Word-Level LSTM Model for Sentence Completion using Shakespeare's Plays

Afaq Alam
B.Sc Data Science
NUCES-FAST
Islamabad, Pakistan

***Abstract*—This paper presents a word-level Long Short-Term Memory (LSTM) model trained on Shakespeare's plays to predict the next word in a sequence. The model is trained using TensorFlow and Keras on a dataset containing Shakespearean dialogues. The trained model is integrated with a user-friendly interface that provides real-time word suggestions. The report discusses the preprocessing steps, model architecture, results, and challenges encountered during implementation.**

***Index Terms*—Long Short-Term Memory, Natural Language Processing, Shakespeare, Autocomplete**

I. Introduction

Sentence completion is an important task in natural language processing (NLP), enabling applications such as autocomplete and text generation. In this project, a word-level LSTM model is trained on Shakespeare's plays to predict the next word given a sequence of words. The model dynamically updates its predictions as a user types, offering real-time word suggestions. The study also explores how hyperparameters affect the model's accuracy and evaluates the coherence of generated sentences.

II. Methodology

A. Dataset

The dataset used is the Shakespeare Plays dataset [1], which contains dialogues from various plays written by William Shakespeare. The dataset was obtained from Kaggle and includes a column 'PlayerLine' that holds the spoken lines.

B. Data Preprocessing

The preprocessing steps include:

- Loading the dataset and extracting the 'PlayerLine' column.
- Tokenizing the text using Keras' 'Tokenizer'.
- Creating sequences of 10 words each for model training.

C. Model Architecture

The LSTM model consists of:

- An Embedding layer with an input dimension equal to the vocabulary size and an output dimension of 100.
- Two LSTM layers with 150 units each and dropout layers to prevent overfitting.
- A fully connected Dense layer with 150 neurons and ReLU activation.
- A final Dense layer with softmax activation for multi-class classification.

The model was compiled using the Adam optimizer and trained using categorical cross-entropy loss. Early stopping was used to halt training if no improvement in loss was observed for five consecutive epochs.

III. Results

The model was trained for different hyperparameter settings. The comparison of models is shown in Table I.

TABLE I
Comparison of Different Hyperparameter Settings

Model	Accuracy	Loss	Epochs	Batch Size
e300_b128	0.9114	0.2939	300	128
e200_b128	0.8780	0.3959	200	128
e100_b128	0.7946	0.6956	100	128
e250_b64	0.8456	0.5232	250	64
e150_b64	0.8359	0.5627	150	64
e50_b64	0.6609	1.2622	50	64
e250_b32	0.8069	0.6803	250	32

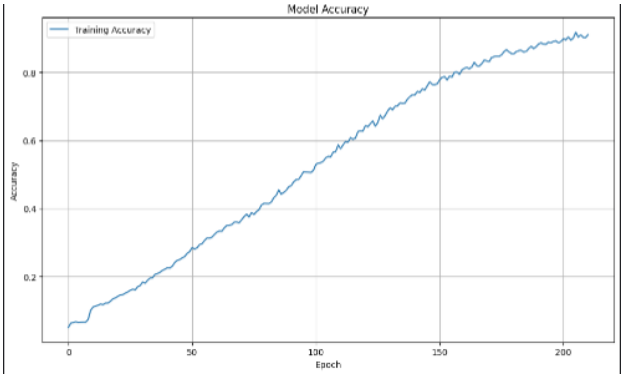

Fig. 1. Training Accuracy of Best Model (e300_b128)

IV. Discussion

The comparison of different models shows that:

- Increasing the number of epochs improves accuracy, as seen in the e300_b128 model achieving 91.14% accuracy.
- Larger batch sizes (e.g., 128) lead to faster convergence but may generalize less than smaller batch sizes.
- Lower epochs and smaller batch sizes, such as in e50_b64, result in higher loss and lower accuracy.

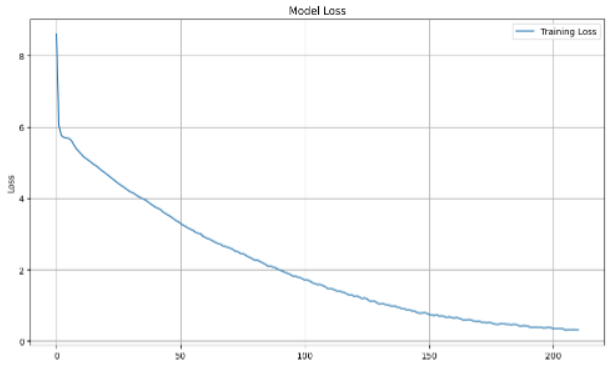

Fig. 2. Training Loss of Best Model (e300_b128)

- The e250_b64 model achieves a balance between accuracy and loss, making it a viable alternative to the highest-performing model.

Overall, a higher number of training epochs and a moderately high batch size improve model performance, but diminishing returns can occur after a certain threshold.

V. Conclusion

This project successfully implemented a word-level LSTM model for sentence completion using Shakespeare's plays. The model provides real-time word predictions, demonstrating the potential for NLP applications.

VI. Prompts

- "To be or not to be, that is the"
- "All the world's a"
- "The fault, dear Brutus, is not in our"

VII. References

References

[1] https://www.kaggle.com/datasets/kingburrito666/shakespeare-plays