Understanding Multi-Head Attention in Transformers

Student Name: Muhammad Sarmad Malik

Student Id: 24089891

GitHub Repository: https://github.com/msarmad2000/multi-head-attention-transformer-tutorial-24089891

Introduction:

Transformers have rapidly become the dominant architecture across natural language processing, computer vision, speech modelling and many other domains. Their success is largely due to the self-attention mechanism, which enables models to evaluate relationships between all positions of a sequence in parallel, rather than processing sequentially as in recurrent neural networks. An especially powerful component of this mechanism is multi-head attention, where several attention "heads" operate simultaneously on different learned projections of the same input. Each head can specialise in a different relational pattern, giving the Transformer an expressive advantage that single-head attention lacks.

This tutorial demonstrates how multi-head attention behaves in practice by training a series of small Transformer encoder models on a controlled synthetic task. By systematically varying the number of attention heads, removing positional encoding, and altering sequence length, we isolate and observe how each architectural component influences learning.

Using a deliberately designed dataset — the shifted-copy task — we illustrate why positional encodings are necessary, how multiple heads specialise, and why longer sequences increase task difficulty. The aim is to help readers build strong intuition for how attention works, making it clearer why Transformers perform so effectively on real-world data.

Dataset Description:

We use a medium-sized synthetic dataset designed specifically to reveal the behaviour of multi-head attention without the noise or unpredictability of natural language data.

Each sample consists of:

- A sequence of 20 tokens
- Token values between 0 and 19
- 10,000 training samples, plus validation and test sets

Rather than the trivial “copy” task, we use a shifted-copy prediction task. For an input sequence:

$$x = [x_0, x_1, x_2, \ldots, x_{n-1}]$$

the target output is:

$$y = [x_{n-1}, x_0, x_1, \ldots, x_{n-2}]$$

This circular shift forces the model to learn the relationship between each position and the previous one, making positional information essential.

This task is ideal because:

It cannot be solved without positional encoding

It produces highly interpretable attention maps

It scales smoothly when increasing sequence length

It makes multi-head specialisation visible

Model Description:

To keep explanations and visualisations manageable, we use a small Transformer encoder with:

- Token embedding layer
- Optional **sinusoidal positional encoding**
- A single **multi-head self-attention block**
- A feed-forward network
- Layer normalisation and residual connections
- A linear output layer that predicts each token independently

Multi-Head Attention:

Multi-head attention splits the embedding into several subspaces, each processed by a separate attention head. Each head computes:

- Queries
- Keys
- Values
- Attention weights
- Weighted output

Heads are then concatenated and linearly projected.

This allows the model to learn:

- Local vs. global dependencies
- Different relational patterns
- Redundant or complementary behaviours

Positional Encoding

Since attention is permutation-invariant, we supply ordering information via sinusoidal positional encodings, following Vaswani et al. (2017). These encodings allow the model to infer which positions are adjacent — essential for the shifted-copy task.

We later remove positional encoding entirely to explore its impact.

Experiments

We run three controlled experiments:

- Varying number of attention heads (1, 2, 4, 8)
- Ablation: removing positional encoding
- Changing sequence length (10, 20, 40)

All models use identical settings except for the variable under test.

Experiment A: Effect of Number of Heads

We train four models that differ only in head count.

