Smart home Energy Consumption prediction Using Deep Learning

Presented by
Bala Manikanta Sai
Keerthi Priya
Ashritha

Abstract

Our project focuses on developing a deep learning model to predict energy usage in a smart home setting by analyzing environmental factors such as temperature, humidity, and other weather-related data. By leveraging the Smart Home dataset, you aim to improve energy efficiency and provide insights into how external conditions influence power consumption. The model could potentially help in optimizing energy use, leading to cost savings and a more sustainable living environment

About the Dataset

columns

time, Use[kw], gen [kw], House Overall [kw], Dishwasher [kw], Furnance1, Furnance2, Home Office[kw], Fridge[kw], Wine Cellar[kw], Garage door [kw], kitchen 12[kw], kitchen 14 [kw], kitchen 38[kw], Barn [kw], Well[kw], Microwave [kw], Living room [kw], Solar[kw], temperature, icon, humidity, visibility, summary, apparent Temperature, pressure, windspeed, CloudCover, wind Bearing, precipintensity, dewPoint, precipprobability.

Total number of useful parameters: 30

Inputs

Temperature, icon , humidity , visibility , summary , apparentTemperature,pressure,windspeed, CloudCover,windBearing,preciipintensity,dewPoint, precipprobability

Total number of inputs :13

Outputs

gen[kw],HouseOverall[kw],Dishwasher[kw],Furnence1, Furnance2,HomeOffice[kw],Fridge[kw],WineCellar[kw], Garage door[kw],kitchen 1[kw],kitchen2 [kw],kitchen 3[kw],Barn[kw],well[kw],Microwave[kw],Livingroom[kw], solar[kw]

Total number of outputs:17

Models taken :

- Long Short Term Memory(LSTM)
- Bidirectional-Long Short Term Memory(Bi-LSTM)

Customized models:

- Convolutional Recurrent Neural Network(CRNN)
- Recurrent Generative Adverisal Network(RGAN)

Implementation Steps for our Models:

1) **Data Loading**
- Load the dataset:(CSV).
- Explore the dataset to understand its structure and key attributes.
- Data Cleaning

2) **Handle missing values**
- Remove or replace invalid data points.
- Feature Selection

3) **Select critical and relevant features for training.**
- Divide features into input (e.g., weather or critical columns) and target labels.
- Data Transformation

4) **Normalize or standardize the data (e.g., MinMaxScaler)**
- Ensure values fall within the range required for LSTM inputs (typically [0, 1]).

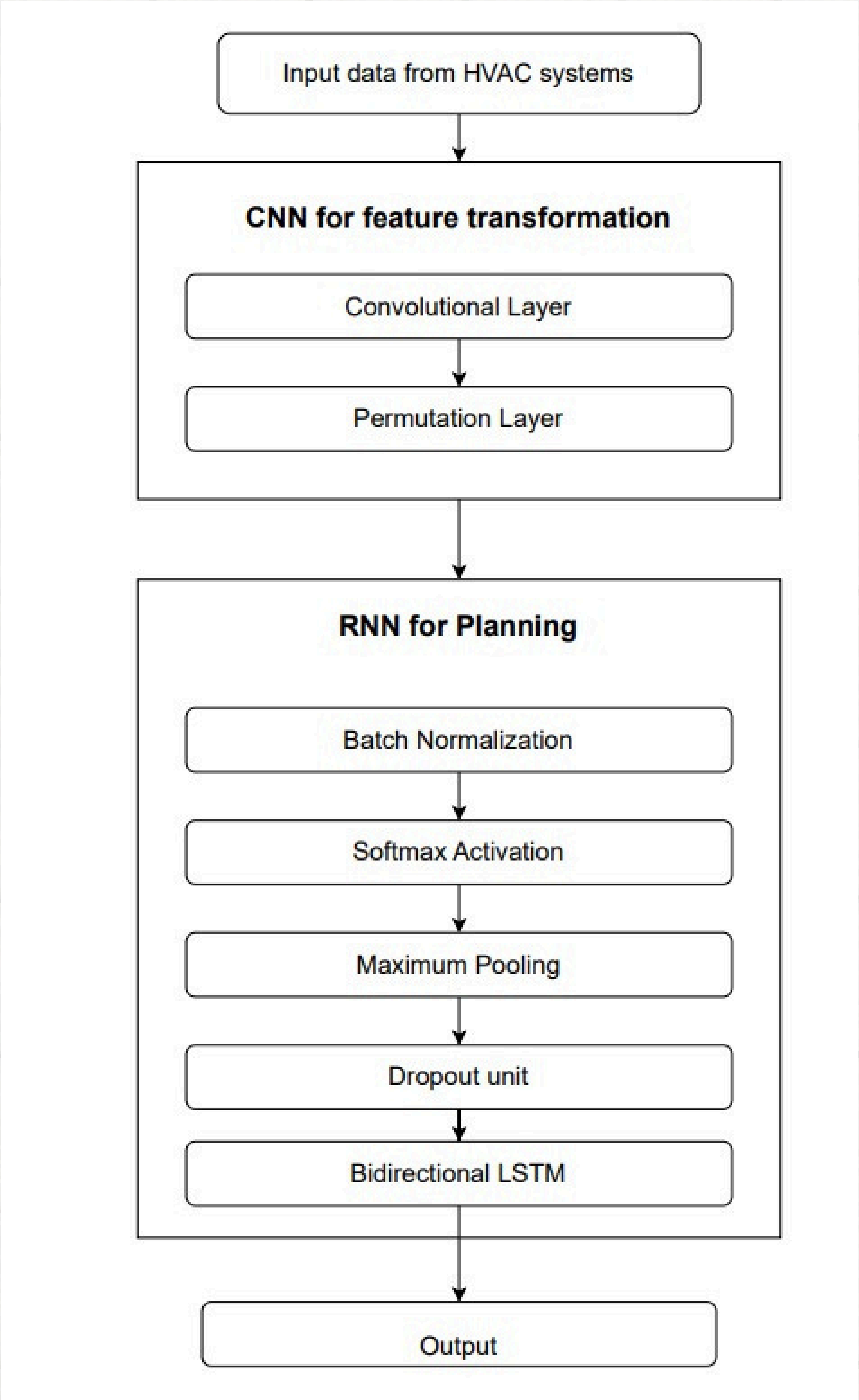
5) **Sequence Generation**
- Reshape the data into sequences (e.g., rolling windows of time steps).

