Meta-Data Extraction from the Rental Agreements

1. Business Requirement - Overview

To build an AI/ML model to extract data from the rental agreements. The rental agreements will be in different data formats and available in the form of PDFs to perform the extraction.

The model should be able to extract the following fields from all the documents,

1. Agreement Value
2. Agreement Start Date
3. Agreement End Date
4. Renewal Notice (Days)
5. Party One
6. Party Two

Note: Please do not use a rule-based/oriented approach (RegEx, static conditions etc).

2. Environment Details

a) The development workspace runs on Ubuntu 16.04 LTS with the following packages pre-installed.

- Python 3.5+
- Nltk
- Keras
- Tensorflow

- Sklearn
- Matplotlib
- Plotly
- Numpy
- Pandas
- Csv
- Datetime
- Word2Vec
- Gensim
- python-docx

In case if you would like to install any specific package for Python3 then use the following command:

```
python3 -m pip install <package_name>

Ex: python3 -m pip install spacy tensorflow
```

b) The environment also has Jupyter notebook. Please use the following steps to access,

- Open the IDE
- Click on **Live preview** button.
- Provide the password as "password".
- While creating the python file make sure you keep the directory as **/home/user/workspace/code**
- Use the `Submit` button in the IDE, to get your models validated

3. Dataset Details

The rental agreements are in the PDF format.

You will also find Extracted Meta data for training and validation sets in separate '.csv' files (mentioned below), these files can be downloaded from the **Problem Description** section, under the same section, where you downloaded this PDF.

These files will be your training and validation data files (correlate by pdf file name, for every file, its corresponding metadata is given), along with their respective PDFs.

- **TrainingTestSet.csv** (Extracted meta data for agreements in Training_data.zip)
- **ValidationSet.csv** (Extracted meta data for agreements in Validation_data.zip)

The training and evaluation datasets are available in the IDE at the following location.

/home/user/workspace/data

The above directory has 2 sub-directories:

- **training/:** contains a total of 43 rental agreements
- **eval/**: contains a total of 8 rental agreements

The training dataset can be downloaded to your local machine from the URL.

4. Assignment Deliverables

If you’re using Jupyter notebook, then store all the code files in **/home/user/workspace/code/** directory.

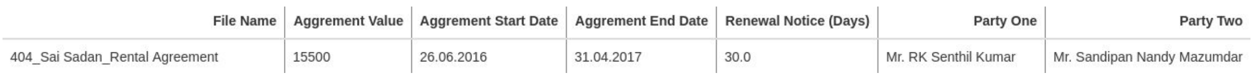
Use the following file locations to write the predictions for the training and the evaluation datasets.

- Write the training CSV to: **/home/user/workspace/output/training.csv**
- Write the evaluation CSV to: **/home/user/workspace/output/eval.csv**

Please use the following format for the CSV files,

File Name,Agreement Value,Agreement Start Date,Agreement End Date,Renewal Notice (Days),Party One,Party Two

404_Sai Sadan_Rental Agreement,15500,26.06.2016 ,31.04.2017,30,Mr. RK Senthil Kumar,Mr. Sandipan Nandy Mazumdar



File Name	Aggrement Value	Aggrement Start Date	Aggrement End Date	Renewal Notice (Days)	Party One	Party Two
404_Sai Sadan_Rental Agreement	15500	26.06.2016	31.04.2017	30.0	Mr. RK Senthil Kumar	Mr. Sandipan Nandy Mazumdar

Note:

1. The **Agreement Value** column should have only the numeric value.
2. The **Agreement Start & End Date** should have the dd.mm.yyyy format

3. The **File Name** shouldn’t contain the file extensions

5. Evaluation Criteria

1. Per field Recall (Training data) should be greater than 90%
2. Per field Recall (Validation Data) should be greater than 80%

Recall here refers to (Per Field)

- True = Number of exact value matches for a document’s metadata given in the training/validation set to the extracted value by the system.
- False = Number of Did not match or Not Extracted
- Recall = (True)/(True + False)