GTM Buddy - Data Science Assignment

Submitted by – SHASHANK GAUTAM (2020B5A32378H)

DATASET USED:

A synthetic dataset is used for the task, which consist of various statements, questions, comparisons, doubts etc. related to Electrical Vehicle Market**. (200 lines of csv file is generated, multidomain and multi-regional)**

INPUT CSV (calls_dataset.csv) - [f20202378_assignment/input.csv at main · Shan212001/f20202378_assignment]

INPUT JSON (domain_knowledge.json) - [f20202378_assignment/input.json at main · Shan212001/f20202378_assignment]

1. Data Preprocessing

The EVChargingClassifier class includes a preprocess_text method that is responsible for cleaning and preparing the text data for downstream tasks.

- **Text Lowercasing**
- **Removing Non-Alphabetic Characters**
- **Tokenization**
- **Stop Word Removal**
- **Lemmatization**

2. Data Augmentation

The augment_data method is designed to address class imbalance by augmenting text samples for minority classes.

- **Class Distribution Analysis**
- **Identifying Minority Samples**
- **Augmentation Techniques**
- **Stop Word Removal**
- **Adding Augmented Samples**
- **Post-Augmentation Analysis**

MODEL USED:

Support Vector Machine (SVM) classifier within a MultiOutputClassifier framework. The pipeline included:

- TF-IDF Vectorizer:
- SVC with Linear Kernel:

Why SVM?

- High Performance on Text Classification
- Multi-Label Compatibility
- Scalability with TF-IDF

Challenges	Class Imbalance
	High Dimensionality
	Interpretability
Solutions	Data Augmentation
	Regularization
	Cross-Validation
	Visualization

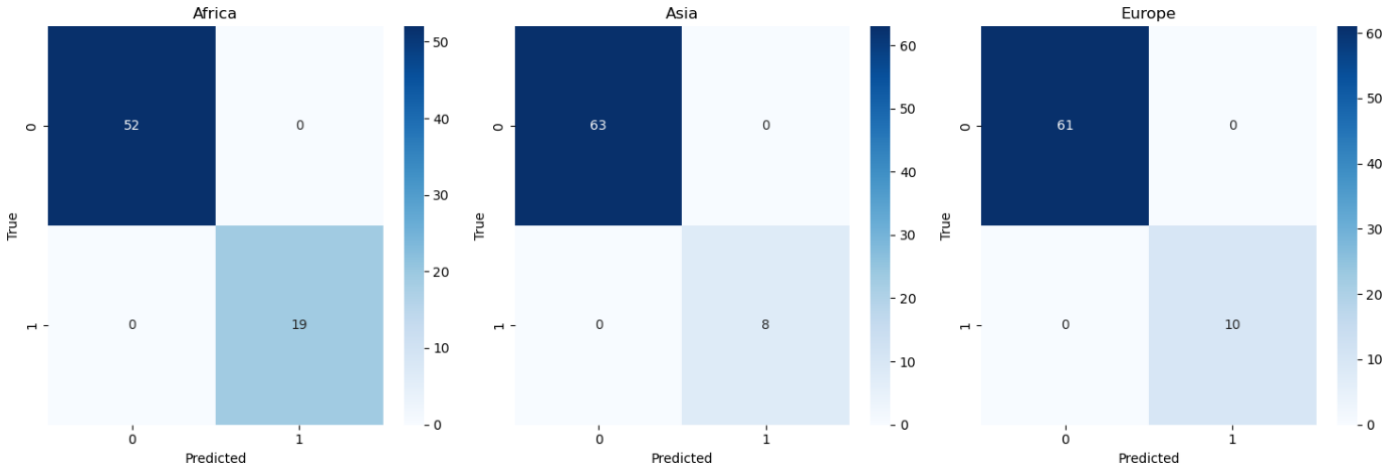
PERFORMANCE RESULTS:

Cross-Validation Scores: [0.625, 0.69642857, 0.60714286, 0.57142857, 0.67857143]

Mean Cross-Validation Accuracy: 0.6357142857142858

Metrics used for evaluation: **Confusion metrix** for all the Labels we got. (The png of all the metrics is part of the GITHUB repo).

