Detection and recognition of text in images for Text2Speech modules

1st Toporas Tudor Andrei
1310B

2nd Luchian Alexandru
1311A

I. Introduction

The necessity of digitisation is rapidly increasing in the modern era. Due to the growth of information and communication technologies (ICT) and the wide availability of handheld devices, people often prefer digitized content over the printed materials including books and newspaper. Also, it is easier to organize digitized data and analyze them for various purposes with many advanced techniques like artificial intelligence etc. So to keep up with the present technological scenario, it is necessary to convert all the information present till now which is in the printed format to digitised format.

II. State of the art & Related work

A. PP-OCR: A Practical Ultra Lightweight OCR System

[3]The Optical Character Recognition (OCR) systems have been widely used in various of application scenarios, such as office automation (OA) systems, factory automations, online educations, map productions etc. However, OCR is still a challenging task due to the various of text appearances and the demand of computational efficiency. In this paper, it is proposed a practical ultra lightweight OCR system, i.e., PP-OCR. The overall model size of the PP-OCR is only 3.5M for recognizing 6622 Chinese characters and 2.8M for recognizing 63 alphanumeric symbols, respectively. It introduces a bag of strategies to either enhance the model ability or reduce the model size. The corresponding ablation experiments with the real data are also provided. Meanwhile, several pre-trained models for the Chinese and English recognition are released, including a text detector (97K images are used), a direction classifier (600K images are used) as well as a text recognizer (17.9M images are used). Besides, the proposed PP-OCR are also verified in several other language recognition tasks, including French, Korean, Japanese and German. All of the above mentioned models are open-sourced and the codes are available in the GitHub repository, i.e., https://github.com/PaddlePaddle/PaddleOCR.

B. OCR-free Document Understanding Transformer

[4]Understanding document images (e.g., invoices) is a core but challenging task since it requires complex functions such as reading text and a holistic understanding of the document. Current Visual Document Understanding (VDU) methods outsource the task of reading text to off-the-shelf Optical Character Recognition (OCR) engines and focus on the understanding task with the OCR outputs. Although such

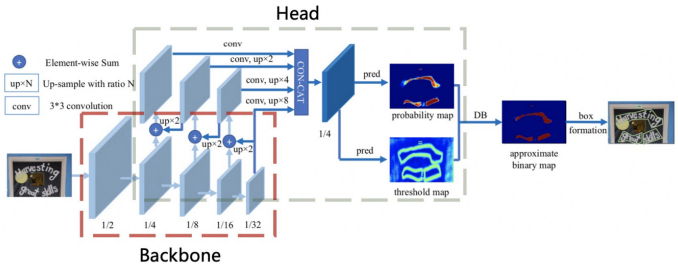

Fig. 1. Architecture of the text detector DB. This figure comes from the paper of DB (Liao et al. 2020). The red and gray rectangles show the backbone and head of the text detector separately.

OCR-based approaches have shown promising performance, they suffer from 1) high computational costs for using OCR; 2) inflexibility of OCR models on languages or types of document; 3) OCR error propagation to the subsequent process. To address these issues, in this paper, the authors introduce a novel OCR-free VDU model named Donut, which stands for Document understanding transformer. As the first step in OCR-free VDU research, we propose a simple architecture (i.e., Transformer) with a pre-training objective (i.e., cross-entropy loss). Donut is conceptually simple yet effective. Through extensive experiments and analyses, we show a simple OCR-free VDU model, Donut, achieves state-of-the-art performances on various VDU tasks in terms of both speed and accuracy. In addition, the authors offer a synthetic data generator that helps the model pre-training to be flexible in various languages and domains. The code, trained model and synthetic data are available at https://github.com/clovaai/donut.

C. Image-based table recognition: data, model, and evaluation

[5]Important information that relates to a specific topic in a document is often organized in tabular format to assist readers with information retrieval and comparison, which may be difficult to provide in natural language. However, tabular data in unstructured digital documents, e.g., Portable Document Format (PDF) and images, are difficult to parse into structured machine-readable format, due to complexity and diversity in their structure and style. To facilitate image-based table recognition with deep learning, we develop the largest publicly available table recognition dataset PubTabNet (https://github.com/ibm-aur-nlp/PubTabNet), containing 568k

table images with corresponding structured HTML representation. PubTabNet is automatically generated by matching the XML and PDF representations of the scientific articles in PubMed Central Open Access Subset (PMCOA). We also propose a novel attention-based encoder-dual-decoder (EDD) architecture that converts images of tables into HTML code. The model has a structure decoder which reconstructs the table structure and helps the cell decoder to recognize cell content. In addition, the authors propose a new Tree-Edit-Distance-based Similarity (TEDS) metric for table recognition, which more appropriately captures multi-hop cell misalignment and OCR errors than the pre-established metric. The experiments demonstrate that the EDD model can accurately recognize complex tables solely relying on the image representation, outperforming the state-of-the-art by 9.7% absolute TEDS score.

