Final Project Report

Farkhad Kuanyshkereyev
Università degli studi dell'Aquila, Italy
Email: farkhad.kuanyshkereyev@student.univaq.it

Academic year: 2022–2023

1 Problem Description

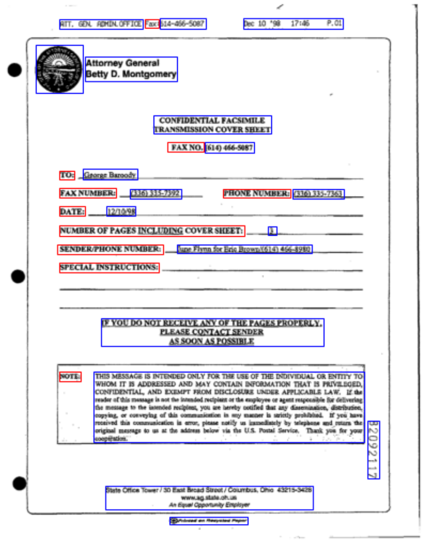
The problem addressed in this report is related to the Document Parsing. A lot of data is being generated every day, including invoices, receipts, bills, etc. Manually parsing information from this data requires plenty of human effort and time. Therefore many companies need to have an automated system of parsing information from document images. Given the dataset of document images, the goal is to detect relevant text fields and parse them in a digital format. Documents can be of different dimensions, resolution, positions. They may have distortions and noise. This complicates the process of parsing document images. This report shows the set of experimental steps for mitigating the defined problem.

2 Data Preparation

2.1 Dataset

The dataset used for this problem is called Form Understanding in Noisy Scanned Documents (FUNSD) [1]. It is a dataset that is used for text detection, optical character recognition (OCR), spatial layout analysis and form understanding. It contains 199 fully annotated forms, 31485 words, 9707 semantic entities, and 5304 relations. For simplicity, only 2 classes from the original dataset were considered. Those classes are questions and others (answers). Answers are regarded as relevant text fields that need to be detected and parsed. All images are in PNG format. Their corresponding annotation files are in JSON format. Annotation files contain class names and bounding box coordinates in PASCAL VOC format. Figure 1 shows the labeled sample image from the FUNSD dataset, where red bounding boxes are questions and blue bounding boxes are answers.

Figure 1: Labeled Sample from FUNSD. Red boxes - questions. Blue boxes - answers.



ATT. GEN. ADMIN.OFFICE Fax:614-466-5087 Dec 10 '98 17:46 P.01

Attorney General
Betty D. Montgomery

CONFIDENTIAL FACSIMILE
TRANSMISSION COVER SHEET

FAX NO. (614) 466-5087

TO: George Baroody

FAX NUMBER: (336) 335-7392 PHONE NUMBER: (336) 335-7363

DATE: 12/10/98

NUMBER OF PAGES INCLUDING COVER SHEET: 1

SENDER/PHONE NUMBER: Jane Flynn for Eric Brown/(614) 466-8980

SPECIAL INSTRUCTIONS:

IF YOU DO NOT RECEIVE ANY OF THE PAGES PROPERLY,
PLEASE CONTACT SENDER
AS SOON AS POSSIBLE

NOTE: THIS MESSAGE IS INTENDED ONLY FOR THE USE OF THE INDIVIDUAL OR ENTITY TO WHOM IT IS ADDRESSED AND MAY CONTAIN INFORMATION THAT IS PRIVILEGED, CONFIDENTIAL, AND EXEMPT FROM DISCLOSURE UNDER APPLICABLE LAW. If the reader of this message is not the intended recipient or the employee or agent responsible for delivering the message to the intended recipient, you are hereby notified that any dissemination, distribution, copying, or conveying of this communication in any manner is strictly prohibited. If you have received this communication in error, please notify us immediately by telephone and return the original message to us at the address below via the U.S. Postal Service. Thank you for your cooperation.

82092117

State Office Tower / 30 East Broad Street / Columbus, Ohio 43215-3428
www.ag.state.oh.us
An Equal Opportunity Employer

Printed on Recycled Paper

2.2 Data Preprocessing

The original dataset was not split into training, validation, and testing. Therefore this was performed locally. The training set contains 159 images, the validation set contains 30 images, and the testing set contains 10 images. Validation set was used to track performance metrics and testing set was used to visualize inference results. All document images were grayscaled and resized to 416 by 416 pixels size. All bounding boxes were rescaled correspondingly. Scaling was applied on images by dividing their pixel values by 255, which is the maximum value for a pixel.

