INNOMATICS®
RESEARCH LABS

INNOVATION. AUTOMATION. ANALYTICS

PROJECT ON

Next Word Prediction Using LSTM

Deep-Learning based NLP Project

Done By: Anuradha K

About me

- **Background ? (B-tech or M-tech)**

B.Arch Graduate

- **Why you want to learn Data Science**

I enjoy learning new things, and data science is a field that's always growing and changing. I want to add new skills like data analysis, programming, and machine learning to my skill set.
In architecture, we often make choices based on design principles. I'm excited to learn how to make decisions based on data, which can provide more precise and impactful results.

- **Any work experience**

No work experience.

- **Share your linkedin and github profile urls**

Anuradha Kilaparthi | LinkedIn
anuradhak0801 (Anuradha K) (github.com)

PROBLEM STATEMENT AND USE CASE DOMAIN

In today's digital world, typing assistance and predictive text tools are widely used. From messaging apps to email platforms, users expect smart systems that understand context and suggest the next word or phrase seamlessly.

Problem:
Can we build a model that predicts the next word in a sentence based on previous words?

Use Case Domain:

- Natural Language Processing (NLP)
- Human-Computer Interaction
- Assistive AI (smart keyboards, virtual assistants, email writing tools)
- Content creation (auto-writing and auto-complete tools)

This problem is a foundational NLP task that supports larger goals like text generation, summarization, and conversation modeling.

OBJECTIVE

The objective of this project is to:

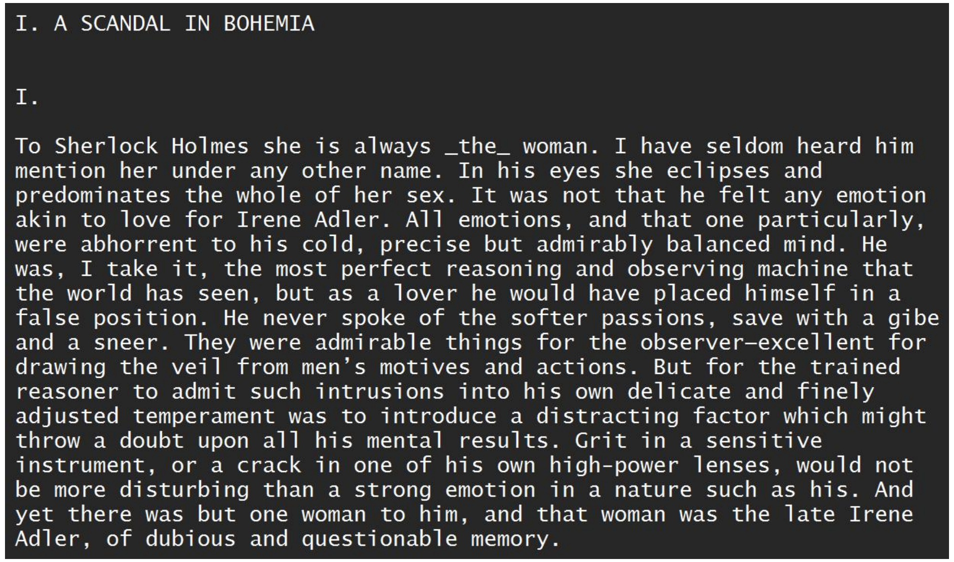
- Build an LSTM-based deep learning model to predict the next word in a sentence.
- Use a real-world English language corpus to train the model.
- Clean and preprocess the data to suit the requirements of sequence-based models.
- Evaluate and tune the model for improved accuracy and practical usability.
- Explore real-time prediction where a user inputs partial text and the model suggests the next word.

DATA OVERVIEW

Used a public domain literary text titled “The Adventures of Sherlock Holmes”, which is a rich English narrative containing various sentence structures, vocabulary, and punctuation.

Why did I choose this dataset?