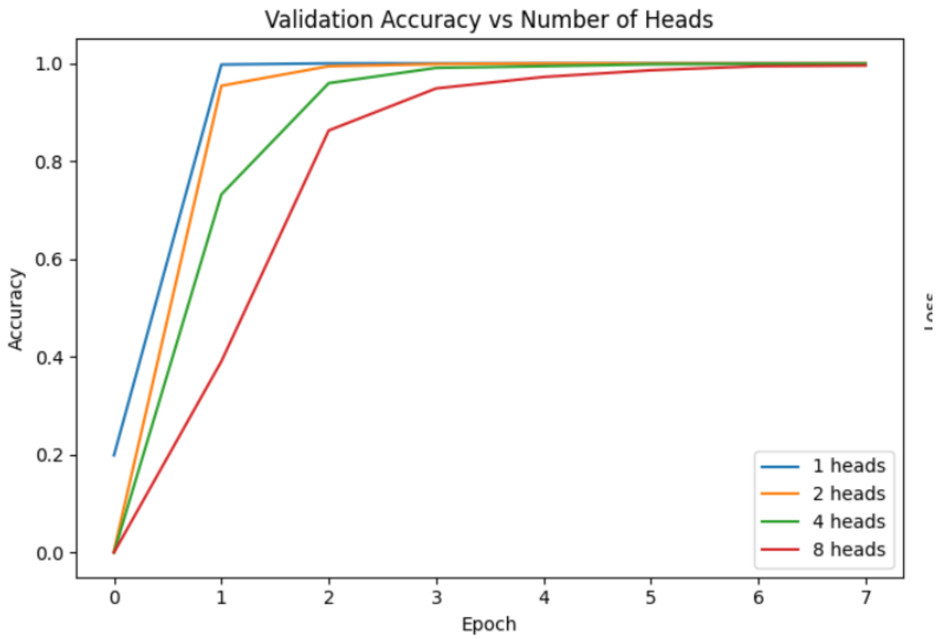

Figure 1 — Validation accuracy for models with 1, 2, 4, and 8 heads.

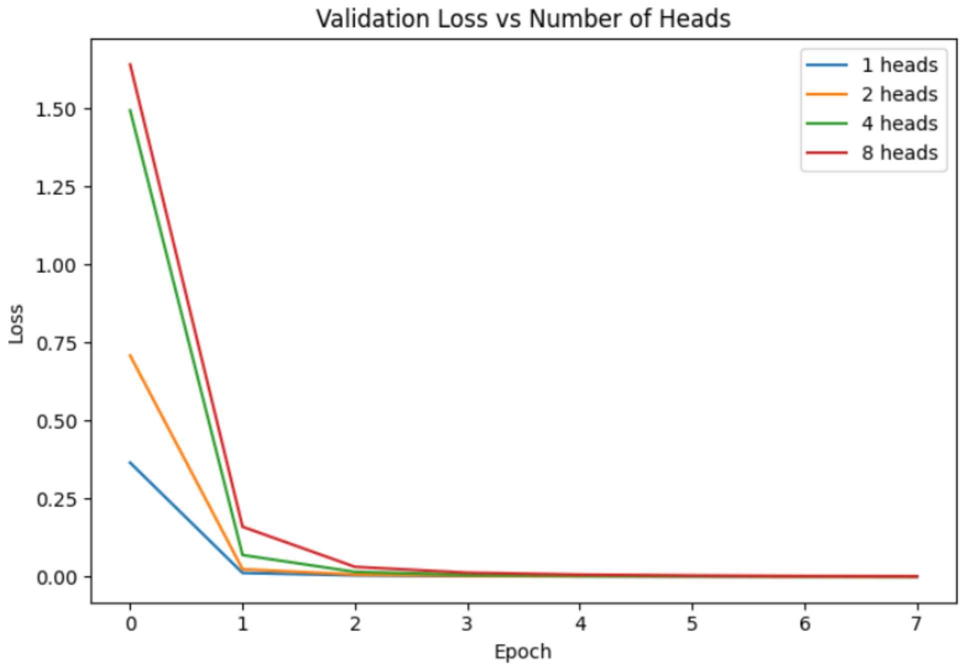

Figure 2 — Validation loss for models with 1, 2, 4, and 8 heads.

Results:

- 1 head learns the correct pattern but slightly slower
- 2 and 4 heads converge fastest and most stably
- 8 heads show diminishing returns and slight redundancy

Interpretation:

Adding more heads increases representational power but also:

- increases parameter count,
- leads to overlapping or redundant heads,
- offers minimal benefit for simple tasks.

This aligns with research showing many attention heads can be pruned with little performance loss (Michel et al., 2019).

