

- ◆ Introduction (Medium Summary)

GRU (Gated Recurrent Unit) is an improved version of RNN that solves the vanishing gradient problem.

It has:

Update gate

Reset gate

Unlike LSTM, it does not have separate memory cell state. It is simpler and faster while still handling long-term dependencies better than SimpleRNN.

```
# =====
# GRU BASED TEXT GENERATION (WORD LEVEL)
# =====

import numpy as np
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Embedding, GRU, Dense

# -----
# 1. Load Dataset
# -----


text = """
artificial intelligence is transforming modern society
it is used in healthcare finance education and transportation
machine learning allows systems to improve automatically with experience
data plays a critical role in training intelligent systems
large datasets help models learn complex patterns
deep learning uses multi layer neural networks
neural networks are inspired by biological neurons
each neuron processes input and produces an output
training a neural network requires optimization techniques
gradient descent minimizes the loss function
"""

# -----
# 2. Tokenization
# -----


tokenizer = Tokenizer()
tokenizer.fit_on_texts([text])
total_words = len(tokenizer.word_index) + 1

input_sequences = []

for line in text.split("\n"):
    token_list = tokenizer.texts_to_sequences([line])[0]
    for i in range(1, len(token_list)):
        n_gram_sequence = token_list[:i+1]
        input_sequences.append(n_gram_sequence)

max_seq_len = max([len(seq) for seq in input_sequences])
input_sequences = pad_sequences(input_sequences, maxlen=max_seq_len, padding='pre')

X = input_sequences[:, :-1]
y = input_sequences[:, -1]

# -----
# 3. Build GRU Model
# -----


model = Sequential([
    Embedding(total_words, 64, input_length=max_seq_len-1),
    GRU(128),
    Dense(total_words, activation='softmax')
])

model.compile(loss='sparse_categorical_crossentropy',
              optimizer='adam',
              metrics=['accuracy'])

model.summary()

# -----
# 4 Train Model
```

```
# -----  
# 5. Text Generation  
# -----  
  
def generate_text(seed_text, next_words=20):  
    for _ in range(next_words):  
        token_list = tokenizer.texts_to_sequences([seed_text])[0]  
        token_list = pad_sequences([token_list], maxlen=max_seq_len-1, padding='pre')  
  
        predicted = np.argmax(model.predict(token_list, verbose=0), axis=-1)  
  
        output_word = ""  
        for word, index in tokenizer.word_index.items():  
            if index == predicted:  
                output_word = word  
                break  
  
        seed_text += " " + output_word  
  
    return seed_text  
  
print("\nGenerated Text:\n")  
print(generate_text("artificial intelligence", 20))
```


Model: "sequential_3"

Layer (type)	Output Shape	Param #
embedding_3 (Embedding)	?	0 (unbuilt)
gru_3 (GRU)	?	0 (unbuilt)
dense_3 (Dense)	?	0 (unbuilt)

Total params: 0 (0.00 B)

Trainable params: 0 (0.00 B)

Non-trainable params: 0 (0.00 B)

