Question Answering Model using RNN

Team 22
Abdelrahman Mohamed Ahmed Abouelkheir 52-5388

April 2025

1 Introduction

In this milestone, we focus on building question answering model using Recurrent Neural Network by making Encoder-Decoder architecture. This model is trained using SQuAD Dataset methodology, discuss limitations, and suggest improvement

2 Methodology

2.1 Model Architecture Rationale

2.1.1 Encoder Design

- **Embedding Layer**:
  - *Purpose*: convert words into contextual representation
  - vocabulary size enables efficient matrix operations while maintaining semantic relationships
- **Single LSTM Layer**:
  - *Configuration*:
    * Tanh activation: Bounded output (-1,1) prevents gradient explosion
    * Sigmoid gates
    * Orthogonal initialization: Preserves norm during state transitions
  - *B*alance model capacity with computational efficiency for sequence encoding

2.1.2 Decoder Design

- **Embedding**:
  - Maintains consistent token representations
  - Reduces parameters while ensuring encoder-decoder semantic alignment
- **State Initialization**:
  - *Mechanism*: Inherits encoder's final hidden states
  - *Choice Rationale*: Preserves contextual information for answer generation
- **Softmax Output**:
  - *Purpose*: Generates probability distribution over vocabulary
  - *Choice Rationale*: Enables gradient-based learning for discrete outputs

2.2 Training Protocol Justification

- **Teacher Forcing**:
  - *Implementation*: Feeds ground truth tokens during training
  - *Choice Rationale*: Accelerates convergence by preventing error accumulation
- **Loss Function**:
  - *Sparse Categorical Cross-Entropy*:
    - *Advantage*: Memory-efficient for large vocabularies
- **Optimization**:
  - *Adam Optimizer*:
    - *Advantage*: Adaptive learning rates per parameter
    - *Alternative tried*: RMSprop showed slower convergence in preliminary tests

2.3 Inference Strategy

- **Autoregressive Generation**:
  - *Process*: Predicts one token per timestep
  - *Choice Rationale*: Required for open-ended sequence generation
- **State Preservation**:
  - *Mechanism*: Passes hidden states between steps
  - *Choice Rationale*: Maintains conversation history and context

2.4 Evaluation Approach

- **Token-Level Accuracy**:
  - *Purpose*: Measures exact word prediction performance
  - *Limitation*: Doesn't capture semantic correctness
- **Loss**:
  - *Purpose*: Indicates overall learning progress
  - *Interpretation*: Stable validation loss suggests proper regularization

3 Limitations

The main limitation while making this model is the lack of computational resources while testinga though kaggle was used. This increases the testing time for the model when training it. Another limitation was the cold start problem where the first generated token has no meaningful input beyond startseq. Moreover, the model needs high computational resources due to the use of large number of LSTM units.