Modeling Triplet Interactions Beyond Pairwise Attention

Mafizur Rahman and Lijun Qian
CREDIT Center, Department of ECE
Prairie View A&M University
Prairie View, Texas, USA
{mrahman13, liqian}@pvamu.edu

Abstract

Standard transformers use pairwise (2D) self-attention, which models only token-to-token interactions. Sparse attention reduces computational cost while keeping the same pairwise structure and cannot capture richer multi-token interactions. Therefore, we propose Higher-Order Block Attention (HOBA), which models explicit triplet interactions using structured 3D attention with block-diagonal unrolling. This design remains computationally efficient while extending attention beyond pairwise relations. Additionally, we demonstrate that cross-block interaction improves HOBA performance by supporting long-range token dependencies beyond triplet modeling. Experiments on ten NLP benchmarks across five tasks show that HOBA consistently outperforms dense and sparse 2D baselines.

1 Introduction

Transformer architectures have become the dominant paradigm for natural language processing (NLP), largely due to the success of self-attention mechanisms in modeling contextual dependencies across tokens (Vaswani et al., 2017; Liu et al., 2023). Standard transformers rely on pairwise (2D) self-attention, which computes interactions between token pairs (Keles et al., 2023). While this formulation is highly effective, it inherently models only pairwise relationships and may struggle to explicitly represent dependencies that arise from interactions among multiple tokens simultaneously (Sekhon et al., 2023). In natural language, many semantic phenomena are inherently compositional and depend on interactions among groups of tokens rather than isolated pairs. For example, in the phrase "*strong quarterly earnings*," the meaning emerges from the joint interaction among *strong*, *quarterly*, and *earnings*, which cannot be fully captured through pairwise relations alone.

A large body of work improves attention efficiency through sparsity or structural approximations, such as Sparse Transformer, Longformer, BigBird, and Linformer, which restrict token interactions to reduce computational cost (Beltagy et al., 2020; Zaheer et al., 2020). Although these approaches significantly improve scalability, they retain the same pairwise attention formulation and therefore do not explicitly model higher-order (HO) token relationships. At the same time, prior attempts to incorporate HO attention remain limited due to their reliance on global modeling strategies such as spectral kernels, full-sequence graph propagation, or dense tensor contractions (Zhou et al., 2022; Sekhon et al., 2023; Choromanski et al., 2020; Hong et al., 2022). These approaches typically operate over the entire sequence and incur cubic or near-cubic computational complexity, making them difficult to integrate into standard transformer pipelines. This limitation raises an important question: *can we design an attention mechanism that captures HO token interactions while remaining computationally efficient and compatible with existing transformer models?* Addressing this question requires balancing two competing objectives: maintaining the expressive power of HO interactions while preserving the efficiency that makes modern Transformer models practical.

In this work, we introduce **H**igher-**O**rder **B**lock **A**ttention (**HOBA**), a scalable attention mechanism that models token triplet interactions through structured HO attention. Instead of computing a full cubic interaction tensor, HOBA decomposes attention into localized block-diagonal regions that capture triplet-level relationships efficiently. This block-based design significantly reduces the computational cost of HO attention while preserving the ability to model richer contextual dependencies. Furthermore, we incorporate a cross-block interaction mechanism that allows HOBA to capture moderate to long-range dependencies across blocks

Findings of the Association for Computational Linguistics: IJCNLP-AACL 2026, pages 2010–2026
November 6-10, 2026 ©2026 Association for Computational Linguistics

without sacrificing computational efficiency.

We evaluate HOBA across ten benchmark datasets spanning five NLP task categories: sentiment analysis, news classification, question classification, natural language inference, and question answering. In addition to standard language understanding tasks, we include reasoning-oriented benchmarks such as HotpotQA and StrategyQA that require modeling complex multi-token dependencies. Experimental results show that HOBA consistently improves performance over standard pairwise attention and recent sparse attention baselines while maintaining efficient computation. **The contributions of this work are as follows:**

1. We design HOBA, *the first scalable triplet-based attention mechanism that models HO interactions utilizing localized block-diagonal structures*. It dramatically reduces the cubic cost of 3D attention while maintaining attention granularity by enabling deployment in real-world tasks.

2. We incorporate a cross-block mechanism into HOBA that forms token triplets across disjoint blocks. *It allows HOBA to capture global dependencies without sacrificing efficiency*. We show that utilizing cross-block interaction can improve accuracy by up to 13% and reduce loss by up to 0.22, which proves its essential role in capturing moderate to long-range dependencies.