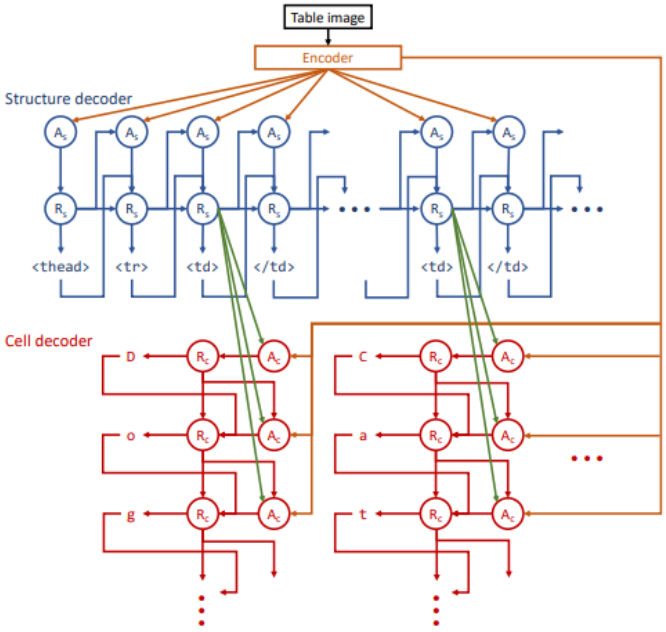

Fig. 2. EDD architecture.

D. Upcycle Your OCR: Reusing OCRs for Post-OCR Text Correction in Romanised Sanskrit

[6]The authors propose a post-OCR text correction approach for digitising texts in Romanised Sanskrit. Owing to the lack of resources our approach uses OCR models trained for other languages written in Roman. Currently, there exists no dataset available for Romanised Sanskrit OCR. So, we bootstrap a dataset of 430 images, scanned in two different settings and their corresponding ground truth. For training, we synthetically generate training images for both the settings. We find that the use of copying mechanism (Gu et al., 2016) yields a percentage increase of 7.69 in Character Recognition Rate (CRR) than the current state of the art model in solving monotone sequence-to-sequence tasks (Schnober et al., 2016). We find that our system is robust in combating OCR-prone errors, as it obtains a CRR of 87.01% from an OCR output with CRR of 35.76% for one of the dataset settings. A human judgment survey performed on the models shows that our proposed model results in predictions which are faster to comprehend and faster to improve for a human than the other systems.

E. Profiling of OCR'ed Historical Texts Revisited

[7]In the absence of ground truth it is not possible to automatically determine the exact spectrum and occurrences of OCR errors in an OCR'ed text. Yet, for interactive post-correction of OCR'ed historical printings it is extremely useful to have a statistical profile available that provides an estimate of error classes with associated frequencies, and that points to conjectured errors and suspicious tokens. The method introduced in Reffle (2013) computes such a profile, combining lexica, pattern sets and advanced matching techniques in a specialized Expectation Maximization (EM) procedure. Here the authors improve this method in three respects: First, the method in Reffle (2013) is not adaptive: user feedback obtained by actual postcorrection steps cannot be used to compute refined profiles. the authors introduce a variant of the method that is open for adaptivity, taking correction steps of the user into account. This leads to higher precision with respect to recognition of erroneous OCR tokens. Second, during post correction often new historical patterns are found. They show that adding new historical patterns to the linguistic background resources leads to a second kind of improvement, enabling even higher precision by telling historical spellings apart from OCR errors. Third, the method in Reffle (2013) does not make any active use of tokens that cannot be interpreted in the underlying channel model. They show that adding these uninterpretable tokens to the set of conjectured errors leads to a significant improvement of the recall for error detection, at the same time improving precision.