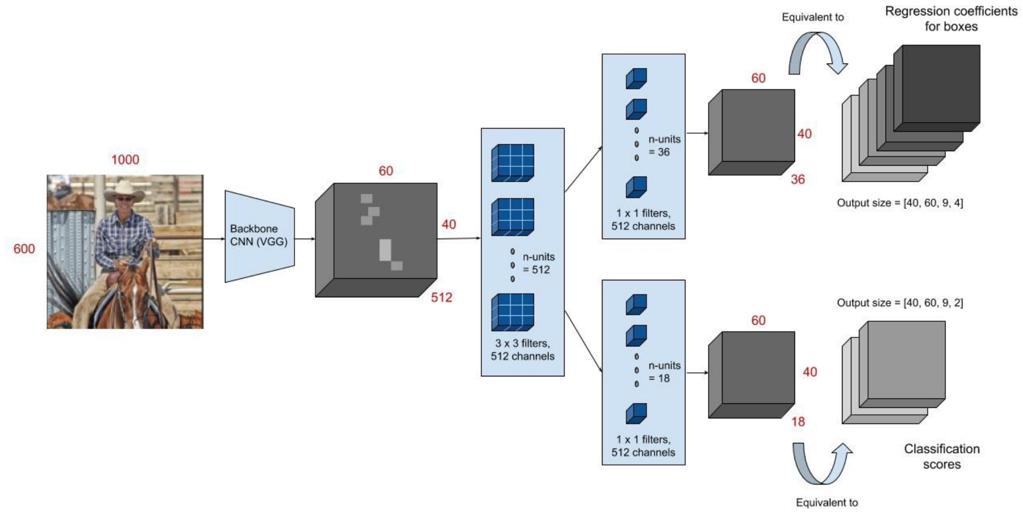
3 Model Training

The deep learning framework that was used to solve the problem is PyTorch. PyTesseract was used to perform OCR on detected bounding boxes. After researching it was found out that stochastic gradient descent is the most popular and flexible optimizer that can be used for the object detection task. Therefore this optimizer was used with learning rate equal to $3e^{-4}$, momentum equal to 0.9, and no weight decay. Hyperparameters for the optimizer were set after a series of experiments.

The main task was to perform relevant text field detection. This is defined as an object detection problem in machine learning. There are many algorithms that can solve this problem such as YOLO, SSD, R-CNN, and so on. For this attempt Faster R-CNN [2] algorithm was used. It is an algorithm that uses the two-stage approach, where the Region Proposal Network (RPN) proposes detected regions and then the Convolutional Neural Network (CNN) classifies them. Faster R-CNN usually outperforms other object detection algorithms. This is the reason why it was chosen for this attempt. The transfer learning approach was applied with pretrained weights for the Faster R-CNN model. Figure 2 shows Faster R-CNN architecture.

The model training was performed for 50 epochs to ensure reaching the plateau in loss function. Faster R-CNN uses its own loss function that consists of box coordinates regression loss and classification loss. The training was performed in Google Colab with GPU hardware accelerator and the batch size was set to 16. The prediction head of the Faster R-CNN, which is a Fast R-CNN predictor, was changed to have 3 classes as output. Those 3 classes include questions, answers, and background.

Figure 2: Faster R-CNN Architecture



1000
600
Backbone CNN (VGG)
60
40
512
n-units = 512
3 x 3 filters, 512 channels
n-units = 36
1 x 1 filters, 512 channels
60
40
36
Equivalent to
Regression coefficients for boxes
Output size = [40, 60, 9, 4]
Output size = [40, 60, 9, 2]
n-units = 18
1 x 1 filters, 512 channels
60
40
18
Classification scores
Equivalent to

