# Module H: Core Model Architectures (RWKV and Transformer Modules, Mixture-of-Experts Style)

Part of the Eidos Unified Framework for Persistent, Dynamic, and Adaptive Multimodal Intelligence

—

## Contents

# 1 Abstract

This module rigorously defines the *Core Model Architectures* component of the Eidos framework. It encompasses two complementary deep learning architectures: the Transformer and the RWKV, integrated in a mixture-of-experts (MoE) style. The Transformer sub-module provides a powerful self-attention mechanism for capturing long-range dependencies, while the RWKV sub-module offers a recurrent, linear-time alternative. A higher-level expert coordinator aggregates outputs from multiple expert modules into a unified model output. This design balances expressive capacity and computational efficiency, supports dynamic expert addition and removal, and enables robust performance across diverse tasks. We present formal definitions, algorithmic descriptions, theoretical guarantees, and integration strategies with maximum academic rigor.

# 2 Introduction and Motivation

The core processing engine of the Eidos framework is designed to efficiently capture complex dependencies in multimodal data. In this module, we integrate two leading deep architectures:

(a) **Transformer Sub-Module:** Leveraging multi-head self-attention, the Transformer captures long-range dependencies and intricate interactions among token representations.

(b) **RWKV Sub-Module:** A recurrent alternative that computes a context vector via weighted accumulation with learnable decay and gating, offering linear-time complexity.

To maximize both capacity and efficiency, these sub-modules are organized in a *mixture-of-experts* (MoE) framework. A dedicated expert coordinator, denoted by $\Gamma$, aggregates the outputs of a set of expert modules, each specialized to different aspects of the input. This architecture not only enables dynamic expert specialization and scaling but also allows for seamless addition or removal of experts without disrupting overall performance.

# 3 Preliminaries and Notation

We adopt the following notation and assumptions:

- Let $X = (x_1, x_2, \ldots, x_n)$ denote an input sequence, where each $x_i$ is processed into a final token representation $E_{\mathrm{F}}(x_i, \xi) \in \mathbb{R}^{d_F}$ (obtained from Module E).
- The overall deep model is a function:

$$f_\theta : (\mathbb{R}^{d_F})^n \to \mathcal{Y},$$

  parameterized by $\theta \in \Theta$.
- The set of expert sub-modules is denoted by:

$$\mathcal{E} = \{f_{\theta_i}^{(i)} \mid i \in I_{\mathrm{exp}}\},$$

  where each $f_{\theta_i}^{(i)}$ represents an individual expert, which may be either a Transformer module $f_{\theta_{\mathrm{T}}}^{\mathrm{T}}$ or an RWKV module $f_{\theta_{\mathrm{R}}}^{\mathrm{RWKV}}$.
- A higher-level expert coordinator $\Gamma$ aggregates expert outputs to produce a unified prediction:

$$f_\theta^{\mathrm{Unified}} = \Gamma\Big(\{f_{\theta_i}^{(i)}\}_{i \in I_{\mathrm{exp}}}\Big).$$

# 4 Formal Definitions and Mathematical Formulation

### Definition H.1 (Transformer Sub-Module)

Let $Z = (z_1, z_2, \ldots, z_n) \in \mathbb{R}^{n \times d_{\text{model}}}$ be the input representation obtained by adding positional encodings to the token embeddings:

$$z_i = E_{\text{F}}(x_i, \xi) + PE(i).$$

For each Transformer layer, the following operations are performed:

(i) **Linear Projections:**

$$Q = ZW^Q, \quad K = ZW^K, \quad V = ZW^V,$$

with $W^Q, W^K \in \mathbb{R}^{d_{\text{model}} \times d_k}$ and $W^V \in \mathbb{R}^{d_{\text{model}} \times d_v}$.

(ii) **Scaled Dot-Product Attention:**

$$\text{Attention}(Q, K, V) = \text{softmax}\Big(\frac{QK^\top}{\sqrt{d_k}}\Big)V.$$

(iii) **Multi-Head Attention:** With $h$ heads, for head $i$ define:

$$\text{head}_i = \text{Attention}(ZW_i^Q, ZW_i^K, ZW_i^V),$$

and concatenate:

$$\text{MHA}(Z) = \text{Concat}(\text{head}_1, \ldots, \text{head}_h)W^O.$$

(iv) **Residual Connection and Layer Normalization:**

$$Z' = \text{LayerNorm}\Big(Z + \text{MHA}(Z)\Big).$$

(v) **Feed-Forward Network:**

$$\text{FFN}(Z') = \sigma\big(Z'W_1 + b_1\big)W_2 + b_2,$$

with subsequent residual and normalization:

$$Z'' = \text{LayerNorm}\Big(Z' + \text{FFN}(Z')\Big).$$

Thus, the Transformer sub-module $f^{\text{T}}_{\theta_{\text{T}}}$ is defined as a stack of such layers.