3. We support our findings with theoretical generalization analysis and qualitative visualizations illustrating how HOBA captures HO token interactions.

Appendix A.8 provides further discussion and a theoretical proof of the advantages of triplet modeling over pairwise interactions.

2 Related Works

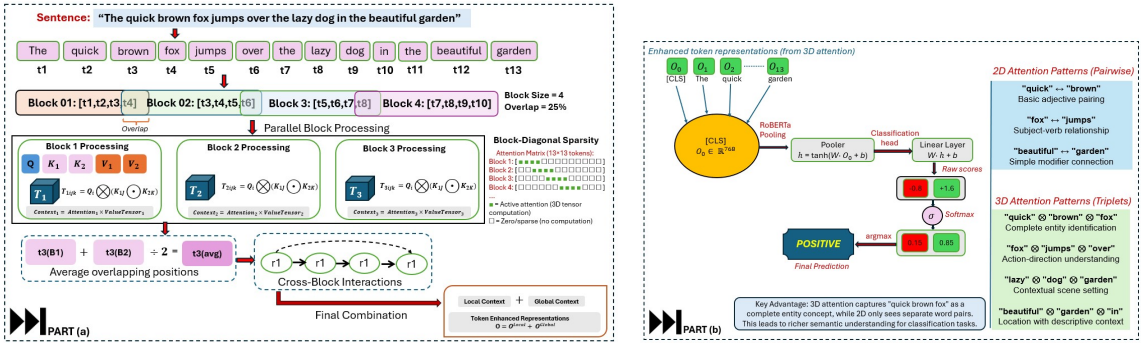
Higher-Order and Sparse Attention. The quadratic cost of standard self-attention has motivated extensive work on structured approximations (Keles et al., 2023; Likhosherstov et al., 2023). Sparse attention models such as Longformer and BigBird introduce local windows and global tokens to lessen computational complexity while maintaining contextual awareness (Beltagy et al., 2020; Zaheer et al., 2020). Similarly, Linformer offers low-rank projections to facilitate attention computation (Wang et al., 2020), while Performer and Routing Transformer replace traditional dot products with kernel-based and routing-based approximations for improved scalability (Choromanski et al., 2020; Roy et al., 2021). Jump Self-Attention (JAT) extends this direction by capturing HO dependencies via spectral convolution, enhancing performance in tasks like natural language understanding (Zhou et al., 2022). Likewise, WIGRAPH presents a trainable layer that learns global word-level interactions, improving both interpretability and predictive accuracy in NLP models (Sekhon et al., 2023). Another work proposes Sparse Self-Attention Fine-tuning (SSAF) that can replace softmax with a controllable sparse transformation to improve the performance and interpretability of BERT fine-tuning (Cui et al., 2019). Beyond sparsity, recent HO transformers reduce computational overhead using Kronecker product factorization and kernelized attention approximations (Omranpour et al., 2024). Similarly, Graph-Induced Attention goes further by embedding syntactic or semantic graph structures directly into the attention mechanism by enabling stronger inductive bias and structural awareness (Hong et al., 2022).

The above methods primarily focus on approximating or sparsifying pairwise attention or modeling HO interactions through spectral convolution, graph propagation, or dense tensor contractions. In contrast, HOBA explicitly models HO token interactions through structured triplet attention, addressing a missing component in prior sparse-attention studies. Importantly, the novelty of HOBA does not lie in block sparsity alone, but in extending the block-diagonal principle from conventional pairwise attention to structured triplet attention. While existing sparse-attention methods sparsify a 2D token–token attention matrix, HOBA decomposes an otherwise cubic 3D interaction tensor into localized block-wise triplet regions and complements them with cross-block interactions. This design preserves explicit HO dependencies while avoiding full-sequence cubic computation, making triplet-based attention computationally practical within standard Transformer architectures. An extended conceptual comparison between prior HO attention methods and HOBA is provided in **Appendix A.7**.

