experiments

June 3, 2024

0.0.1 Stage 1: Data Ingestion

- □ Define Configuration for connecting with Kaggle Account
- □ Download Kaggle Dataset using Kaggle Credentials and Public API
- □ Extract Data

[1]:

```
import os
os.chdir('../')
print(f'Current Working Directory: {os.getcwd()}')
```

```
Current Working Directory: /Users/geovicco/Desktop/Codespace/dvc-
basics/DeepGlobeRoadExtraction
```

Configuration

[3]:

```
from dataclasses import dataclass
from pathlib import Path

@dataclass(frozen=True)
class DataIngestionConfig:
    # Kaggle Credentials saved in secrets.yaml
    username: str
    token: str
    # Config.yaml
    root_dir: Path
    kaggle_dataset_id: str
    download_dir: Path

from DeepGlobeRoadExtraction import CONFIG_FILE_PATH, SECRETS_FILE_PATH
from DeepGlobeRoadExtraction.utils.common import read_yaml, create_directories

class ConfigurationManager:
    def __init__(self, config_filepath=CONFIG_FILE_PATH,␣
 ↪secrets_filepath=SECRETS_FILE_PATH) -> None:
        self.config = read_yaml(config_filepath)
        self.secrets = read_yaml(secrets_filepath)
        create_directories([self.config.data_ingestion.root_dir])

    def get_data_ingestion_config(self) -> DataIngestionConfig:
        config = self.config.data_ingestion
        secrets = self.secrets.kaggle
        cfg = DataIngestionConfig(
            username=secrets.username,
            token=secrets.token,
            root_dir=Path(config.root_dir),
            kaggle_dataset_id=config.kaggle_dataset_id,
            download_dir=Path(config.download_dir)
        )
        return cfg
```

[4]:
```
# Check Configuration
config = ConfigurationManager().get_data_ingestion_config()
config
```

```
[2024-06-03 00:02:47,348: INFO: common: yaml file: config.yaml loaded
successfully]
[2024-06-03 00:02:47,351: INFO: common: yaml file: secrets.yaml loaded
successfully]
[2024-06-03 00:02:47,351: INFO: common: created directory at: data]
```

[4]:
```
DataIngestionConfig(username='adityasharma47',
token='077f426e4ed99cebc79ad82781eab4b8', root_dir=PosixPath('data'),
kaggle_dataset_id='balraj98/deepglobe-road-extraction-dataset',
download_dir=PosixPath('data/deepglobe-road-extraction-dataset'))
```

Components

[25]:
```
import os
import subprocess
import json
from DeepGlobeRoadExtraction import logger
from kaggle.api.kaggle_api_extended import KaggleApi

class DataIngestionComponent:
    def __init__(self, config: DataIngestionConfig) -> None:
        self.config = config

    # Initialise Kaggle API
    def kaggle_init(self):
        logger.info(f'---------- Initialising Kaggle Account ----------')
        KAGGLE_CONFIG_DIR = os.path.join(os.path.expandvars('$HOME'), '.kaggle')
        KAGGLE_CONFIG_FILE = os.path.join(KAGGLE_CONFIG_DIR, 'kaggle.json')

        # Check if the kaggle.json file already exists and is not empty
        if os.path.exists(KAGGLE_CONFIG_FILE) and os.path.
 ↪getsize(KAGGLE_CONFIG_FILE) > 0:
```

```
            logger.warning(f'---> Kaggle Account Credentials Found!␣
↪{KAGGLE_CONFIG_FILE}. Remove this file and re-initialise if API token is␣
↪invalid or has expired.')
            return

        os.makedirs(KAGGLE_CONFIG_DIR, exist_ok = True)
        try:
            username = self.config.username
            api_key = self.config.token
            api_dict = {"username":username, "key":api_key}
            with open(KAGGLE_CONFIG_FILE, "w", encoding='utf-8') as f:
                json.dump(api_dict, f)
            cmd = f"chmod 600 {KAGGLE_CONFIG_FILE}"
            output = subprocess.check_output(cmd.split(" "))
            output = output.decode(encoding='UTF-8')
        except Exception as e:
            logger.error(f'Failed to Initialise Kaggle Account!')
            raise e

    # Download Kaggle Dataset
    def download_dataset(self) -> None:
        logger.info(f'---------- Downloading Kaggle Dataset: {self.config.
↪kaggle_dataset_id} ----------')
        try:
            api = KaggleApi()
            api.authenticate()
            api.dataset_download_files(
                dataset=self.config.kaggle_dataset_id,
                path=self.config.download_dir,
                unzip=True,
                force=False
            )
            logger.info(f'---> Kaggle dataset saved to {self.config.
↪download_dir}')
        except  Exception as e:
            logger.error('Kaggle dataset download failed!')
            raise e
```

Pipeline

[26]:

```
pipeline = DataIngestionComponent(ConfigurationManager().
↪get_data_ingestion_config())
```

```
[2024-06-03 00:20:38,262: INFO: common: yaml file: config.yaml loaded
successfully]
[2024-06-03 00:20:38,264: INFO: common: yaml file: secrets.yaml loaded
successfully]
[2024-06-03 00:20:38,264: INFO: common: created directory at: data]
```

[27]:

```
pipeline.kaggle_init()
pipeline.download_dataset()
```

```
[2024-06-03 00:20:42,758: INFO: 968594923: ---------- Initialising Kaggle
Account ----------]
[2024-06-03 00:20:42,759: WARNING: 968594923: ---> Kaggle Account Credentials
Found! /Users/geovicco/.kaggle/kaggle.json. Remove this file and re-initialise
if API token is invalid or has expired.]
[2024-06-03 00:20:42,760: INFO: 968594923: ---------- Downloading Kaggle
Dataset: balraj98/deepglobe-road-extraction-dataset ----------]
Dataset URL: https://www.kaggle.com/datasets/balraj98/deepglobe-road-extraction-
dataset
[2024-06-03 00:26:44,810: INFO: 968594923: ---> Kaggle dataset saved to
data/deepglobe-road-extraction-dataset]
```