Image Captioning using Attention Mechanism

Week 4: Learning Basics of Natural Language Processing (NLP)

Welcome to the **fourth** week (mostly the continuation of the third week) of our project.This week, we will delve into essential concepts and techniques that form the backbone of NLP applications.

What We Will Cover This Week:

1. **Introduction to NLP and Text Processing**
    - Understanding the scope of NLP and its applications in analysing and processing natural language data.
    - Exploring the challenges and opportunities presented by textual data.
2. **Text Preprocessing Techniques**
    - Learning essential preprocessing steps such as tokenization, lowercasing, removing punctuation, and handling stopwords.
    - Implementing techniques to normalise text data and prepare it for further analysis.
3. **Word Embeddings and Distributed Representations**
    - Introduction to word embeddings and their role in representing words as dense vectors in a continuous vector space.
    - Exploring popular embedding techniques like Word2Vec, GloVe, and FastText.
4. **Introduction to Sequence-to-Sequence Models**
    - Understanding the architecture and applications of sequence-to-sequence (Seq2Seq) models in NLP tasks such as machine translation and summarization.
    - Exploring the components of Seq2Seq models, including encoder-decoder frameworks and attention mechanisms.

Resources

Complete Natural Language Processing (NLP) Tutorial in Python! (with examples)

Word Embeddings

[Classic] Word2Vec: Distributed Representations of Words and Phrases and their Com…

Text Preprocessing in NLP | Python

https://www.youtube.com/live/ElmBrKyMXxs?si=Xf7nkOm2XGN2tMNe

Assignment-3: Stock Prediction using Sequential Models (Deadline: 30th June)

Problem Statement:

Develop an LSTM (Long Short-Term Memory) model to predict stock prices based on historical data. Utilise a dataset containing historical stock prices (e.g., Apple's stock prices) to construct and train the LSTM model. The objective is to build a model that can accurately predict future stock prices based on past trends and evaluate its performance using standard metrics.

Tasks:

Data:

Dataset Acquisition:

- Download a dataset containing historical stock prices (e.g., Apple's stock prices from Yahoo Finance or Kaggle).

Sequence Creation:

- Create sequences of past stock prices to predict future prices.

Model Development:

Construct the LSTM Model:

- Build an LSTM model with appropriate layers, including LSTM layers, dropout layers to prevent overfitting, and dense layers for the final output.

Model Architecture:

- Input Layer: Sequences of past stock prices.
- LSTM Layers: To capture sequential dependencies.
- Dropout Layers: To reduce overfitting.
- Dense Layer: For the final prediction.

Training the Model:

Train the LSTM Model:

- Train the model using the training data.
- Monitor training and validation loss to ensure the model is learning effectively.

Evaluation:

Evaluate the Model:

- Evaluate the model's performance on a separate test set not used during training.

- Plot training and validation loss to visualise the model's learning process.
- Make predictions using the test data and visualise the results.

Submission Instructions

Submit the code and analysis on your respective GitHub and Google Drive link as stated in the Submission Form.