3 Block-Diagonal Unrolling Framework

3.1 Problem Formulation

Let $X = [x_1, x_2, \ldots, x_L] \in \mathbb{R}^{L \times d}$ be an input sequence of L tokens, where each token embedding $x_i \in \mathbb{R}^d$. The goal is to learn a transformation $f_\theta(X)$ that produces a contextual representation $H \in \mathbb{R}^{L \times d'}$ for a variety of natural lan-



(a) 3D Block-Diagonal Attention Processing

(b) Classification Pipeline and Pattern Comparison

Figure 1: Complete HOBA unrolling Framework. (a) Once the input sentence is tokenized, we split the tokens into several blocks and process the 3D tensor parallelly within a block-diagonal sparsity pattern that eliminates redundant full-matrix computations. The block-diagonal attention matrix shows how our method gains computational efficiency by restricting attention to local block regions while preserving sparse zero-computation areas outside the diagonal blocks. Later, we do overlap averaging and compute cross-block interactions. Finally, we get the enhanced representation of the tokens with lowered complexity from $O(n^2)$ standard attention to $O(nb^2)$ block-diagonal attention. (b) Enhanced token representations flow through RoBERTa pooling to the final prediction. Additionally, we show (right side) how 3D attention patterns capture richer semantic relationships than standard 2D attention, leading to improved classification performance.

guage tasks. In Transformer models, f_θ consists of stacked feedforward and self-attention layers. The self-attention module allows each token to aggregate contextual information from the sequence.

Given projections $Q = XW_Q$, $K = XW_K$, and $V = XW_V$, with learnable parameters $W_Q, W_K, W_V \in \mathbb{R}^{d\times d_k}$, standard self-attention is computed as

$$H = \text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \quad (1)$$

The resulting attention matrix $A \in \mathbb{R}^{L\times L}$ captures pairwise token interactions. However, this formulation has two limitations. First, the attention operation has quadratic complexity $\mathcal{O}(L^2 d)$ in time and memory, which restricts scalability. Second, pairwise attention does not explicitly encode higher-order interactions among multiple tokens. To address these issues, we introduce a structured higher-order attention operator $\widetilde{H} = \mathcal{F}_{3D}(X)$, where $\mathcal{F}_{3D}$ explicitly models triplet interactions while exploiting block-diagonal sparsity for efficiency.

3.2 3D Block-Diagonal Attention: HOBA

Our 3D attention mechanism, HOBA, generalizes standard 2D self-attention by introducing a third-order interaction space. Rather than computing only pairwise interactions (x_i, x_j), HOBA explicitly models triplets (x_i, x_j, x_k), allowing the model to capture richer relational patterns. The complete 3D block-diagonal attention pipeline is illustrated in Figure 1.

For clarity, we omit the batch and attention-head dimensions below and describe the computation for a single head. In addition to the query projection, HOBA uses two key and two value projections:

$$Q = XW_Q,\ K_1 = XW_{K_1},\ K_2 = XW_{K_2},$$
$$V_1 = XW_{V_1},\ V_2 = XW_{V_2}.$$

where all projected tensors lie in $\mathbb{R}^{L\times d_k}$.

For a query token x_i and context-token pair (x_j, x_k), we define the unnormalized triplet interaction score as

$$S_{i,j,k} = \frac{1}{\sqrt{d_k}} \sum_{r=1}^{d_k} Q_{i,r} K_{1,j,r} K_{2,k,r} \quad (2)$$

which yields a third-order score tensor $S \in \mathbb{R}^{L\times L\times L}$.

For each query token i, the triplet scores are normalized jointly over the two context-token dimensions:

$$\alpha_{i,j,k} = \frac{\exp(S_{i,j,k})}{\sum_{j'=1}^{L}\sum_{k'=1}^{L}\exp(S_{i,j',k'})} \quad (3)$$

where $\alpha_{i,j,k}$ denotes the normalized attention weight assigned to the triplet (x_i, x_j, x_k).

We define the interaction between the two value vectors as

$$g(x_j, x_k) = V_{1,j} \odot V_{2,k} \quad (4)$$

where $\odot$ denotes element-wise multiplication. The contextual representation for token x_i is therefore

$$h_i = \sum_{j=1}^{L} \sum_{k=1}^{L} \alpha_{i,j,k} \left(V_{1,j} \odot V_{2,k} \right) \tag{5}$$

This formulation makes the higher-order computation explicit: x_i acts as the query token, while (x_j, x_k) jointly define a contextual interaction. However, directly computing the full $L \times L \times L$ tensor is prohibitively expensive. HOBA therefore performs the same operation within local blocks and combines the resulting block-level contextual outputs $C^{(m)}$ through overlap aggregation and cross-block interaction. The equivalent *einsum* implementation and multi-head formulation are provided in **Appendix A.3**.

