ATTRIBUTES EXTRACTION

TE PARTS

Contents

OBJECTIVE

Need to extract the attributes data for the selected parts from the respective catalogs (in PDF format). Attributes data resides in the catalogs as:

a. Plain Text Data
b. Table Data

Using ICT catalog as a base template

1. Write a python script to extract the PDF data
2. Generalize the python script in order to automate the data extraction from different PDFs.

EXPLORATORY APPROACHES

1. Direct Data Extraction: Used the following products available in the market to extract data from PDFs from tabular as well as text sections
    a. *Tabula (Free)*
    b. *PyPDF (Free)*
    c. *PDFTables (Paid)*
    d. *PDFMiner (Free)*
    e. *Adobe PDF to Excel/Word Converter (Paid)*

2.

    Limitations: Most of the above tools/libraries seem to use the table borders as criteria to distinguish the contents of the table from external text. But in most of the catalogs, the borders are missing such as vertical/horizontal borders or borders around headers. Due to this, the table contents are not identified properly and results in distorted output.

2. PDF to HTML/XML Converter: Converted the PDF into an HTML/XML document and tried to extract the required data using the HTML/XML structure.

3.

    Limitations: The HTML/XML document does not have a definite structure even within a single PDF, hence it is difficult to extract the data using this approach and especially difficult to generalize it for different catalog templates.

3. OCR – Optical Character Recognition: Converted the PDF document into an image, tried to locate the location and dimensions of the data table using the pixel data (primarily RGB data) and extracted the data from the required pixel area using the Tesseract (used the python wrapper)

    Limitations: Since the script currently uses the catalog template information such as RGB data and that information might not be consistent across the catalogs, the script needs to be modified every time a new template is encountered.

FINAL APPROACH

OCR – Optical Character Recognition

OCR is the finalized approach since the ability to customize the data extraction process as well as the accuracy of the results is relatively much higher as compared to the other approaches. Below is the detailed process followed for the data extraction using the OCR technique.

NOTE: The below process is designed based on ICT Catalog template. For other templates, there might have to be some changes to the scripts to get the result.

Step 1: Splitting the PDF Document by individual pages
Step 2: Removing the existing image elements from each PDF page using *ImageMagik* library
Step 3: Converting each PDF page into an image (.jpeg, .png etc) using the *ImageMagik* library and getting the image information (RGB, width, height) using PIL, a python library
Step 4: Identifying all the vertical and horizontal lines in the page using the pixel color information
Step 5: Getting a list of possible combinations of the above vertical and horizontal lines and keep only those which are potentially the borders of a data table
Step 6: Using the list of finalized combinations, get the table count in the page along with the location, width and height of the table
Step 7: For each of the identified tables, get all the cells' location using the intersection of vertical and horizontal borders
Step 8: Crop the image for each of the cells and extract the text from the cropped image using the Tesseract python wrapper
Step 9: For each table, Identify the header location using the top of the vertical borders and the font color of header elements
Step 10: Distinguish the parent headers from the regular headers using the horizontal border separating the two of them
Step 11: Extract the text value for each of the headers using the same approach as used for the rest of the table cells
Step 12: Storing the resulting values into a data frame and exporting the output in a .csv/.txt format

NEXT STEPS

Objective

Data Extraction from Additional Catalogs

Challenges

The attribute data from plain text sections in the catalogs can be extracted with minor modifications to the script. The challenge lies in extracting the tabular data since the additional catalog templates are quite different from the ICT catalog template as described below:

a. No vertical borders in most of the additional catalog templates, which makes it difficult to separate one column from another and the ones which do have vertical borders, have discontinuities in between two rows making it difficult to identify the two rows as parts of the same table
b. No different font color used for header elements making it difficult to distinguish between normal table cell, header cell and parent header cell
c. Even within the same catalog, the template in which the tabular data is stored varies
d. In some of the catalogs, a single data table extends across multiple pages

These are a few challenges I could identify looking at the catalogs. The existing python script needs to be customized keeping this challenges into consideration.