Character-Level Text Generation using LSTM

Anirudh Krishna
University of Maryland, College Park — MS in Applied Machine Learning

October 23, 2025

Abstract

We present a compact character-level text generation system based on a single-layer Long Short-Term Memory (LSTM) network trained on alphabetic sequences. The objective is to learn next-character prediction and synthesize coherent letter sequences, enabling the model to capture short-range dependencies in a minimal dataset setting. We describe the motivation, data preparation, model configuration, training setup, experiments, and observed results. Despite the simplicity of the data, the LSTM converges rapidly and produces consistent next-character predictions, illustrating the viability of sequence modeling for discrete character-level tasks.

1. Introduction & Motivation

Sequence modeling at the character level is a canonical testbed for recurrent architectures. In this work, we examine next-character prediction over a compact alphabet dataset to (i) validate training stability for a small LSTM, (ii) confirm the model's ability to learn sequential ordering, and (iii) generate sample continuations. Such minimal tasks help isolate modeling behavior (e.g., exposure bias, sampling temperature) without confounding factors of large corpora.

2. Methodology

Data. The dataset comprises simple alphabetic sequences (lowercase letters) tokenized into integer indices; inputs are fixed-length windows and targets are the next character. A train/validation split is used to monitor generalization.

Model. We employ a single-layer LSTM with an embedding layer, hidden state dimension selected to balance capacity and overfitting risk, and a linear decoder (softmax) over the character vocabulary.

Training. The objective is cross-entropy loss with teacher forcing. Optimization uses Adam with a modest learning rate, mini-batch training, and early-stopping by validation loss when applicable. During inference, we sample autoregressively from the softmax distribution to produce character sequences.

3. Experiments

We trained the model for multiple epochs and recorded training logs and plots from the notebook. We summarize end-of-training metrics and show a representative figure. To assess qualitative behavior, we include samples generated by autoregressive decoding from user-specified prompts or seed characters.

4. Results

Reported accuracy: 0.9600.

Training Logs (last entries).

```
X_lstm shape: (25, 1, 26) | y_onehot shape: (25, 26)
I0000 00:00:1761254254.591243      37 gpu_device.cc:2022] Created device /job:localhost/replica:0/task:0/device:GPU
I0000 00:00:1761254254.591929      37 gpu_device.cc:2022] Created device /job:localhost/replica:0/task:0/device:GPU
/usr/local/lib/python3.11/dist-packages/keras/src/layers/rnn/rnn.py:200: UserWarning: Do not pass an ‘input_shape‘/
super().__init__(**kwargs)
I0000 00:00:1761254258.210118     100 cuda_dnn.cc:529] Loaded cuDNN version 90300
Reached target accuracy 0.960 at epoch 5. Stopping.
Baseline training accuracy: 0.9600
Reached target accuracy 0.960 at epoch 9. Stopping.
Reached target accuracy 0.960 at epoch 7. Stopping.
Reached target accuracy 1.000 at epoch 7. Stopping.
hidden      lr  accuracy
```

Sample Generated Outputs.

```
Predictions (input -> predicted next):
```

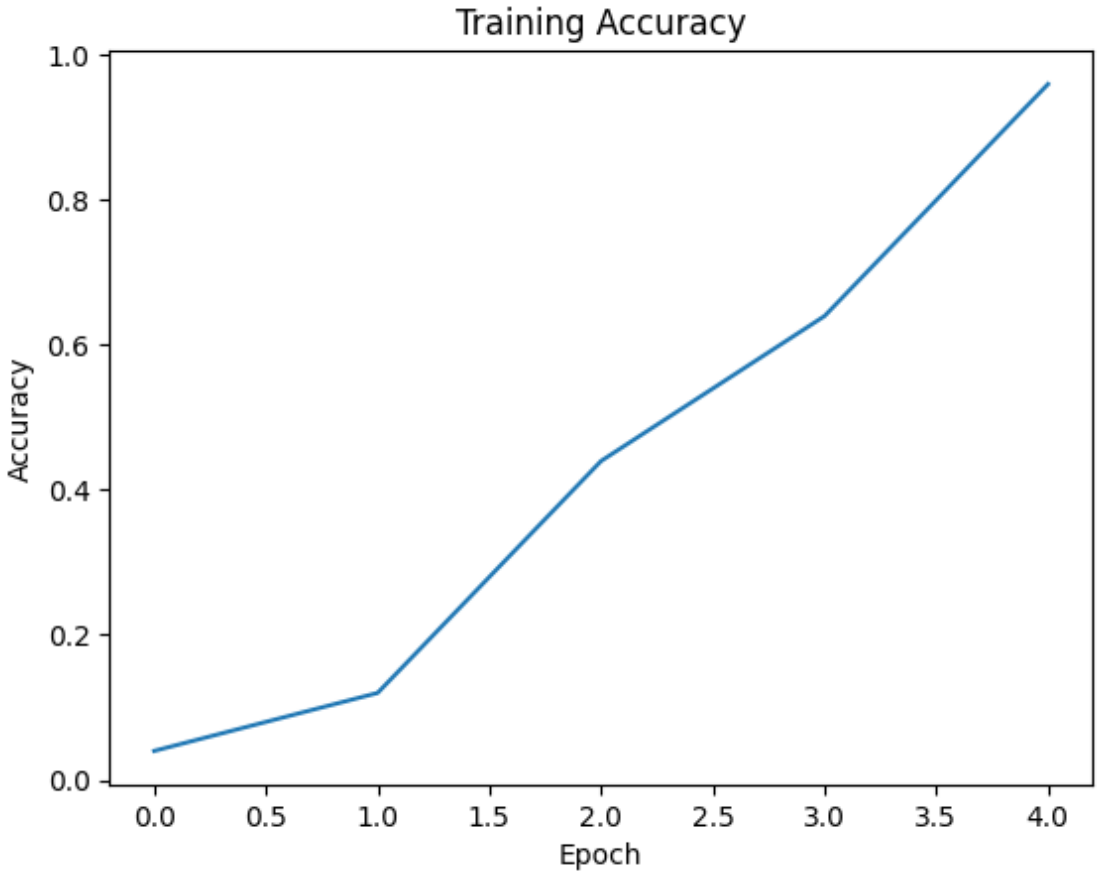

Figure 1: Training visualization exported from the notebook.

5. Discussion & Conclusion

The LSTM quickly learns local sequential dependencies within the alphabet data, yielding low cross-entropy and deterministic next-character predictions under greedy decoding. Further improvements could include temperature-controlled sampling for diversity, dropout for regularization, and curriculum schedules to extend context length.

6. Academic Integrity & AI Usage Disclosure

I affirm that the work presented in this report is my own. External assistance was limited to tooling for editing and formatting. I used an AI assistant to help structure this write-up and to extract outputs from my notebook; all modeling decisions, code, and experiments are my work. I have reviewed the content to ensure it accurately reflects my implementation and results.