Project Description:

This project tries to cover the knowledge of Python, data cleaning, model training and clustering. We will get pdf files from smart city, and process it with clustering model.

The main steps are data cleaning, model training and select the best model. The functions will be written in a project3.ipynb and project3.py. We will write test files about the functions from project3.py.

Data cleaning. We will do data cleaning in project3.py as a function called data\_clean. Firstly, we load the files with pdf format as raw data and use PdfReader to extract raw text from pdf files. Then we use dataframe to store city information and raw text. After that, we do data cleaning with nltk in a simple way as normalize\_document to delete abnormal character. As the way in contractions.py and text\_normalizer.py takes too much time, I just discard it and use a simple way instead.

Model training. We will train the clustering model for Kmeans, DBSCAN and  Hierarchical with different k and will choose the best model.

How to run

pipenv run python project3/project3.py --document city.pdf –-summarize --keywords

Functions

project3.py \

get\_wordnet\_pos ()- this function changes pos\_tag to adapt WordNetLemmatizer. Input is word and output is the part of speech.

normalize\_document ()- this function normalizes the doc, it discards special characters and change the doc to lower case and discard stop words. The input is doc and processed doc is returned.

get\_raw\_data ()- this function reads pdf and return the raw data of pdf.

data\_clean()-this function cleans the raw\_data with normalize\_document.

summarize()- this function gets summary and keywords of text.

parse\_config()-this function parse the command line to get config information and return a config dict.

test\_get\_raw\_data.py \

test ()- this function will test get\_raw\_data function from project3.py.

test\_data\_clean.py \

test()- this function will test data\_clean function from project3.py.

test\_summary.py \

test()- this function will test summary function from project3.py.

Bugs and Assumptions

Bugs:

When we test data clean functions from contractions.py and text\_normalizer.py, it takes too much time.

Assumptions:

As for data cleaning, we can get a better data cleaning method if we can process the data faster. We can delete some common words. The result of pdf reading will affect the result either.