Experiment B: Positional Encoding Ablation

We compare two models:

- 4-head model WITH positional encoding
- 4-head model WITHOUT positional encoding

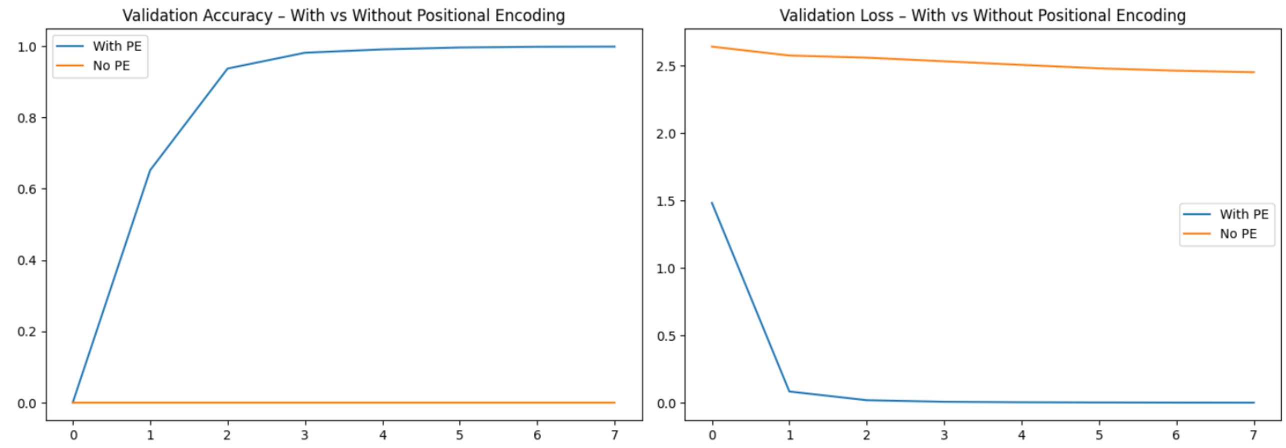

Figure 3 — Validation accuracy with vs without positional encoding

Results

- The model with positional encoding learns quickly and achieves high accuracy.
- The model without positional encoding fails to learn the shifted dependency.
- Validation accuracy remains low
- Loss decreases slowly
- Attention maps are disorganised

Interpretation

Without positional encoding, the model cannot determine:

- which token precedes which,
- where attention should be directed,
- how the output index relates to the input index.

This confirms a core insight of Transformer architecture: attention cannot learn order unless order is explicitly encoded.

Experiment C: Effect of Sequence Length

We repeat training for sequence lengths:

- 10
- 20 (baseline)
- 40

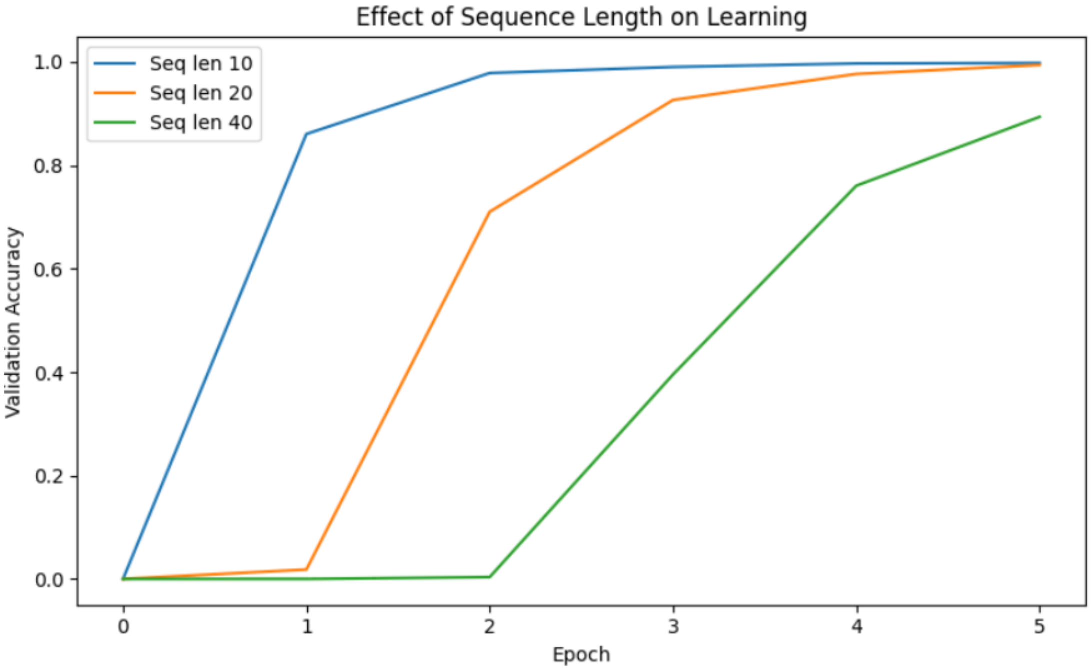

Figure 4 — Validation accuracy for sequence lengths 10, 20, and 40.

Results

- Length 10 learns fastest
- Length 20 performs well (baseline)
- Length 40 converges slowly and may plateau lower

Interpretation

Longer sequences:

- require attention over more positions
- increase the quadratic cost of attention
- amplify long-range dependencies
- cause heads to spread attention more diffusely

This mirrors real-world challenges in long-sequence processing and motivates alternatives like sparse attention and linear-time Transformers.

Attention Visualisation

A unique advantage of the synthetic task is the interpretability of attention maps.

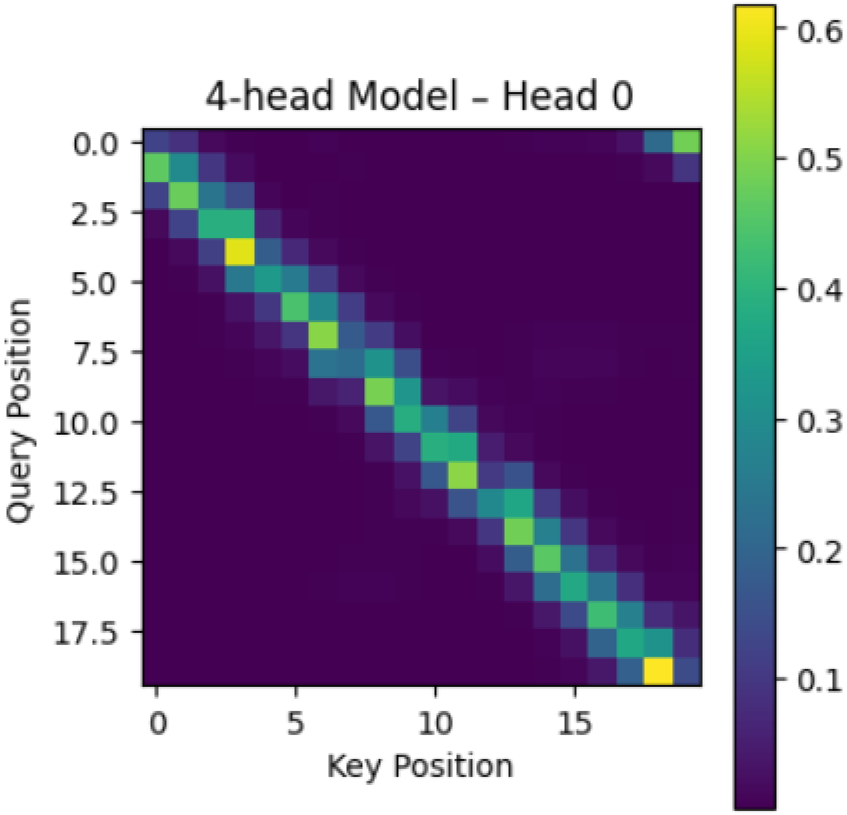

Figure 5. Attention map from head 0 of a 4-head model.