4 Evaluation

4.1 Metrics

Evaluation of the model was performed on validation set after every epoch. Metrics that were computed are intersection over union (IoU), precision, recall, and f1-score. Those metrics were computed pixel-wise by considering masks of true and predicted bounding boxes. An empty mask with the shape of the document image is filled with 0. A copy of that mask of the same shape is created with 1 placed in true bounding box regions. Third mask of the same shape has values of 2 in placed of predicted bounding box regions. Then, the second and the third masks are added together to produce the combined mask. 0 denotes True Negatives, 1 denote False Negatives, 2 denote False Positives, and 3 denote True Positives. Metrics were calculated by considering all cell values. Formulas are shown below:

$$IoU = \frac{TP}{FN + FP + TP}$$

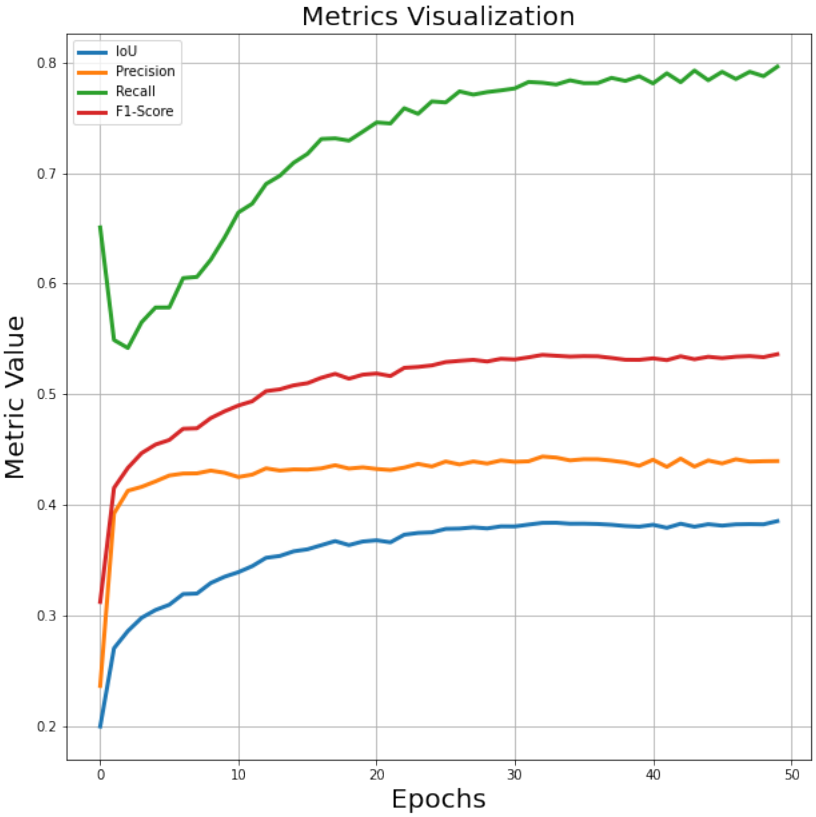
$$Precision = \frac{TP}{FP + TP}$$

$$Recall = \frac{TP}{FN + TP}$$

$$F1 = \frac{TP}{TP + 0.5 * (FP + FN)}$$

Those metrics were calculated separately for questions and answers and then averaged for each epoch. Their visualization across 50 epochs is shown in Figure 3.

Figure 3: Metrics Visualization



4.2 Inference

After finishing the training, the best model's state dictionary was loaded and then inference was performed on the testing dataset. The best model was chosen based on F1-Score. Figure 4 shows the inference result. The confidence threshold was set to 0.6. Thus, any predicted bounding box with a score lower than or equal to that confidence threshold is omitted in visualization.

To parse the relevant information, each predicted bounding box with the class answers was considered. PyTesseract OCR was applied on each blue bounding box. Bounding box shape was increased by adding a buffer of 10 pixels on each side of the box. This is done to ensure better readability of the text region. Figure 5 shows 3 results of applying OCR. Predictions are shown at the top of each image. Other inference results can be seen in the Google Colab notebook that comes with the report.

Figure 4: Inference Result. Left - true bounding boxes. Right - predicted bounding boxes.