Epoch 1/100

3/3 2s 17ms/step - accuracy: 0.0000e+00 - loss: 4.1911

Epoch 2/100

3/3 0s 15ms/step - accuracy: 0.0580 - loss: 4.1778

Epoch 3/100

3/3 0s 16ms/step - accuracy: 0.2095 - loss: 4.1677

Epoch 4/100

3/3 0s 18ms/step - accuracy: 0.1162 - loss: 4.1592

Epoch 5/100

3/3 0s 16ms/step - accuracy: 0.1124 - loss: 4.1518

Epoch 6/100

3/3 0s 16ms/step - accuracy: 0.1472 - loss: 4.1436

Epoch 7/100

3/3 0s 16ms/step - accuracy: 0.1745 - loss: 4.1337

Epoch 8/100

3/3 0s 15ms/step - accuracy: 0.1897 - loss: 4.1243

Epoch 9/100

3/3 0s 16ms/step - accuracy: 0.2830 - loss: 4.1140

Epoch 10/100

3/3 0s 16ms/step - accuracy: 0.2713 - loss: 4.1018

Epoch 11/100

3/3 0s 23ms/step - accuracy: 0.2752 - loss: 4.0846

Epoch 12/100

3/3 0s 16ms/step - accuracy: 0.2713 - loss: 4.0680

Epoch 13/100

3/3 0s 16ms/step - accuracy: 0.2287 - loss: 4.0501

Epoch 14/100

3/3 0s 16ms/step - accuracy: 0.1745 - loss: 4.0249

Epoch 15/100

3/3 0s 16ms/step - accuracy: 0.1627 - loss: 3.9917

Epoch 16/100

3/3 0s 16ms/step - accuracy: 0.1394 - loss: 3.9582

Epoch 17/100

3/3 0s 17ms/step - accuracy: 0.1668 - loss: 3.9065

Epoch 18/100

3/3 0s 16ms/step - accuracy: 0.0814 - loss: 3.8438

Epoch 19/100

3/3 0s 16ms/step - accuracy: 0.0620 - loss: 3.8250

Epoch 20/100

3/3 0s 15ms/step - accuracy: 0.0620 - loss: 3.7672

Epoch 21/100

3/3 0s 15ms/step - accuracy: 0.0852 - loss: 3.7321

Epoch 22/100

3/3 0s 15ms/step - accuracy: 0.1550 - loss: 3.6974

Epoch 23/100

3/3 0s 15ms/step - accuracy: 0.2016 - loss: 3.6741

Epoch 24/100

3/3 0s 16ms/step - accuracy: 0.1861 - loss: 3.6467

Epoch 25/100

3/3 0s 17ms/step - accuracy: 0.1706 - loss: 3.5926

Epoch 26/100

3/3 0s 16ms/step - accuracy: 0.1898 - loss: 3.5743

Epoch 27/100

3/3 0s 19ms/step - accuracy: 0.2055 - loss: 3.5488

Epoch 28/100

3/3 0s 16ms/step - accuracy: 0.1472 - loss: 3.5193

Epoch 29/100

3/3 0s 16ms/step - accuracy: 0.1939 - loss: 3.4077

Epoch 30/100

X Slightly less expressive than LSTM 0s 16ms/step - accuracy: 0.2014 - loss: 3.3841

Epoch 31/100

X Still sequential (slow compared to Transformers) 0s 16ms/step - accuracy: 0.2637 - loss: 3.3172

Epoch 32/100

X Needs sufficient data for best performance 0s 16ms/step - accuracy: 0.2364 - loss: 3.2789

Epoch 33/100

3/3 0s 15ms/step - accuracy: 0.2133 - loss: 3.2333

Epoch 34/100

3/3 0s 15ms/step - accuracy: 0.1977 - loss: 3.1948

Epoch 35/100

3/3 0s 16ms/step - accuracy: 0.2365 - loss: 3.0837

Epoch 36/100

3/3 0s 16ms/step - accuracy: 0.2556 - loss: 3.0524

Epoch 37/100

3/3 0s 17ms/step - accuracy: 0.3335 - loss: 2.9313

Epoch 38/100

3/3 0s 16ms/step - accuracy: 0.3257 - loss: 2.9167

Epoch 39/100

```
3/3 ━━━━━━━━ 0s 15ms/step - accuracy: 0.2829 - loss: 2.8554
Epoch 40/100
3/3 ━━━━━━ 0s 17ms/step - accuracy: 0.3063 - loss: 2.7525
Epoch 41/100
3/3 ━━━━ 0s 16ms/step - accuracy: 0.2869 - loss: 2.7252
Epoch 42/100
3/3 ━━━━ 0s 16ms/step - accuracy: 0.2944 - loss: 2.7199
Epoch 43/100
3/3 ━━━━ 0s 21ms/step - accuracy: 0.3529 - loss: 2.6036
Epoch 44/100
3/3 ━━━━ 0s 15ms/step - accuracy: 0.3138 - loss: 2.5669
Epoch 45/100
```