Report

by

Healers

AI Cure - PARSEC 4.0

January 2024

Indian Institute of Technology, Dharwad

FILE: training.ipynb

The following file has been used to train the model on the sample data as given in the problem statement of ‘AI Cure’.

Data Loading and Preprocessing:

1. **Loading the Training Data:**
    - The training data is loaded from a CSV file named "train_data.csv" using the pandas library.
2. **Dropping Unnecessary Columns:**
    - A column is said to be unnecessary if it doesn’t comprise of numerical data required for the model in its learning phase.
    - The column 'datasetId' is identified and removed from the dataset as it is considered unnecessary.
3. **Calculating the Correlation Matrix:**
    - A correlation matrix is computed for the numeric features in the dataset to explore relationships between variables.

Feature Scaling:

1. **Selecting Target Variable and Features:**
    - The target variable HR (heart rate) and specific numeric features are chosen for further processing.
2. **Standard Scaling:**
    - Min-Max scaling is applied to the selected numeric features, ensuring uniformity in their magnitudes.

Data Splitting:

- The dataset is split into training and testing sets using an 80:20 ratio, with a random seed for reproducibility.

LSTM Model Construction:

- A sequential LSTM model is constructed with 50 units, followed by a dense output layer with a single unit.
- The model is compiled using the Adam optimizer and Mean Squared Error (MSE) as the loss function.

Model Training:

- The model is trained for 50 epochs with a batch size of 5 using the training data.
- Validation data is utilized to monitor the model's performance during training.

Model Saving and Serialization:

- The trained model is saved to 'model.h5'.

- Model serialization is performed using pickle, and the serialized model is stored in a JSON file.

Model Evaluation and Visualization:

- The training and validation loss values are plotted to visualize the model's learning progression.

Prediction and RMSE Calculation:

- Predictions are generated for both training and test datasets.
- Root Mean Squared Error (RMSE) is calculated to assess the model's performance on both datasets.

Results:

- The model achieves an RMSE of 1.3599 on the training data and 1.35096 on the test data.

Conclusion:

- The LSTM model demonstrates promising performance in predicting Heart Rate.
- Further optimization and exploration may enhance the model's accuracy and generalization.
- Monitoring additional metrics and exploring model interpretability can improve practical utility.

FILE: run.py

This file will be used for executing the input data set file (.csv format) as provided by the user, who can edit the name of the file in the code or use the standard name ‘sample_test_data.csv’.

Model Execution on Test Data

1. Loading Test Data:

- The test data is read from the "sample_test_data.csv" file.

2. Dropping Unnecessary Columns

- Unnecessary columns, specifically 'datasetId,' are removed from the test data, similar to the one we did in the training file.

3. Loading Trained Model:

- The LSTM model, previously trained and saved in 'model.h5,' is loaded for making predictions.

4. Loading Scaler Parameters:

- Scaler parameters, saved during training in 'scaler_params.pkl,' are loaded to ensure consistent scaling of the test data.

5. Preprocessing Test Input Data:

- *Selecting Numeric Features*
  - Numeric features are selected from the test data.
- *Scaling Test Data*
  - The numeric features are scaled using the scaler parameters obtained from the training data.

5. Reshaping and Making Predictions:

- The test data is reshaped to match the model's input shape, and predictions are made using the loaded LSTM model.

6. Creating Predictions DataFrame:

- A new DataFrame is created, including 'uuid' and corresponding predictions.

7. Results and Output:

- The predictions are saved to a CSV file named 'results.csv' for further analysis and comparison.

8. Sample Predictions:

- The first few rows of the predictions DataFrame are printed for a quick overview.