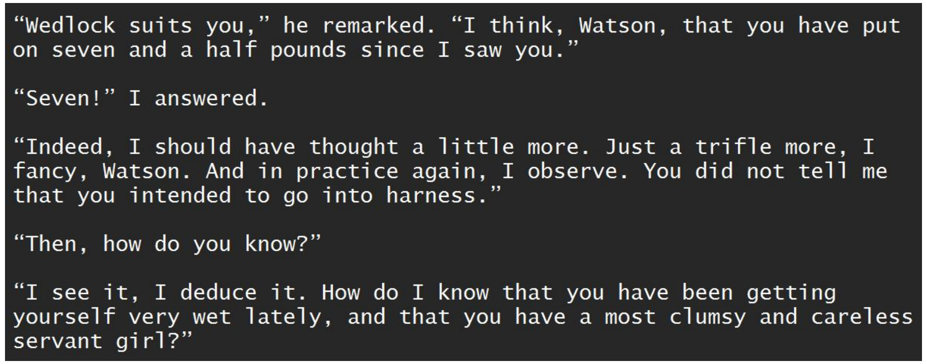
- It’s large enough to learn meaningful word patterns.
- It provides a variety of sentence structures for generalization.
- It's publicly available and contains natural language without artificial tagging.



```
I. A SCANDAL IN BOHEMIA


I.

To Sherlock Holmes she is always _the_ woman. I have seldom heard him
mention her under any other name. In his eyes she eclipses and
predominates the whole of her sex. It was not that he felt any emotion
akin to love for Irene Adler. All emotions, and that one particularly,
were abhorrent to his cold, precise but admirably balanced mind. He
was, I take it, the most perfect reasoning and observing machine that
the world has seen, but as a lover he would have placed himself in a
false position. He never spoke of the softer passions, save with a gibe
and a sneer. They were admirable things for the observer—excellent for
drawing the veil from men’s motives and actions. But for the trained
reasoner to admit such intrusions into his own delicate and finely
adjusted temperament was to introduce a distracting factor which might
throw a doubt upon all his mental results. Grit in a sensitive
instrument, or a crack in one of his own high-power lenses, would not
be more disturbing than a strong emotion in a nature such as his. And
yet there was but one woman to him, and that woman was the late Irene
Adler, of dubious and questionable memory.
```



```
“Wedlock suits you,” he remarked. “I think, Watson, that you have put
on seven and a half pounds since I saw you.”

“Seven!” I answered.

“Indeed, I should have thought a little more. Just a trifle more, I
fancy, Watson. And in practice again, I observe. You did not tell me
that you intended to go into harness.”

“Then, how do you know?”

“I see it, I deduce it. How do I know that you have been getting
yourself very wet lately, and that you have a most clumsy and careless
servant girl?”
```

Snippets of the text

DATA PRE-PROCESSING

Preprocessing Steps:

- Lowercasing – to reduce vocabulary size and unify word forms.
- Removing special characters and numbers – to keep only meaningful text.
- Sentence segmentation – split using punctuation (`.`, `!`, `?`) to ensure logical units.
- Tokenization – each word is converted to an integer using a Keras tokenizer.
- Sequence generation – n-gram style sequences are generated from each sentence.
- Padding – sequences are padded to the same length so the model receives uniform input.

Why these steps?

Text data is messy and inconsistent. Cleaning it ensures the model doesn't learn noise, while sequences and padding make it suitable for training an LSTM, which expects fixed-length numerical input.

MODEL BUILDING

Used a Sequential model with the following layers:

Input Layer: Receives a sequence of word indices that represent the words in a sentence.

Embedding Layer: Converts each token into a 100-dimensional vector to capture semantic meaning.

LSTM Layer-1: With 150 neurons.

LSTM Layer-2: With 100 neurons.

Dense Output Layer: Softmax activation outputs the probability for each word in the vocabulary as the next word.

Input shape: Fixed-length padded sequences (Longest length of sequences from training data).

Output classes: Number of unique words (vocabulary size).

Loss Function: Categorical crossentropy – best for multi-class prediction.

Optimizer: Adam – adaptive and efficient for NLP tasks.

Training Epochs: 100

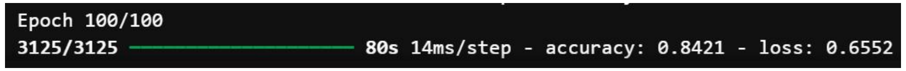
EarlyStopping: Stops training if loss does not improve for 3 epochs.

