1. Self-attention is a mechanism in transformer models that allows each element in a sequence to focus on other parts of the same sequence, helping capture contextual information. Its main purposes are:
    a. **Capturing Context**: It allows tokens to consider other tokens in a sequence, capturing both nearby and distant dependencies.
    b. **Parallelization**: Unlike RNNs, which operate step-by-step, self-attention processes all elements simultaneously, enabling faster computations.
    c. **Handling Variable-length Sequences**: It works flexibly with sequences of different lengths.

    In essence, self-attention allows a token to gather information from all other tokens in a sequence, making it powerful for understanding context in language tasks.

2. Positional encodings are essential in transformers to address the challenge of order within a sequence, a factor not captured by word embeddings alone. Word embeddings give words meaning but lack positional context. To bridge this gap, positional encodings are generated using sine and cosine functions, yielding unique values for each position. These encodings are then added to the word embeddings, allowing the model to differentiate based on both word identity and position. This combined representation, rich in positional information, ensures that the transformer can accurately perceive and utilize the sequential nature of the input data. This understanding is crucial for tasks such as translation, summarization, and language modeling, where the order of words significantly impacts the meaning and output.

Analysis:

batch_size = 30

num_heads = 8

num_layers = 2

max_sequence_length = 350

```
Avg Train Loss: 2.4504
Avg Validation Loss: 2.3403
Avg Test Loss: 2.3118


Train BLEU Score: 0.0395
Test BLEU Score: 0.0335
Validation BLEU Score: 0.0370
```