6) **Define sequence length (number of past observations as input for predictions).**
- Data Splitting
- Split the dataset into training, validation, and test sets.
- Maintain time-ordering to prevent look-ahead bias.

7) **Training and validation**
- Trained the models on the training dataset and evaluated their performance using the validation set to monitor generalization.

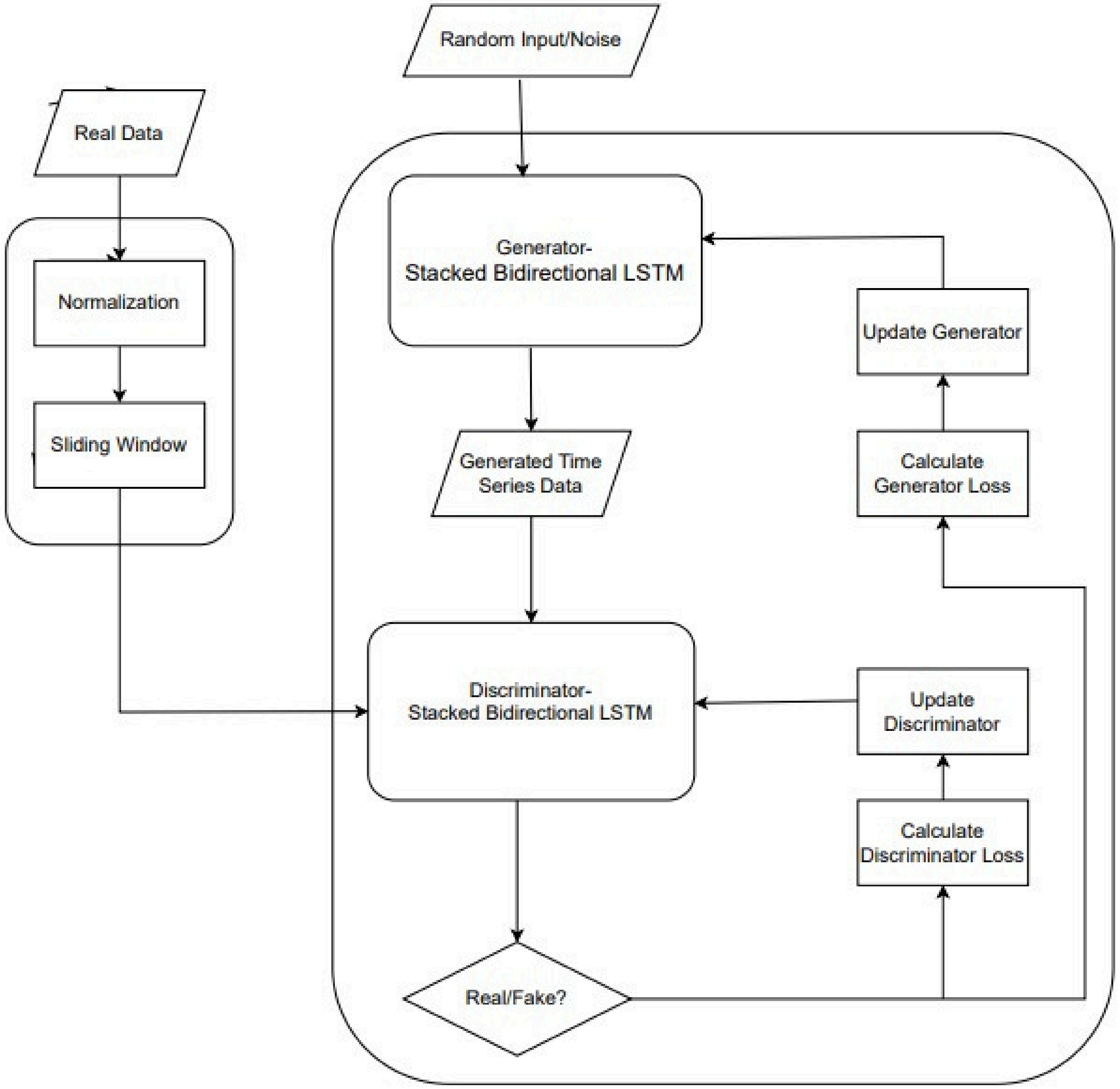
Customized model 1

Convolutional Recurrent Neural Network(CRNN)



Customized model 2

Recurrent Generative Adverisal Network(RGAN)



Bidirectional Recurrent Generative Adversarial Network

Results

TEST LOSS COMPARISON ACROSS MODELS

Model Name	Test Loss
LSTM	0.0292
Bi-directional LSTM	0.0158
CRNN	0.0758
BiRGAN (Proposed Model)	0.0076

Thank you very much!