**README**

**Overview**

This repository contains multiple Jupyter Notebooks and Python scripts focused on hazard analysis, product categorization, and ensemble modeling. The provided notebooks and scripts implement various machine learning techniques, including transformer-based models and ensemble methods, to classify food hazards and product categories. Below is a comprehensive description of their roles, functionality, and key components.

**Notebooks and Scripts**

1. **CICLe\_Haz\_Prod\_Cat.ipynb**

**You are required to have a Hugging Face API Key to run this.**

**Purpose:** This notebook focuses on analyzing hazards and product categorization. It processes input data related to these domains and implements specific techniques to extract insights or classify information.

**Structure:**

* **Code Cells:** 17
* **Markdown Cells:** 0

**Primary Operations:**

* Library imports for data handling and modeling.
* Preprocessing steps for cleaning or organizing the dataset.
* Functions for categorization, analysis, or visualization.

**Example Code Features:**

* Import statements for libraries such as pandas, numpy, or custom tools.
* Function definitions related to hazard analysis or categorization workflows.
* Data handling or feature engineering logic.

**2. Ensemble\_Combined.ipynb**

**Purpose:** This notebook implements ensemble modeling techniques, combining outputs from various models to achieve better predictive performance. It evaluates these ensemble methods against standalone models.

**Structure:**

* **Code Cells:** 37
* **Markdown Cells:** 0

**Primary Operations:**

* Loading and preparing datasets for ensemble modeling.
* Training and combining models using methods such as stacking, bagging, or boosting.
* Evaluation of ensemble models using performance metrics.
* Visualizations or comparisons of ensemble versus individual models.

**Example Code Features:**

* Data preprocessing and splitting into training and testing sets.
* Implementation of ensemble techniques like Random Forests, Gradient Boosting, or custom combinations of model outputs.
* Libraries such as scikit-learn for model training and evaluation.

**3. Ensemble\_Haz\_Prod\_Cat.ipynb**

**Purpose:** This notebook integrates ensemble modeling techniques into hazard and product categorization tasks. It applies advanced ensemble methods specifically tailored to categorize products based on hazard-related data.

**Structure:**

* **Code Cells:** 36
* **Markdown Cells:** 0

**Primary Operations:**

* Dataset preparation for hazard and product classification.
* Training and evaluating ensemble models for classification tasks.
* Incorporating domain-specific logic for handling hazard-related data.

**Example Code Features:**

* Data transformations for hazard/product classification tasks.
* Application of ensemble models, potentially fine-tuned for these use cases.
* Use of metrics such as accuracy, F1 score, or AUC to assess model performance.

**4. Ensemble\_Haz\_Prod\_Cat.ipynb**

**You are required to upload the file of “data.csv” as well for this run and execute.**

**Purpose:** This script trains and evaluates transformer-based models for classifying food hazard categories and product categories. It includes data preprocessing, model training, evaluation, and prediction steps.

**Key Components:**

* **Libraries and Dataset Download:** Imports essential libraries and downloads datasets from a GitHub repository.
* **Data Preprocessing:** Loads the training data, combines columns, drops unnecessary columns, and splits the dataset into chunks.
* **Dataset Class:** Defines a custom FoodHazardDataset class for handling the dataset.
* **Model Definition:** Defines a TransformerForFoodHazardClassification class, initializing a transformer model and classifiers.
* **Training and Evaluation:** Defines a compute\_metrics function, implements a data\_collator function, and trains the model.
* **Prediction and Analysis:** Loads the test data, gets predictions, averages logits, decodes predicted labels, and saves results to CSV files.

**Outputs:**

* Training and evaluation results, including loss, accuracy, and F1 scores.
* Prediction results saved to CSV files.

**5. Baseline\_NLP\_Project\_Subtask1\_Subtask2\_BERT.ipynb**

**Purpose:** This script trains and evaluates a BERT-based model for classifying food hazard categories and product categories. It includes data preprocessing, model training, evaluation, and prediction steps.

**Key Components:**

* **Libraries and Dataset Download:** Imports essential libraries and downloads datasets from a GitHub repository.
* **Data Preprocessing:** Loads the training data, combines columns, drops unnecessary columns, and splits the dataset into chunks.
* **Dataset Class:** Defines a custom FoodHazardDataset class for handling the dataset.
* **Model Definition:** Defines a BertForFoodHazardClassification class, initializing a BERT model and classifiers.
* **Training and Evaluation:** Defines a compute\_metrics function, implements a data\_collator function, and trains the model.
* **Prediction and Analysis:** Loads the test data, gets predictions, averages logits, decodes predicted labels, and saves results to CSV files.

**Outputs:**

* Training and evaluation results, including loss, accuracy, and F1 scores.
* Prediction results saved to CSV files.

**6. NLP Project Subtask 1ANN.ipynb**

**Takes the file of “final\_cleaned\_train.csv” as in input**

**Purpose:** Generating the same prediction labels using ANNs.

**Key Components:**

**• Libraries and Dataset Download**

**• Classification based on "text" only done in section 1**

**• Classification based on "title" only done in section 2**

**Outputs:**

Results printed for each cell.

•Bar charts showing overall scores for both subtasks received based on the predicted output.

Note: both subtasks graphs have been visualized in one place from prerecorded results for easier comparison.

**7. NLP Project Subtask 2ANN.ipynb**

**Takes the file of “final\_cleaned\_train.csv” as in input**

**Purpose**: Generating the same prediction labels using ANNs for sub task 2 of predicting hazard and products labels,

**Key Components:**

**• Libraries and Dataset Download**

**• Classification based on "text" only done in section 1**

**• Classification based on "title" only done in section 2**

**Outputs:**

Results printed for each cell.

• Bar charts are present in subtask 1's file showing overall scores for both subtasks received based on the predicted output

**Common Themes Across Notebooks and Scripts**

* **No Markdown Cells:** The notebooks lack explanations or documentation in Markdown format, which could make them harder to follow for new users.
* **Code-Heavy:** Each notebook consists exclusively of code cells, focusing heavily on implementation rather than description.
* **Ensemble Modeling:** The use of ensemble techniques is prominent, highlighting a shared theme of improving model accuracy via combined approaches.

**Recommendations**

* **Add Documentation:** Include Markdown cells within the notebooks to explain the purpose, methodology, and outcomes of each section.
* **Comment Code:** Ensure that each code cell is well-commented to clarify its functionality.
* **Visualization Outputs:** If applicable, include plots or charts to illustrate key findings and make the outputs more interpretable.
* **Dependencies:** Clearly list dependencies (e.g., required Python libraries) for seamless execution.

This README provides a high-level overview based on the notebooks' and scripts' structure. A deeper inspection of the exact code and outputs would yield additional insights to further enhance this documentation.