### Definition H.2 (RWKV Sub-Module)

For a recurrent alternative, the RWKV module processes a sequence $(x_1, \ldots, x_n)$ as follows:

(i) **Embedding:** For each token, compute $z_t = E_{\text{F}}(x_t, \xi)$.

(ii) **Linear Projections:**

$$k_t = W_k z_t + b_k, \quad v_t = W_v z_t + b_v, \quad r_t = \sigma(W_r z_t + b_r),$$

where $r_t \in (0, 1)^d$ acts as a gating (receptance) vector.

(iii) **Recurrent Accumulation:** Initialize $S_0 = \mathbf{0}$ and $Z_0 = \epsilon \mathbf{1}$ (with small $\epsilon > 0$). Then for $t \geq 1$:

$$S_t = \lambda \odot S_{t-1} + \exp(k_t) \odot v_t, \quad Z_t = \lambda \odot Z_{t-1} + \exp(k_t),$$

where $\lambda \in [0,1]^d$ is a (possibly learnable) decay parameter.

(iv) **Output Computation:** The output at time $t$ is given by:

$$y_t = r_t \odot \left( \frac{S_t}{Z_t} \right).$$

The RWKV sub-module $f_{\theta_{\mathrm{R}}}^{\mathrm{RWKV}}$ is defined by applying these recurrent operations over the entire sequence.

### Definition H.3 (Mixture-of-Experts Coordination)

Let the set of expert modules be:

$$\mathcal{E} = \{ f_{\theta_i}^{(i)} \mid i \in I_{\mathrm{exp}} \},$$

where each expert $f_{\theta_i}^{(i)}$ is instantiated as either a Transformer module (as in Definition H.1) or an RWKV module (as in Definition H.2), possibly specialized for different tasks or data aspects.

Define the expert coordinator as a function:

$$\Gamma : \prod_{i \in I_{\mathrm{exp}}} \mathcal{F}^{(i)} \to \mathcal{F}^{\mathrm{Unified}},$$

which aggregates the outputs $\{y^{(i)}\}$ of the individual experts to produce a unified output:

$$f_\theta^{\mathrm{Unified}}(X) = \Gamma\Big( \{ f_{\theta_i}^{(i)}(X) \}_{i \in I_{\mathrm{exp}}} \Big).$$

A common instantiation of $\Gamma$ is a weighted sum or concatenation followed by a linear projection:

$$f_\theta^{\mathrm{Unified}}(X) = \Big[ \sum_{i \in I_{\mathrm{exp}}} w_i \, f_{\theta_i}^{(i)}(X) \Big] W^C,$$

with weights $w_i$ (possibly learned or dynamically computed) and a coordinator projection $W^C$.

## 5 Algorithmic Description

The following pseudocode summarizes the forward pass of the unified core model architecture:

**Algorithm 1** Unified Core Model Forward Pass (Module H)

**Input:** Token sequence $X = (x_1, \ldots, x_n)$; final token representations $\{E_{\mathrm{F}}(x_i, \xi)\}$
**Output:** Model prediction $\hat{y} \in \mathcal{Y}$
**Begin:**
  Compute input representation $Z$ from $\{E_{\mathrm{F}}(x_i, \xi)\}$
  // **Expert Processing:**
**for** each expert $i \in I_{\text{exp}}$ **do**
  **if** $f^{(i)}$ is a Transformer expert **then**
    $y^{(i)} \leftarrow f_{\theta_i}^{\mathrm{T}}(Z)$
  **else**
    $y^{(i)} \leftarrow f_{\theta_i}^{\mathrm{RWKV}}(Z)$
  **end if**
**end for**
  // **Expert Coordination:**
  Compute unified output:

$$y_{\text{unified}} \leftarrow \Gamma\Big(\{y^{(i)}\}_{i \in I_{\text{exp}}}\Big)$$

