Image Segmentation and Text Extraction from PDF based on PDF processing and OCR with Python

Context: I want an python component (using Python 3.11) that receives a file-path to a pdf file as input and generates data for a SQLite DB on the structure of the pdf document and extract text data for textual paragraphs in the detected language and/or is using Latex, if content is mathematical or within a table all using open-source software (with e.g. Nougat (facebookresearch/nougat: Implementation of Nougat Neural Optical Understanding for Academic Documents (github.com) https://github.com/facebookresearch/nougat and other python based software components like py pdf 4.2.0 https://pypdf.readthedocs.io/en/stable/index.html). There are also other open source PDF, OCR, Latex solutions.

- Data SQLite DB model will be provided
- Text/Content within the pages should be segmented into rectangles at 400dpi resolution with a 1 (one) pixel additional padding on the non-white content. – these rectangles should be stored with the x/y position values relative to upper left corner and together with the width and height of the rectangle.
    - An rectangle should be around ** an entire text-paragraph, ** a headline (of any level), page-title/subtitle, authors, abstract/summary, ** page headers, ** page footers, ** page-based footnotes, ** around a table, ** around formula outside paragraphs, ** around figures (like block-diagrams, or data graphs), ** around images/pictures, ** reference lists
    - Does the pages has 1, 2 or 3 (text) columns.
    - If possible font information should be extracted from the text, i.e. font-size, is it bold, what is the pixel difference between lines in paragraphs, between paragraphs and between directly adjacent elements within the document
    - also: is the text (i.e. paragraphs with more than 2 or 3 lines) aligned to the left, right, centered or justified. Does the paragraph has a positive indent (i.e.,, white spaces in the 1st line on the left), or a negative indent (i.e. only the 1st line is more to the left than the next lines), and if justified, has the last line on the page an indent on the right or not.
- Storing the png (400 dpi) rectangles of none-white content in the SQLite DB relate to each PDF page with absolute x/y position values
- Generating textual content from 3 independent sources in DB:
    - Extract content from Nougat for the entire file and segment it in pages – (and in paragraphs – if that is an easy step)
    - Extract content from pypdf for the entire file and segment it in pages – (and in paragraphs – if that is an easy step)
    - Extract content via OCR for each rectangle and storing it in 3 modes:
        - RAW – output from the OCR – which include false identifications
        - Improved – improved OCR data with the use of data from Nougat and pypdf + potential removal of hyphenations from line endings.
        - Failure(s) – data related to false OCR identifications – i.e. data reacted to the word that was falsely processed by OCR within the image .. i.e. rectangle data: x/y position + height width and the correct data from Nougat/pypdf (if they exists). Data are stored as csv with “,” and “;” as delimiters