Due to the computational complexity of simultaneously computing interactions between query vectors and pairs of key vectors, we implement the higher-order contractions using PyTorch's *einsum* operator. This enables an efficient implementation of the triplet score and value interaction without repeated matrix multiplications and reshaping.

3.2.1 Block-Diagonal Unrolling

HOBA uses higher-order interactions in a structured and localized manner to avoid the cubic cost of full 3D attention. Instead of operating on the full $L \times L \times L$ interaction tensor, we split the sequence $X \in \mathbb{R}^{L \times d}$ into B non-overlapping blocks of size b, such that $L = B \cdot b$.

Suppose that $L = 12$ and the block size is $b = 4$. The sequence is partitioned as

$$\begin{aligned} X^{(1)} &= [x_1, x_2, x_3, x_4], \\ X^{(2)} &= [x_5, x_6, x_7, x_8], \\ X^{(3)} &= [x_9, x_{10}, x_{11}, x_{12}]. \end{aligned}$$

For each block $X^{(m)}$, we compute a local triplet attention tensor

$$\mathcal{A}^{(m)} \in \mathbb{R}^{b \times b \times b},$$

where each entry $\mathcal{A}^{(m)}_{i,j,k}$ captures the interaction among a query token and a pair of context tokens within the same block. The corresponding contextual output is denoted by

$$C^{(m)} \in \mathbb{R}^{b \times d_k}.$$

The block-diagonal structure reduces the local higher-order complexity from

$$\mathcal{O}(L^3)$$

to

$$\mathcal{O}(Bb^3) = \mathcal{O}(Lb^2),$$

where $b \ll L$. Thus, for a fixed block size b, the local higher-order computation scales linearly with sequence length. The overall complexity additionally includes the reduced cross-block term analyzed in **Appendix A.6**.

3.2.2 Overlapping Block Structure

We extend the block partitioning strategy by introducing overlapping blocks to improve information flow across block boundaries. Each block $X^{(m)}$ spans a window determined by an overlap ratio $\lambda \in (0, 1)$. Specifically, block m covers token positions

$$[m \cdot b(1-\lambda),\ m \cdot b(1-\lambda) + b) \tag{6}$$

This overlapping scheme allows neighboring blocks to share contextual information without substantially increasing memory usage. If token x_i appears in multiple overlapping blocks, its local outputs are averaged:

$$o_i^{(\text{local})} = \frac{1}{|B_i|} \sum_{m \in B_i} o_i^{(m)} \tag{7}$$

where B_i denotes the set of blocks containing token x_i, and $o_i^{(m)}$ is the output for token x_i produced by block m.

3.2.3 Cross-Block Interaction Mechanism

Although local 3D attention captures rich intra-block dependencies, it does not directly model global interactions across distant blocks. We therefore introduce a cross-block attention mechanism based on block-level summaries.

For each block $X^{(m)}$, we compute a representative vector $r^{(m)} \in \mathbb{R}^{d_k}$ by averaging the contextual token representations within the block:

$$r^{(m)} = \frac{1}{b} \sum_{i=mb}^{(m+1)b-1} h_i \tag{8}$$

We then compute cross-block attention scores between block representatives:

$$s_{m,m'} = \frac{(r^{(m)})^\top r^{(m')}}{\sqrt{d_k}} \tag{9}$$

and normalize over the destination block index m':

$$\mathcal{A}^{(\text{cross})}_{m,m'} = \frac{\exp(s_{m,m'})}{\sum_{t=0}^{B-1} \exp(s_{m,t})} \tag{10}$$

Finally, the token representation is updated by combining the local higher-order output with the global block-level context:

$$o_i = o_i^{(\text{local})} + \sum_{m'=0}^{B-1} \mathcal{A}_{m,m'}^{(\text{cross})} r^{(m')} \quad (11)$$

where $m = \lfloor i/b \rfloor$ denotes the block containing token x_i. The final sequence representation is

$$O = [o_1, \ldots, o_L].$$

See **Appendix A.3** for the equivalent tensor-contraction implementation, multi-head aggregation, and additional implementation details.

Tasks & Datasets. We consider five task categories: sentiment analysis (SA), news classification (NC), question classification (QC), natural language inference (NLI), and question answering (QA). For each task, we evaluate on representative ten benchmark datasets (AG News, MNLI, SST, TREC, SQuAD, IMDB, Yelp Polarity, LRA-Text, HotpotQA, StrategyQA), with details provided in the **Appendix A.2**.

