Disaster Response Pipeline Project

Github Link: Disaster Response Code

1. Installation

I use python 3.5 to create this project and the main libraries I used are:

- sikit-learn ==0.19.1
- nltk == 3.2.5
- Flask==1.0.2
- gunicorn==19.9.0
- numpy==1.15.0
- pandas==0.23.4
- plotly==3.3.0
- sqlalchemy==1.2.12
- jsonschema==2.6.0
- Please check detailed version information in requirement.txt.

---

2. Project Motivation

This web application is a project in Udacity Data Scientist Nanodegree program. In this project, I analyze disaster data to build a model for an API that classifies disaster messages. I find a data set containing real messages that were sent during disaster events.

I create a machine learning pipeline to categorize these events so that I can send the messages to an appropriate disaster relief agency. I split the data into a training set and a test set. Then,I create a machine learning pipeline that uses NLTK, as well as scikit-learn's Pipeline and GridSearchCV to output a final model. Finally, I export the model to a pickle file.

The project also include a web app with bootstap and Flask where an emergency worker can input a new message and get classification results in several categories. disaster graph2 The web app also displays visualizations of the data as follows: disaster graph1

3. File Descriptions

```
project
│   README.md
│
└───app
│   │   run.py
│   │
│   └───templates
│       │   go.html
│       │   master.html
```

```
|
└───data
|     |   disaster_categories.csv
|     |   disaster_messages.csv
|     |   DisasterResponse.db
|     |   ETL Pipeline Preparation.ipynb
|     |   process_data.py
|
└───models
|     |   classifier.pkl
|     |   ML Pipeline Preparation.ipynb
|     |   train_classifier.py
```

4.Instructions:

1. Run the following commands in the project's root directory to set up your database and model.

    - To run ETL pipeline that cleans data and stores in database `python data/process_data.py data/disaster_messages.csv data/disaster_categories.csv data/DisasterResponse.db`
    - To run ML pipeline that trains classifier and saves `python models/train_classifier.py data/DisasterResponse.db models/classifier.pkl`

2. Run the following command in the app's directory to run your web app. `python run.py`

3. Go to http://0.0.0.0:3001/
