s Vision and Language APIs to generate OCR text transcriptions and name entity values for LADI’s corpus of La Información newspapers. The metadata we generate feeds into and enhances LADI’s existing, MODS-compliant metadata schema. The added value of this project is that the name entity values generated for each transcript— stored as a list of “topics” alongside the object’s other metadata — function as key words/search terms. These ‘topics’/’key words’ ultimately provide LADI database users with a more robust, searchable database.

1.4 Suggested Readings

- Latin American Digital Initiative Documentation
    - Site: http://ladi.lib.utexas.edu/
    - History of LADI work: http://ladi.lib.utexas.edu/en/archive
- Google Vision API Documentation
    - Site: https://cloud.google.com/vision/
    - How to Guides: https://cloud.google.com/vision/docs/

    - Video tutorial: https://techcrunch.com/2016/02/18/google-opens-its-cloud-vision-api-to-all-developers/
    - Quick Guide: https://cloud.google.com/vision/docs/quickstart
    - Supported languages: https://cloud.google.com/vision/docs/languages
- Google Natural Language API Documentation
    - Site: https://cloud.google.com/natural-language/
    - How to Guides: https://cloud.google.com/natural-language/docs/
    - Quick Guide: https://cloud.google.com/natural-language/docs/quickstart-client-libraries
    - Supported languages: https://cloud.google.com/natural-language/docs/languages

2. Quality Assessment

Our quality assurance practices focused on providing suggested steps to optimize the various processed outlined in this project. Each quality assurance practice aims to prevent mistakes or identify possible pitfalls. This in particular is crucial for digital processing as each digital asset can vary drastically as far condition, consistency, and accuracy.

2.1 Preprocessing Images for OCR

Prior to conducting optical character recognition and natural language processing processes, digitally scanned newspaper articles must be assessed for optimal image quality. Processed images provide higher data accuracy. The following tasks should be reviewed prior to OCR process.

- Calculate if image needs to be deskewed. If so,
    - find reference lines in the image
    - Calculate the angle of the lines
    - Calculate the skew angle as an average of the angles
    - Rotate the image
- Document file naming structure for the segmented images of the original

2.2 Uploading Images for OCR

Assessing the quality of images for text recognition must meet the minimum standards imposed by the Google Vision API and the provider, Google Cloud. Several factors could be identified from the Google Vision documentation that outlined minimum expectations for highest quality outputs. These included supported image file formats, resolution of image, and file size.

- Supported file formats include: JPEG, PNG8, PNG24, GIF, Animated GIF (first frame only), BMP, WEBP, RAW, ICO.

- File resolution is outlined as a standard minimum of 640 x 480 pixels (about 300k pixels), but notes that OCR processes require 1024 x 768 pixels for TEXT_DETECTION and DOCUMENT_TEXT_DETECTION as OCR requires more resolution to detect characters.

- Image size as stated on the Google documentation materials notes that in addition to the file resolution, files sent to Google Cloud Vision API should not exceed 4 MB. Users should also note that

Vision API imposes an 8 MB per request limit.

2.3 Optical Character Recognition Extracted Data
Typically, OCR technology providers provide measurement metrics to assess the accuracy of the optical character recognition results on a character level. The Google Cloud Vision API does not provide this feature as part of the interface. As a result, this project using a 99% character recognition accuracy rate (meaning 1 out of 100 characters) agreed on a minimum of a 90% accuracy rate to represent high quality.

2.4 Identified NPL Entities
For this aspect of the project, we decided to use a word recognition accuracy method. This was necessary as each entity would need to be a fully recognizable word in order to better meet user expectations.

2.5 Metadata Quality Framework
This project utilized seven dimensions of information quality: completeness, accuracy, provenance, conformance to expectations, logical consistency and coherence, timeliness, and accessibility. as described by Thomas R. Bruce, Legal Information Institute, Cornell Law School; Diane I. Hillmann from the National Science Digital Library. These dimensions originated from the Quality Assurance Framework (QAF) developed by Statistics Canada.
READ MORE HERE: https://ecommons.cornell.edu/handle/1813/7895

3. Command Line
The command line interface descriptions outlined below aim to help demystify the python arguments used in this projects. The descriptions are also included in the python arguments themselves.

3.1 Optical Character Recognition Argument
Name of Argument: ladi_ocr.py
Uses Google-Vision API to print transcriptions of one or more .png screenshots stored in a single folder location. These can then be saved to a .txt file using command line functionality.

Usage:
```
python scripts/ladi_ocr.py <path to images> (argument)
python scripts/ladi_ocr.py --help
python scripts/ladi_ocr.py --version
```

Options:
```
--help     Show this screen.
--version     Show version.
```

Parameters:
```
<path to images>
```

3.2 Natural Language Processing Argument

Name of Argument: ladi_nlp.py

Uses Google-Language API to extract and print named entities (“person”, “location”, “organization”) from the provided text file. These can then be saved to a .txt file using command line functionality.

Usage:

```
python scripts/ladi_ocr.py <path to text file user want to extract NER from> (argument)
python scripts/ladi_nlp.py --help
python scripts/ladi_nlp.py --version
```

Options:

```
--help          Brief suggestion about what to do.
--version     Prints script’s name and the version user is using.
```

Parameters:

```
<path to text file user wants to extract NER from>
```