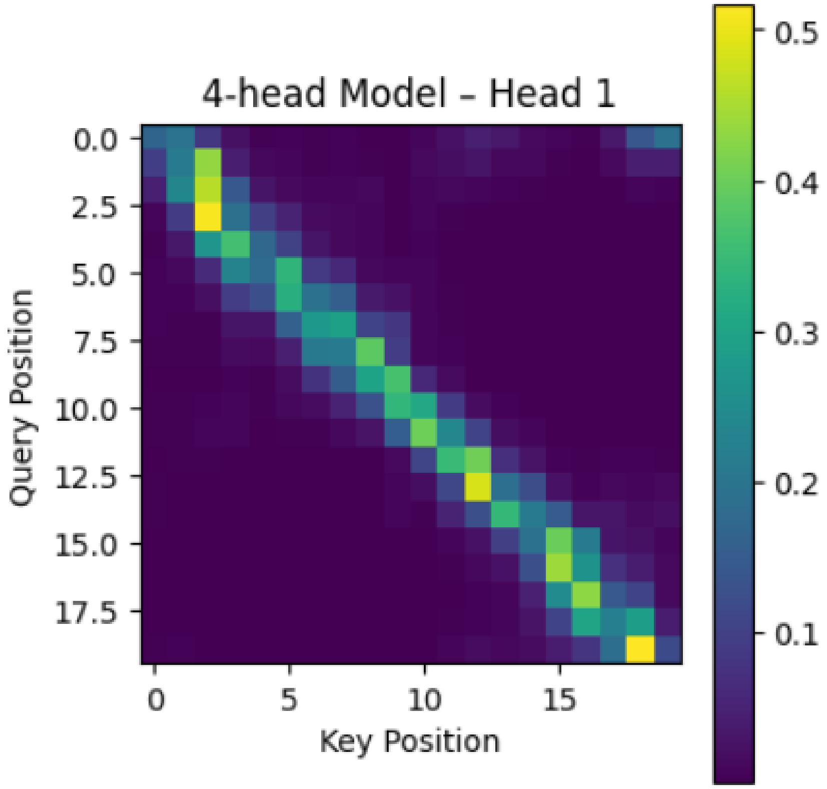

Figure 6. Attention map from head 1 of a 4-head model

Findings

- One head consistently attends to the previous position, forming a strong band just below the diagonal — exactly matching the shifted-copy rule.
- Other heads attend to different contextual patterns, such as smoothing or identity-like behaviour.
- Without positional encoding, attention maps become noisy or uniform.

Interpretation

Attention visualisation confirms:

- Head specialisation
- Correct learning of relational structure
- Failure of order reasoning without positional encoding

Such interpretability is a major reason why attention mechanisms are popular in research environments.

Results

Across all experiments, we observe:

- Multi-head attention improves learning, but extra heads yield diminishing gains.
- Positional encoding is essential for any order-based task.
- Longer sequences are harder, causing slower convergence and noisier attention.
- Attention maps clearly reveal how heads specialise, making behaviour more interpretable than in RNNs.

The controlled environment of the synthetic shifted-copy task exposes the core principles behind Transformer performance without the distractions of natural language complexity.

Conclusion

This tutorial provided a focused exploration of multi-head attention using a carefully designed synthetic dataset. By varying head count, removing positional encodings, and changing sequence length, we isolated the specific factors that influence Transformer learning behaviour. Both quantitative metrics and qualitative attention visualisations offer clear evidence of how attention heads specialise and why positional structure is indispensable.

These experiments reinforce foundational knowledge relevant to any student or practitioner working with Transformer-based models. Understanding these principles is not only useful for academic insight but also essential for debugging, optimising, and designing Transformer architectures in real-world machine-learning applications.

References:

- Vaswani, A., et al. (2017). *Attention is All You Need.*
- Clark, K., et al. (2019). *What Does BERT Look at? An Analysis of Attention.*
- Michel, P., Levy, O., & Neubig, G. (2019). *Are Sixteen Heads Really Better Than One?*
- Dosovitskiy, A., et al. (2021). *An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale.*
- Bahdanau, D., et al. (2015). *Neural Machine Translation by Jointly Learning to Align and Translate.*