4 Results Analysis

We compare vanilla RoBERTa models with our proposed HOBA variants under identical training configurations to ensure fairness. All models use the same backbone architecture, hidden dimension, number of layers, training protocol, and evaluation settings unless otherwise stated. For computational comparisons, we use matched sequence lengths and model configurations so that differences primarily reflect the attention mechanism itself. All results are averaged over three independent trials. Details of the experimental setup and hyperparameter settings are provided in **Appendix A.9**.

4.1 Infeasible Full-3D Attention to Sparse and Feasible HOBA

Before scaling to large benchmarks, we first conduct a sanity check in a small-scale setting with a maximum sequence length of 64. This setup helps us isolate the source of HOBA's performance gains by evaluating the feasibility of HO attention in its pure form. In particular, we evaluate full 3D attention without block sparsity or cross-block interactions on two lightweight datasets: SST-2 (10k samples) and TREC (5k samples). This setting is intentionally simple: the sequence length is short, the number of classes is small, and the datasets are relatively limited in size. The results in Table 1 show that full 3D attention underperforms, reaching only 82–85% accuracy, and significantly below both vanilla 2D and HOBA across both 3-layer and 6-layer models. In contrast, HOBA consistently recovers accuracy, reaching 86–94% and closely matching or even surpassing the 2D baseline.

Dataset	Classes	Accuracy			Loss		
		Full-3D	HOBA	2D	Full-3D	HOBA	2D
3-Layer Model							
SST-2	2	$83.4_{\pm 0.8}$	$86.9_{\pm 0.6}$	**88.8**$_{\pm 0.5}$	$0.58_{\pm 0.03}$	$0.46_{\pm 0.02}$	$0.42_{\pm 0.02}$
TREC	3	$85.1_{\pm 0.9}$	$90.7_{\pm 0.7}$	**92.4**$_{\pm 0.6}$	$0.62_{\pm 0.04}$	$0.48_{\pm 0.03}$	$0.44_{\pm 0.02}$
6-Layer Model							
SST-2	2	$82.3_{\pm 0.7}$	$90.4_{\pm 0.5}$	**91.1**$_{\pm 0.4}$	$0.51_{\pm 0.03}$	$0.39_{\pm 0.02}$	$0.37_{\pm 0.02}$
TREC	3	$84.9_{\pm 0.8}$	$94.1_{\pm 0.5}$	**94.7**$_{\pm 0.4}$	$0.57_{\pm 0.03}$	$0.36_{\pm 0.02}$	$0.34_{\pm 0.02}$

Table 1: SST-2 and TREC results with *Full-3D*, *HOBA*, and *2D*. Full-3D underperforms even at L=64, while HOBA recovers accuracy with better stability and efficiency. HOBA outperforms Full-3D but remains comparable to 2D (due to limited sample size).

Later, in large benchmarks, HOBA achieves superior or comparable accuracy (see Table 2) in 11 out of 15 comparisons. For example, HOBA reaches 96.2% accuracy on SQuAD with 3 layers and 97.2% with 6 layers, consistently outperforming the 2D baseline. On MNLI, HOBA improves from 62.5% (2D) to 63.3% (3-layer) and further to 83.9% (12-layer), showing benefits on more complex reasoning tasks. In addition to accuracy, HOBA consistently yields lower or comparable loss values: the 12-layer HOBA loss ranges from 0.29–0.52, compared to 0.21–0.57 in 2D. Surprisingly, the 2D model performs slightly better than HOBA on the small TREC dataset (5.5K samples), due to the limited training size. Since HO attention is designed to capture complex token triplet interactions, its expressive capacity may be underutilized on small datasets. Overall, HOBA provides efficient and robust modeling of token triplets within localized blocks, offering deeper contextual knowledge and outperforming standard 2D attention under more challenging training conditions.

Practical Insight: Training Stability

In limited-data settings, scratch training may exhibit higher gradient noise and slower convergence during early updates. We recommend using knowledge distillation as a practical stabilization strategy in such cases.

