Amardeep Singh

E23CSEU2189

```
# ================= IMPORTS =================
import tensorflow as tf
import numpy as np

# ================= LOAD DATASET FROM FILE =================
with open("sample text.txt", "r", encoding="utf-8") as f:
    text = f.read().lower()

# ================= PREPROCESSING =================
chars = sorted(list(set(text)))
vocab_size = len(chars)

char_to_idx = {c: i for i, c in enumerate(chars)}
idx_to_char = {i: c for i, c in enumerate(chars)}

encoded = np.array([char_to_idx[c] for c in text])

# ================= CREATE SEQUENCES =================
seq_len = 50
X, y = [], []

for i in range(len(encoded) - seq_len):
    X.append(encoded[i:i + seq_len])
    y.append(encoded[i + seq_len])

X = np.array(X)
y = np.array(y)

# ================= RNN MODEL =================
model_rnn = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size, 64, input_length=seq_len),
    tf.keras.layers.SimpleRNN(128),
    tf.keras.layers.Dense(vocab_size, activation="softmax")
])

model_rnn.compile(
    loss="sparse_categorical_crossentropy",
    optimizer="adam"
)

# ================= TRAIN =================
model_rnn.fit(X, y, epochs=30, batch_size=64)

# ================= TEMPERATURE SAMPLING =================
def sample_with_temperature(preds, temperature=0.8):
    preds = np.asarray(preds).astype("float64")
    preds = np.log(preds + 1e-8) / temperature
    exp_preds = np.exp(preds)
    preds = exp_preds / np.sum(exp_preds)
    return np.random.choice(len(preds), p=preds)

# ================= TEXT GENERATION =================
def generate_text(seed, length=600, temperature=0.8):
    result = seed

    for _ in range(length):
        encoded_seed = [char_to_idx[c] for c in seed]
        encoded_seed = np.array(encoded_seed).reshape(1, -1)

        preds = model_rnn.predict(encoded_seed, verbose=0)[0]
        next_idx = sample_with_temperature(preds, temperature)
        next_char = idx_to_char[next_idx]

        result += next_char
        seed = seed[1:] + next_char

    return result

# ================= OUTPUT =================
# Use a seed that exists in the corpus
seed_text = text[:50]

print("\nGenerated Text (Simple RNN):\n")
print(generate_text(seed_text))
```

```
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 2.3267
Epoch 8/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 2.2049
Epoch 9/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 2.0853
Epoch 10/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 2.0108
Epoch 11/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 1.8539
Epoch 12/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.8255
Epoch 13/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.7434
Epoch 14/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.6936
Epoch 15/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.5760
Epoch 16/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.5157
Epoch 17/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.4555
Epoch 18/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.4182
Epoch 19/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.3489
Epoch 20/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.2761
Epoch 21/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 1.1845
Epoch 22/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.1582
Epoch 23/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 1.0562
Epoch 24/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 1.0060
Epoch 25/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 0.9745
Epoch 26/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 0.8970
Epoch 27/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 0.8491
Epoch 28/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 7ms/step - loss: 0.8341
Epoch 29/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 0.7562
Epoch 30/30
34/34 ━━━━━━━━━━━━━━━━━━━━ 0s 6ms/step - loss: 0.7382

Generated Text (Simple RNN):


artificial intelligence is one of the most importan s enccing anterin ancerninatificict anterensureq ore fypune
ferp datd andes and rping are to ivalicg aravloy anl, anchenpinantumon urtofiri les
teve aqronqel anengees do malelon macal inge andrate dole ce dfacitars th therutie erpones
ed.
inspiniallig
anre us breaninsf seathe thal duall witguce s dolilgare nonenteus dos. ans allagraite.
endigme tuate drage tith ied interelay callicing asevin  andunse doofire fharn, anthort mus erancece wodedles,
```