Sample:



Link to the entire set : [f20202378_assignment/confusion_matrices_with_class_names.png at main · Shan212001/f20202378_assignment]

ERROR ANALYSIS:

A. Examples of Mistakes

- Confusion Among Labels
- Incorrect Entity Extraction
- Summarization Errors

B. Areas of Improvement

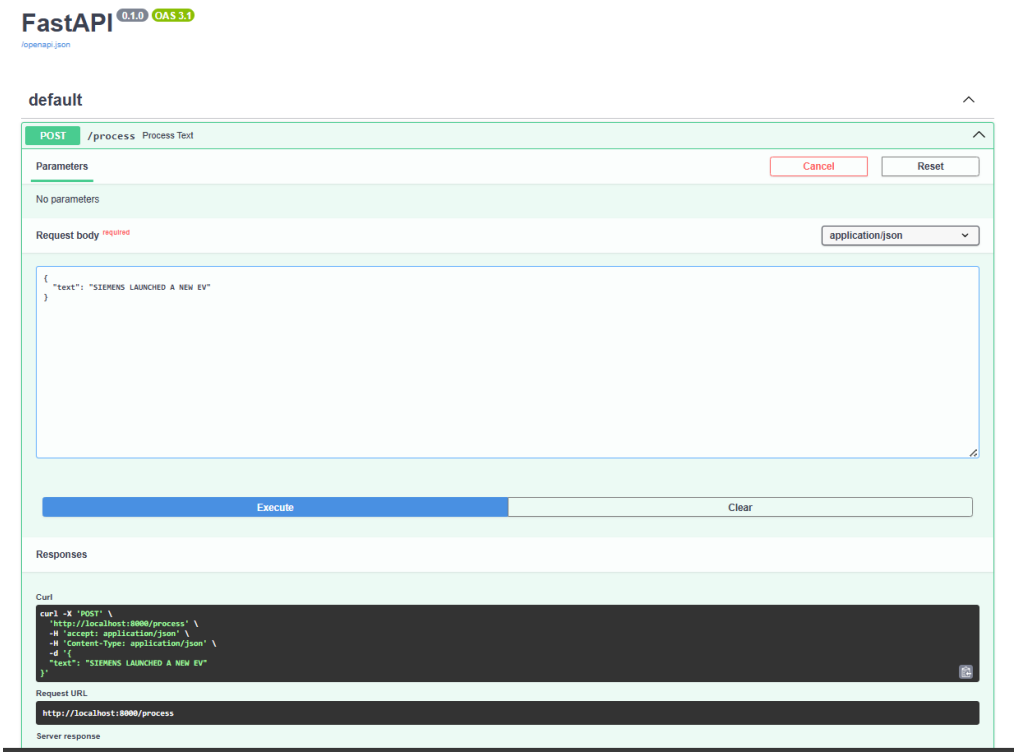
- Improved Entity Recognition
- Class Imbalance
- Summarization Enhancement

FUTURE WORK:

1. **Better Data Curation:** A more refined data can be used, also the size of the dataset can be bigger instead of 200 rows of data we used.
2. **Model Improvements:** LLM based transformer-based models like BERT or RoBERTa can be used, due to machine and gpu constraints, I was unable to use LLM based model but it can be used for better result
3. **Advanced Fine-Tuning:** Fine-tune transformer models (e.g., T5, BART) for abstractive summarization tailored to EV-related texts.
4. **Deployment Enhancements:** Implement scalable cloud-based deployment using services like AWS Lambda or Google Cloud Run.

HOW TO USE THE NLP PIPELINE?

- The deployment steps have been given in the GITHUB readme.
- Using the FastApi Ui also we can generate response using the localhost @ port 8000.



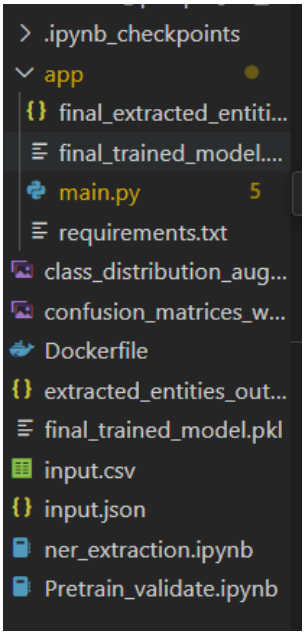
CURL COMMAND:

```
curl -X 'POST' \
  'http://localhost:8000/process' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d '{
"text": "SAMPLE_TEXT"
}'
```

Link to the GITHUB repo: [Shan212001/f20202378_assignment]

Link to the Google collab: [Preprocess_train_validate.ipynb - Colab]

Link to Host the FastAPI: http://localhost:8000/docs



NOTE:

A. *Deployment on the Heroku required a paid subscription, because of which I hosted the pipeline on the FastAPI.*
B. *While running the code on any other device please maintain the directory structure*