MODEL EVALUATION AND PREDICTION

After training the model:

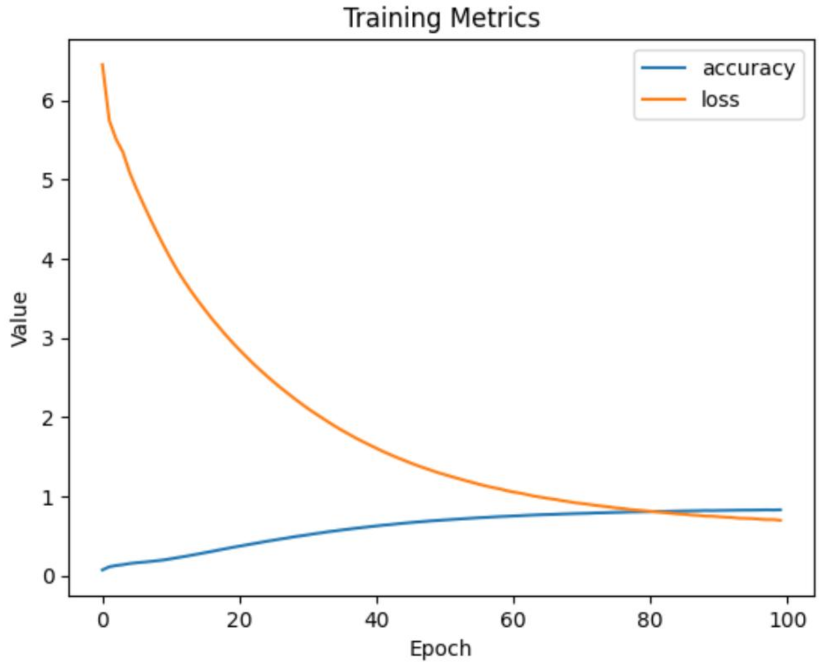
Final Accuracy: **84.2%**

Final Loss: **0.65**



```
Epoch 100/100
3125/3125 ━━━━━━━━━━━━━━━━━━━━ 80s 14ms/step - accuracy: 0.8421 - loss: 0.6552
```

This indicates the model is learning meaningful word patterns and improving its predictability.



Evaluation Observations:

- Steady learning curve
- No signs of overfitting
- Reasonable balance between complexity and performance.

The model is now capable of predicting the next word in a sentence based on previous context.

- The input will be a partial sentence (e.g. “I shall”).
- It's cleaned, tokenized, and padded to match training format.
- The model outputs a probability distribution over all vocabulary words. The word with the highest probability is selected.

Key Business Question

Can we develop a model that mimics human language understanding to improve typing and communication efficiency?

Conclusion (Key finding overall)

- LSTM can effectively model, language sequences and predict contextually appropriate words.
- Preprocessing and data structure are just as important as the model architecture.
- The model shows promising accuracy and could be integrated into auto-complete tools.
- There’s potential to improve using larger datasets, validation monitoring, or transformer models.

Your Experience/Challenges Working On The Machine Learning Project

Chellenges:

- It was hard to figure out the best way to clean the text without removing important parts of it.
- Creating sequences and training the model used a lot of memory and time. Managing that was a bit tricky.
- I wasn’t sure how many epochs would be enough. I didn’t want the model to stop too early or keep training when it didn’t need to.
- I also had trouble deciding which loss function to choose. Categorical crossentropy or Sparse categorical crossentropy.
- Understanding how the model's predictions (from softmax) work took some time.
- It was difficult to judge the accuracy since the model had to choose from so many possible words.

Your Experience/Challenges Working On The Machine Learning Project

My Experience:

- Working with language data needs both technical skills and a basic understanding of how language works. Learnt how tokenization works and how it helps the model in learning language structure.
- LSTM models are good for this kind of task, but they are sensitive to how we prepare the data.
- It’s really helpful to plot training accuracy and loss to see how well the model is learning.
- Using early stopping made the training more efficient.
- The order of preprocessing steps and how we prepare the text can make a big difference.

Q&A
?
?
?
?
?
?
?

THANK YOU

INNOMATICS
RESEARCH LABS