Generative AI Transformer

WIDS PROJECT (UID 85)

Report[1]

Gurnoor Sohal

24b1042

🔗 Github Repository of my work

(https://github.com/G-Sohal/Generative-AI-Transformer)

Week 1:

About: Introduction to python libraries and gradient descent

Status: Done

Details:

Assignment:

- NumPy Assignment: Performed different tasks using NumPy functions for mathematical and logical operations on arrays.
- Matplotlib and Pandas Exercises: Practiced simple plotting methods and data visualization to improve familiarity with Matplotlib and Pandas.
- Gradient Descent Implementation: Optimized the given mathematical function using the gradient descent algorithm.

Resources Covered:

- Python Fundamentals: Explored py concepts like data types, data structures, loops, functions, and file handling.
- Jupyter Notebooks: Got familiarized with the fundamentals of working within Google Colab for running Jupyter Notebooks and tried its integrated computational capabilities.
- NumPy: Learned about NumPy's ability to process numerical operations effectively.
- Matplotlib: Learned about basic plotting functions and their usage in data visualization.
- Pandas: Learned Pandas data structures (Series, DataFrames) and their application in data analysis.

Key Learnings:

---

[1] *For evaluation purposes, the instructor may check Week 1, Week 2 or Week 3 (though I prefer Week1 = Week2 > Week3)*

I achieved skills in manipulating arrays and numerical calculations with NumPy.
Learned basic plotting concepts with Matplotlib. Also gained experience with data manipulation with Pandas. Implemented gradient descent to optimize a function, building an understanding of iteratively optimizing them.

Week 2:

About: Understanding Neural Networks and Backpropagation

Status: Done

Details:

Assignment:

- Studied feedforward neural networks, understanding how data flows through layers from input to output.
- Explored forward propagation, including weighted sum calculations and activation functions like sigmoid, tanh and relu.
- Learned about loss functions such as cross-entropy loss (for classification) and mse/mean squared error (for regression).
- Understood backpropagation, including gradient descent and weight updates in order to minimize loss.

Resources Covered:

- Feedforward Neural Networks: Learned how neurons process data and pass activations to subsequent layers. (3b1b vids were very helpful)[2]
- Activation Functions: Different activation functions (sigmoid, tanh, ReLU, and leaky ReLU) were compared, and their impact on training was also examined.
- Optimization Problems: Studied gradient descent, loss functions, and their role in nn training.
- Backpropagation: Explored how gradients are calculated and propagated backwards to update weights.

Key Learnings:

- Understood how information flows through a neural network via forward propagation.
- Learned the role of activation functions in introducing non-linearity.
- Gained intuition about gradient descent and how backpropagation updates model weights.
- Practiced building neural networks from scratch, reinforcing theoretical concepts through hands-on implementation

[2] They were definitely helpful throughout all weeks, actually.

Via the assignments:

- Built a perceptron for an AND gate using a single neuron with no hidden layers.
- Attempted to implement an XOR gate using a perceptron (which fails due to its linear separability).
- Successfully implemented XOR with a hidden layer, overcoming the limitation of a single-layer perceptron.
- Developed a full adder using perceptrons and extended it into a ripple carry adder.
- Implemented a basic neural network from scratch using numpy and used functions for initialization, forward propagation, cost calculation and backpropagation in a various classes.

Week 3:

About: Implementing Neural Networks with TensorFlow

Status: Done

Details:

Assignment:

- Learned how to implement a neural network using TensorFlow
- Explored both built-in layers and custom layers for designing deep learning models
- Developed an MNIST handwritten digit classifier using TensorFlow's predefined layers.
- Worked on the Boston Housing Prices prediction task, applying regression techniques with TensorFlow.

Extra exploration:

- Experimented with other MNIST-based datasets (Fashion_mnist) and analyzing model performance across variations.

Resources Covered:

- Basic tensorflow usage: Followed a step-by-step guide on how to use TensorFlow for training neural networks.
- Deep learning with TF: Understood how layers, loss functions, optimizers, and training loops work.
- Compared using TensorFlow's predefined layers vs manually defining custom layers.

Key Learnings:

- Gained hands-on experience in TensorFlow model development.
- Learned the difference between using inbuilt vs. custom layers and their impact on flexibility.
- Understood how to preprocess and train models on MNIST and regression datasets.

Week 4:

About: Understanding Transformers for Text Generation

Status: Partially done

Details:

Explored Transformer Architecture: Learned how transformers process and generate text, studied self-attention mechanisms, and how they assign meaning to words in a sequence. I watched the 3B1B video series on transformers for a visual understanding. I went through the GFG guide on implementing a transformer using TensorFlow (I noted some bugs in the code).

However, I did not complete the assignment.

I did, however, explore the theory and implementation details but did not train a Shakespearean text generator.

Key Learnings:

Understood the core components of transformers, including self-attention, positional encoding, and multi-head attention. I also gained insights into how transformers generate text and their advantages over older sequence models like RNNs and LSTMs.

Also, I identified common challenges in implementing transformers from scratch.