

Постановка задачи.

Вариант 7

Необходимо построить рекуррентную нейронную сеть, которая будет прогнозировать значение некоторого периодического сигнала. Необходимо:

- Преобразовать последовательность в датасет, который можно подавать на вход нейронной сети (можно использовать функцию `gen_data_from_sequence` из примера);
- Разбить датасет на обучающую, контрольную и тестовую выборку;
- Построить и обучить модель;
- Построить график последовательности, предсказанной на тестовой выборке (пример построения также есть в примере).

Выполнение работы.

Функцией `gen_sequence` из задания генерируется последовательность. Пример приведен на рисунке 1.

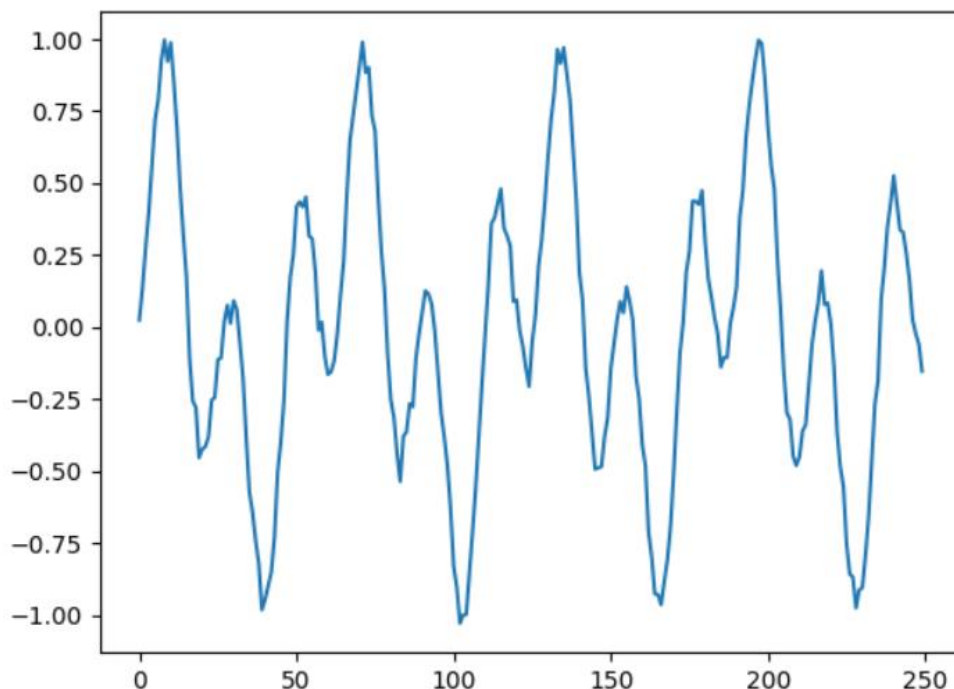


Рисунок 1. Сгенерированная последовательность

Функция `gen_data_from_sequence` из примера преобразует последовательность в датасет:

```
def gen_data_from_sequence(seq_len=1006, lookback=10):  
    seq = gen_sequence(seq_len)  
    past = np.array([[[seq[j]] for j in range(i, i+lookback)] for i in  
range(len(seq) - lookback)])  
    future = np.array([[seq[i]] for i in range(lookback, len(seq))])  
    return past, future
```

Датасет разбивается на обучающую, валидационную и тестовую выборку в пропорциях 70%, 15%, 15%:

```
dataset_size = len(data)  
train_size = (dataset_size // 10) * 7  
val_size = (dataset_size - train_size) // 2  
  
train_data, train_res = data[:train_size], res[:train_size]  
val_data, val_res = data[train_size:train_size+val_size],  
res[train_size:train_size+val_size]  
test_data, test_res = data[train_size+val_size:],  
res[train_size+val_size:]
```

Построена модель сети:

```
model = Sequential()  
model.add(layers.GRU(32, recurrent_activation='sigmoid',  
input_shape=(None, 1), return_sequences=True))  
model.add(layers.LSTM(32, activation='relu', input_shape=(None, 1),  
return_sequences=True, dropout=0.2))  
model.add(layers.GRU(32, input_shape=(None, 1), recurrent_dropout=0.2))  
model.add(layers.Dense(1))
```

Замена слоя LSTM на GRU или наоборот не влияют значительно на итоговую точность, так же, как и добавление дополнительного слоя, поэтому сохранена изначальная архитектура сети.

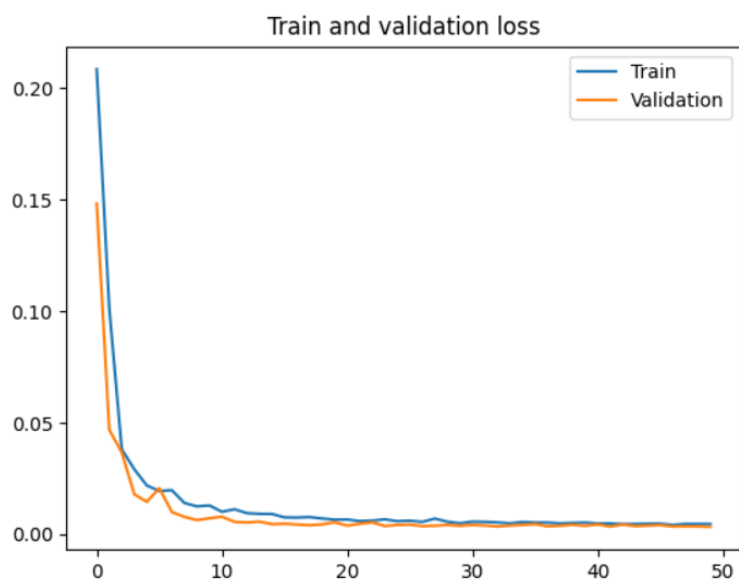


Рисунок 2. Ошибки сети на обучающих и контрольных данных

На рисунке 3 представлены предсказанная и ожидаемая последовательности. Наибольшие отклонения находятся в точках перегиба.

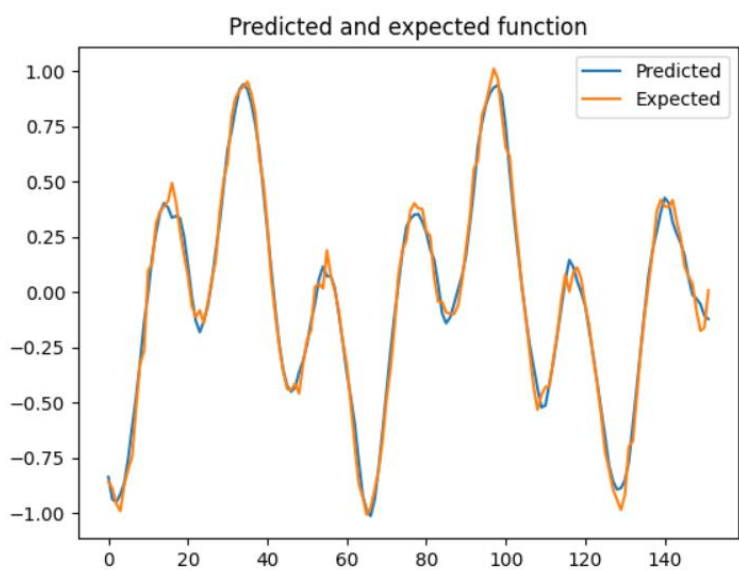


Рисунок 3. Предсказанная и ожидаемая последовательности