Documentation

Folder Structure:

- ctpn
  - ctpn_boxes.py : modified from ctpn/demo_pb.py (original ctpn repo: refer to setup installation below)
  - framework.py : framework to parse data from ctpn + tesseract output and save it to json format
  - text.yml : contains ctpn configuration (from original repo)
- data : contains protobuf file for ctpn computational graph defintion
- lib : ctpn network implementation (source: https://github.com/eragonruan/text-detection-ctpn)
- sampleids : sample images provided
- tessdata : contains best trained LSTM tesseract data for english language

Usage:

---

python ctpn/framework.py <image_path>

Input: image path
Output: json file with details in the image path

Methodology:

---

1. Used CTPN to detect bounding boxes in the image.
2. Crop image with every bounding box and pass output to tesseract.
3. Get all textual information in order of their appearance in image from top to bottom
4. Parse this textual information and bounding boxes to obtain different entities.
5. Save the obtained entities in a dictionary and dump into a json file.

Heuristics used for obtaining different entities:

Common terms:

output text: a list of strings where each string is the tesseract output of the ctpn bounding boxes
bbox: bounding box output of CTPN network
psm: page segmentation mode

Name :-
Voter_ID: Search "Elector's Name" in output text and return output after ':'
DL: Search "Name" in output text and return text of next element in output text

Father's/Husband's Name :-
Voter_ID: Search "Husband's Name" in output text and return output after ':'
DL: Search "S/W/D" in output text and return text of next element in output text

ID number :-
Voter_ID: Search "IDENTITY CARD" in output text and return output after 'CARD'
DL: Search "Licence No" in output text and return output after ': ‘

For DL only:

DoB:-
Search "DOB" in output text and return output after ' '

Blood Group:-
Search "BG" in output text and return output after 'BG: ‘, Tesseract might recognize 'O' as '0'

Address:-
Since Address keyword is not present in template but location of address is below blood group and above Date of issue, so I formulated the bbox coordinates of address from these two fields and obtained the address by passing it into the tesseract.

For Voter_ID only:
Age:-
Search "Age" in output text and return output after ': '

Gender:-
Search "Sex" in output text and pass the cropped image of the next bounding box in tesseract with psm = 9 to treat as single word in a circle to get the gender as Male or Female.

Installation Setup (Mac OS):

Install tesseract 4.0 dependencies:

```
brew install automake autoconf autoconf-archive libtool
brew install pkgconfig
brew install icu4c
brew install leptonica
brew install gcc
```

Install tesseract from source code:

```
git clone https://github.com/tesseract-ocr/tesseract/
cd tesseract
./autogen.sh
./configure CC=gcc CXX=g++ CPPFLAGS=-I/usr/local/opt/icu4c/include
 LDFLAGS=-L/usr/local/opt/icu4c/lib
make -j
make install
```

Installation of pytesseract (wrapper over tesseract binaries):

```
pip install pytesseract
```

Get best LSTM trained model for english:

```
git clone https://github.com/tesseract-ocr/tessdata_best.git
copy eng.traineddata into the tessdata directory of the submission folder
```

CTPN tensorflow implementation:

```
git clone https://github.com/eragonruan/text-detection-ctpn.git
```