# BasedAI: A Novel Framework for Privacy-Preserved, Scalable Computation with Large Language Models

BasedAI Research Team

October 10th, 2023

## 9. Appendix

BasedAI introduces Cerberus Squeezing as a method to enhance the practicality and performance of Fully Homomorphic Encryption (FHE) applications. This approach optimizes the efficiency and speed of neural network operations on encrypted data by focusing on computational resource allocation within multi-head attention mechanisms. Furthermore, BasedAI's research into "Encrypted Data Synthesis" and "Quantization-Aware Training" aims to address challenges such as limited data accessibility and the need for models to adapt to quantization effects, thereby expanding secure training resources and improving protocol performance.

### 9.1 Cerberus Squeezing

"Cerberus Squeezing" is a technique aimed at optimizing multi-head attention mechanisms in neural network models, particularly for processing encrypted data under FHE. This involves selectively focusing computational resources on the most impactful attention heads to enhance efficiency and performance.

**Multi-Head Attention Mechanism** The multi-head attention mechanism in a neural network model can be represented as a function $A$ that operates on an input sequence $X$ to produce an output sequence $Y$, where each head $h$ focuses on different parts of $X$:

$$Y = A(X) = \text{Concat}(head_1, head_2, \ldots, head_k)W_O$$

, with each head computed as:

$$head_i = \text{Attention}(XW_i^Q, XW_i^K, XW_i^V)$$

, where $W_i^Q$, $W_i^K$, and $W_i^V$ are weight matrices for the i-th head's query, key, and value, respectively, and $W_O$ is the output weight matrix.

**Selective Optimization** Cerberus Squeezing identifies and prioritizes heads that contribute most significantly to the model's performance. If $S(h)$ represents the significance score of head $h$, computational resources are focused on heads with $S(h) > \theta$, where $\theta$ is a threshold determining head significance.

**Resource Allocation** This optimization can be formalized as a constrained optimization problem, aiming to maximize model performance under computational resource constraints:

$$\max_{W_i^Q, W_i^K, W_i^V} \text{Performance(Model)},$$

subject to

$$C(W_i^Q, W_i^K, W_i^V) < C_{\max}$$

, where $C(\cdot)$ measures the computational cost, and $C_{\max}$ represents the maximum allowable computational budget.

## 9.2 Further Research

**9.2.1 Encrypted Data Synthesis** "Encrypted Data Synthesis" refers to the process of generating synthetic data points from existing encrypted datasets without decrypting them. If applied to BasedAI, this technique would leverage the inherent patterns and structures within a given set of data, preserved under encryption, to fabricate new, realistic data points that can enhance BasedAI model training.

**Encrypted Data Representation** Consider an encrypted data point represented as $E(x_i)$, where $x_i$ is the original data point, and $E(\cdot)$ denotes the encryption function. The dataset of encrypted points is denoted as $D_{\text{enc}} = \{E(x_1), E(x_2), \ldots, E(x_n)\}$.

**Pattern Extraction** Assuming a function $F_{\text{pattern}}$ that operates on encrypted data to identify patterns without decryption:

$$F_{\text{pattern}}(D_{\text{enc}}) \rightarrow P_{\text{enc}}$$

, where $P_{\text{enc}}$ represents the pattern information extracted from $D_{\text{enc}}$, still in encrypted form.

**Data Fabrication** The fabrication of new data points, $E(x_{\text{new}})$, uses the pattern information $P_{\text{enc}}$ to guide the generation process through a generative model $G$:

$$E(x_{\text{new}}) = G(P_{\text{enc}})$$

, where $G$ is designed to work within the FHE scheme, ensuring that the output is a realistically fabricated data point, still encrypted.

**9.2.2 Quantization-Aware Training** Applying further research into Quantization-Aware Training (QAT) would allow BasedAI to optimize its neural network models for efficient deployment in environments where computational resources are limited, such as devices performing encrypted data computations. This technique adjusts models to operate effectively with lower precision arithmetic, which is critical for maintaining performance under the constraints of Fully Homomorphic Encryption (FHE).

**Model Quantization** Quantization reduces the precision of the model's parameters and activations, represented as moving from floating-point representations $F(x_i)$ to lower bit-width representations $Q(x_i)$. The quantized model's parameters and activations can be denoted as $Q_{\text{param}}$ and $Q_{\text{act}}$, respectively.

**Quantization Function** The quantization operation can be mathematically represented as:

$$Q(x) = \text{round}\left(\frac{x - \mu}{\sigma}\right) \cdot q_{\text{scale}} + q_{\text{zero}}$$

where $x$ is the input, $\mu$ and $\sigma$ are the mean and standard deviation of the input distribution, $q_{\text{scale}}$ is the scaling factor for quantization, and $q_{\text{zero}}$ is the zero-point offset.

**Quantization-Aware Training Process** During QAT, the model is trained or fine-tuned with simulated quantization effects. This involves integrating the quantization function into the forward pass, allowing the model to adapt to the quantized representation before actual deployment:

$$F_{\text{QAT}}(x) = Q^{-1}(Q(x))$$

where $F_{\text{QAT}}$ represents the model's forward pass incorporating quantization and $Q^{-1}$ denotes the dequantization operation to simulate the effect of quantization on the model's performance.

**Objective** The objective of QAT is to minimize the discrepancy between the performance of the quantized model and its full-precision counterpart, optimizing for a loss function $L$ that incorporates the quantization effects:

$$\min_{Q_{\text{param}}, Q_{\text{act}}} L(F_{\text{QAT}}(x), y)$$

where $y$ is the true output. By training the model to anticipate and adjust to quantization, QAT would ensure that BasedAI's neural networks remain efficient and accurate when deployed in resource-constrained FHE environments.