  // **Final Prediction:**
  $\hat{y} \leftarrow \operatorname{softmax}\big(y_{\text{unified}} W^P + b^P\big)$
**Return:** $\hat{y}$

# 6 Theoretical Analysis and Guarantees

### Theorem H.1 (Expressivity of the Mixture-of-Experts Model)

**Statement:** Assume that each expert $f_{\theta_i}^{(i)}$ is a universal approximator over its domain and that the coordinator $\Gamma$ is a non-degenerate aggregation operator. Then, the unified model

$$f_\theta^{\text{Unified}}(X) = \Gamma\Big(\{f_{\theta_i}^{(i)}(X)\}_{i \in I_{\text{exp}}}\Big)$$

is a universal approximator for functions from $(\mathbb{R}^{d_F})^n$ to $\mathcal{Y}$.
**Proof Sketch:** Since each expert can approximate any function to arbitrary accuracy and the coordinator aggregates these approximations in a weighted (or concatenated) manner, standard universal approximation theorems for neural networks imply that the composite function can approximate any target function over a compact domain. □

### Proposition H.2 (Computational Efficiency)

The use of both Transformer and RWKV experts enables balancing of computational complexity. Transformers have a complexity of $O(n^2)$ per layer due to self-attention, while RWKV modules run in $O(n)$ time. The mixture-of-experts framework can dynamically allocate computational resources to experts based on input characteristics, thus optimizing overall efficiency.

# 7 Integration with the Overall Eidos Framework

Module H is the central processing engine of the Eidos framework. It:

- Accepts the final token representations $\{E_{\mathrm{F}}(x_i, \xi)\}$ produced by Module E.

- Processes these representations using multiple expert sub-modules (Transformers and RWKV), each of which may specialize in different aspects of the input.
- Aggregates expert outputs using the coordinator $\Gamma$ to produce a unified prediction.
- Provides an interface for subsequent modules (e.g., memory, training, decoding) to operate on the model's output.

# 8 Implementation Considerations

- **Expert Specialization:** The set of experts $\{f_{\theta_i}^{(i)}\}$ may be pre-assigned or dynamically adjusted based on input characteristics or training objectives.
- **Coordinator Design:** The aggregation function $\Gamma$ can be implemented as a weighted sum, a gating mechanism, or even a small neural network that learns how to fuse expert outputs.
- **Parallelization:** Experts can be computed in parallel, leveraging modern hardware accelerators (GPUs/TPUs) to reduce latency.
- **Dynamic Expert Management:** Mechanisms for adding, removing, or re-weighting experts should be incorporated to allow for scalability and adaptivity.
- **Training Strategy:** Joint or staged training of experts and the coordinator may be used, with careful tuning of learning rates and regularization to avoid overfitting any single expert.

# 9 Conclusion

In this module, we have defined a comprehensive core model architecture that integrates Transformer and RWKV sub-modules in a mixture-of-experts style. Key contributions include:

- A formal definition of both Transformer and RWKV sub-modules with their internal operations.
- The formulation of a mixture-of-experts framework wherein multiple expert modules operate in parallel.
- The introduction of an expert coordinator $\Gamma$ that aggregates individual expert outputs into a unified prediction.
- Theoretical guarantees regarding the expressivity and computational efficiency of the unified model.
- Detailed algorithmic pseudocode outlining the forward pass and integration of expert modules.

This module is the primary processing engine of the Eidos framework, providing both high-level semantic abstraction and computational efficiency while enabling flexible, dynamic adaptation through expert specialization.

## 10 Module Summary

**Completed:**

- Module A: Input Processing.
- Module B: Universal Communication & Data Handling Interface and Coordination.
- Module C: Universal Streaming/Handling/Loading/Indexing Module.
- Module D: Multidimensional Vocabulary and Tokenization System.
- Module E: Contextual NLU/NLP Embedding and Multidimensional Tokenization.
- Module F: Deep Knowledge Graphs System (Base and Personal).
- Module G: Infinite RoPE Context Scaling and Dynamic Vocabulary Updating.
- Module H: Core Model Architectures (RWKV and Transformer Modules, Mixture-of-Experts Style).

**Remaining Modules:**

- Module I: Titans Memory Architecture (Multi-Layer Memory Module).
- Module J: Recursive Adaptive Dynamic Idempotent Feedback and State-Based Runtime Learning and Inference.
- Module K: Universal Training System.
- Module L: Final Decoding and Multimodal Output.