**Transformers Log**

Transformers are deep learning architectures introduced in the paper Attention Is All You Need by Vaswani et al. They rely on self-attention mechanisms to process input sequences in parallel, making them highly effective for NLP tasks. Two key types of Transformer-based models include:

* **Encoders (BERT)**: Used for tasks like text classification, named entity recognition, and masked language modelling.
* **Decoders (GPT-2)**: Primarily used for text generation tasks, leveraging autoregressive modelling.

**Dataset and Preprocessing**

The models work with pre-trained weights and require minimal preprocessing:

* **Tokenization**: Text is tokenized using respective tokenizer models.
* **Padding & Masking**: Necessary for BERT’s masked language modelling task.
* **Batching**: Input text is converted into tensor format for model inference.

**Model Implementation and Architecture**

1. **BERT: Encoder Model (Masked Language Modelling)**

* Pretrained model: bert-base-uncased
* Task: Predicting missing words in a sentence
* Tokenizer: Converts input text into tokens
* Logits extraction: Identifies the most probable word for the masked token
* Output: Most probable word replacing [MASK]

**Performance Observation:**

* The model successfully predicts masked words based on the surrounding context.
* Minor initialization warnings due to unused weights in different tasks.

1. **GPT-2: Decoder Model (Causal Language Modelling)**

* Pretrained model: gpt2
* Task: Autoregressive text generation
* Tokenizer: Converts input prompt into tensor format
* Generation: Predicts next words iteratively
* Output: Expanded text with contextual coherence

**Performance Observation:**

* The model generates coherent, contextually relevant text.
* Warning messages about padding token settings appeared but do not affect output quality.

**Findings and Work Log Summary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase** | **Work Done** | **Duration** | **Difficulty Level (1-10)** |
| Dataset Preparation | Tokenization, masking for BERT | 10 mins | 3 |
| Model Loading | Load pre-trained BERT & GPT-2 models | 5 mins | 2 |
| BERT Implementation | Encode input, predict masked word | 20 mins | 5 |
| GPT-2 Implementation | Tokenize prompt, generate text | 20 mins | 5 |
| Model Evaluation | Analyse results and warnings | 15 mins | 4 |

**Conclusion & Future Enhancements:**

* **Performance Improvement**: Fine-tuning on domain-specific datasets can enhance accuracy.
* **Hyperparameter Tuning**: Adjusting parameters like beam search width, temperature, and top-k sampling for better text generation.
* **Data Augmentation**: Expanding the training corpus with diverse examples can improve robustness.

This log provides a structured approach to implementing Transformer models and forms the basis for further improvements in NLP tasks.