III. Method Description

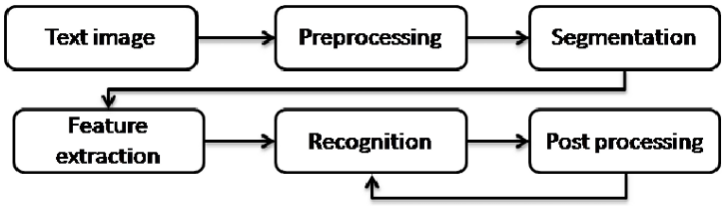

Fig. 3.

Through our app we want to make text detection and recognition for the letters in the english alphabet but also to include the special characters in the romanian alphabet (i.e. ă, î, â, ș, ț).

In order to get there, we first need to go through a series of steps to get to the desired result. After the image is acquired we apply an adaptive Gaussian thresholding [1]. After which, on the binary image we apply the histogram projection method in order to get the line level segmentation. [2]

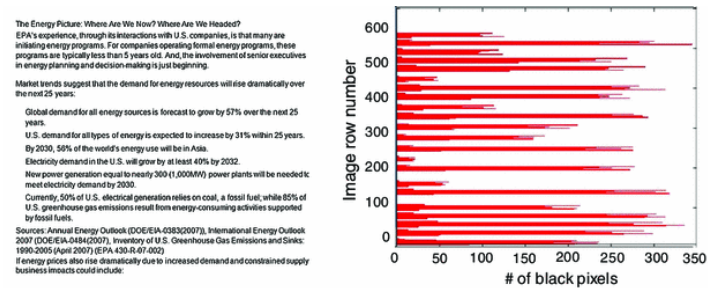

Fig. 4.

IV. Preliminary Results

We have an interface which we use to pass the an image to be processed.

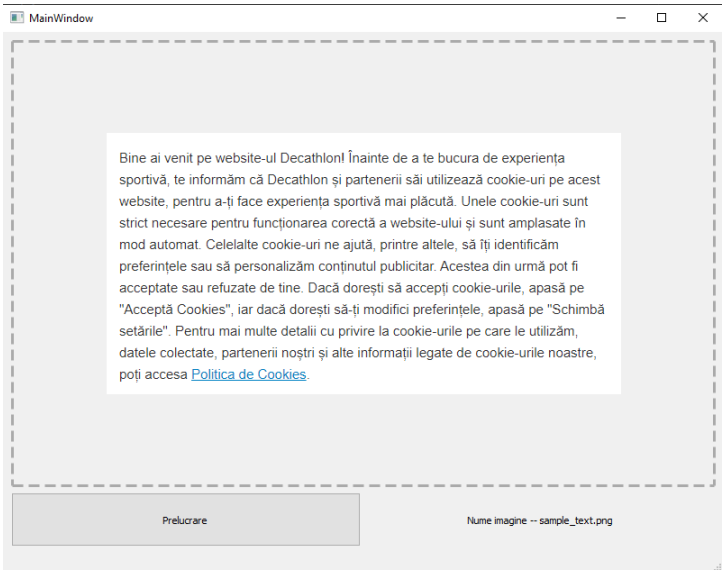

Fig. 5.

Past the interface, we use the methods previously mentioned to segment the lines in the original image.

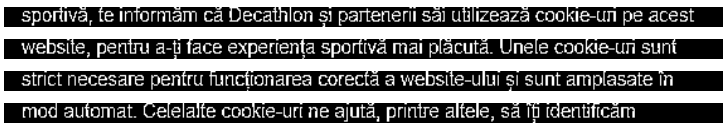

Fig. 6.

V. Preliminary Conclusion

In conclusion, we managed to do a line level segmentation of a quite uniform image. In the next stages of development we will focus on making the binarization more consistent on images with more challenging lighting, and most importantly on the recognition of the characters.

References

[1] OpenCV for python https://docs.opencv.org/4.5.4/d6/d00/tutorial_py_root.html
[2] Susmith Reddy article writer https://towardsdatascience.com/pre-processing-in-ocr-fc231c6035a7
[3] https://paperswithcode.com/paper/pp-ocr-a-practical-ultra-lightweight-ocr
[4] https://paperswithcode.com/paper/donut-document-understanding-transformer
[5] https://paperswithcode.com/paper/image-based-table-recognition-data-model-and
[6] https://paperswithcode.com/paper/upcycle-your-ocr-reusing-ocrs-for-post-ocr
[7] https://paperswithcode.com/paper/profiling-of-ocred-historical-texts-revisited