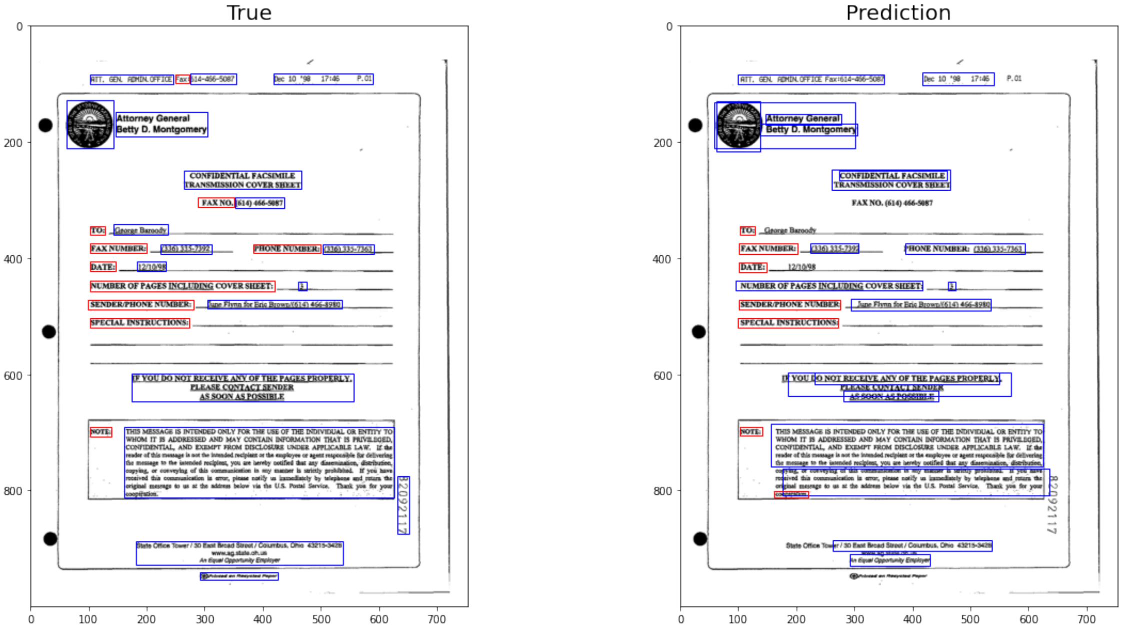
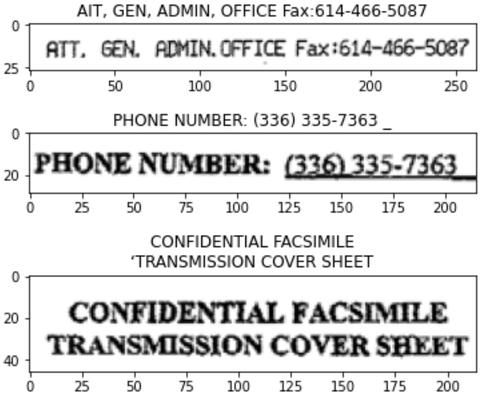


Figure 5: PyTesseract OCR Results.


AIT, GEN, ADMIN, OFFICE Fax:614-466-5087

PHONE NUMBER: (336) 335-7363 _

CONFIDENTIAL FACSIMILE
'TRANSMISSION COVER SHEET

5 Discussion

Currently the best model has F1-Score equal to 0.536. This score was achieved before reaching the plateau. It relatively low because the model has high recall and low precision values. The score can be improved by considering a different object detection algorithm such as YOLO, SSD, DETR, etc. Another part that can be improved is to perform data augmentation techniques for artificially increasing the size of the dataset. In the current attempt only answers were detected. One idea for experimenting is to try to detect both questions and answers and then use the Graph Neural Network to perform link prediction between them. This way the information from document image can be parsed in a dictionary format, where each answer is mapped to a certain question. Document Parsing is a wide problem that cannot be completely addressed by considering the limited FUNSD dataset. The dataset contains only forms. There could be invoices and receipts with dynamic fields such as tables that may have multiple items. In certain cases documents may be comprised of multiple pages which further complicates the problem. Therefore having a larger dataset will certainly help in improving the performance of the relevant text detection part.

References

[1] Jean-Philippe Thiran Guillaume Jaume, Hazim Kemal Ekenel. Funsd: A dataset for form understanding in noisy scanned documents. In *Accepted to ICDAR-OST*, 2019.

[2] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *NIPS*, pages 91–99, 2015.