Deeper Models on Reasoning Tasks. To test whether HOBA generalizes to deeper architectures and reasoning tasks, we evaluate it using large Transformer backbones (Table 3). On *HotpotQA*

Dataset	Classes	Accuracy		Loss	
		HOBA	2D	HOBA	2D
3-Layer Model					
AG News	4	91.3$_{\pm 0.6}$	**92.1**$_{\pm 0.4}$	0.56$_{\pm 0.03}$	0.28$_{\pm 0.01}$
MNLI	3	**63.3**$_{\pm 0.8}$	62.5$_{\pm 0.9}$	0.74$_{\pm 0.03}$	0.61$_{\pm 0.03}$
SST	2	**79.4**$_{\pm 0.9}$	79.0$_{\pm 0.6}$	0.54$_{\pm 0.04}$	0.90$_{\pm 0.04}$
TREC	5	90.7$_{\pm 0.7}$	**94.2**$_{\pm 0.5}$	0.44$_{\pm 0.03}$	0.40$_{\pm 0.05}$
SQuAD	4	**96.2**$_{\pm 0.5}$	93.2$_{\pm 0.6}$	0.41$_{\pm 0.03}$	0.56$_{\pm 0.03}$
6-Layer Model					
AG News	4	**92.3**$_{\pm 0.6}$	92.2$_{\pm 0.5}$	0.47$_{\pm 0.03}$	0.36$_{\pm 0.02}$
MNLI	3	**76.2**$_{\pm 0.7}$	74.6$_{\pm 0.7}$	0.71$_{\pm 0.02}$	0.98$_{\pm 0.05}$
SST	2	**80.7**$_{\pm 0.6}$	80.1$_{\pm 0.4}$	0.45$_{\pm 0.02}$	0.45$_{\pm 0.03}$
TREC	5	89.4$_{\pm 0.8}$	**95.4**$_{\pm 0.5}$	0.55$_{\pm 0.03}$	0.16$_{\pm 0.01}$
SQuAD	4	**97.2**$_{\pm 0.4}$	95.4$_{\pm 0.4}$	0.44$_{\pm 0.03}$	0.76$_{\pm 0.04}$
Full-Layer Model (12 Layers)					
AG News	4	**94.6**$_{\pm 0.4}$	94.2$_{\pm 0.4}$	0.31$_{\pm 0.03}$	0.33$_{\pm 0.02}$
MNLI	3	**83.9**$_{\pm 0.5}$	82.2$_{\pm 0.6}$	0.52$_{\pm 0.02}$	0.57$_{\pm 0.03}$
SST	2	**84.3**$_{\pm 0.4}$	83.0$_{\pm 0.5}$	0.34$_{\pm 0.02}$	0.36$_{\pm 0.02}$
TREC	5	94.0$_{\pm 0.5}$	**96.1**$_{\pm 0.4}$	0.29$_{\pm 0.05}$	0.21$_{\pm 0.01}$
SQuAD	4	**98.2**$_{\pm 0.3}$	97.6$_{\pm 0.3}$	0.30$_{\pm 0.02}$	0.41$_{\pm 0.02}$

Table 2: Performance comparison between HOBA and standard 2D attention models across five datasets for shallow (3-layer), intermediate (6-layer), and full (12-layer) architectures.

(6k train / 600 dev), a *24-layer RoBERTa-Large* model improves EM from 19.1 to 19.9 and F1 from 25.2 to 26.1. On *StrategyQA*, integrating HOBA into *DeBERTa-Large* improves accuracy from 65.8% to 67.1% while reducing loss. Thus, HOBA scales effectively to deeper transformer models and improves performance on reasoning tasks that require modeling complex relational dependencies between multiple tokens.

Model	Acc / EM	F1	Loss
HotpotQA (RoBERTa-Large, 24 Layers)			
2D Attention	19.1$_{\pm 0.2}$	25.2$_{\pm 0.2}$	1.82$_{\pm 0.03}$
HOBA	**19.9**$_{\pm 0.4}$	**26.1**$_{\pm 0.3}$	1.73$_{\pm 0.04}$
StrategyQA (DeBERTa-Large, 24 Layers)			
DeBERTa-Large (2D)	65.8$_{\pm 0.4}$	–	0.62$_{\pm 0.02}$
DeBERTa-Large + HOBA	**67.1**$_{\pm 0.5}$	–	0.58$_{\pm 0.02}$

Table 3: Scaling evaluation of HOBA on deeper architectures and reasoning benchmarks.

The gains on HotpotQA and StrategyQA are consistent with HOBA's explicit triplet inductive bias: rather than composing pairwise relations only across layers, HOBA can score three-token interactions directly within each block, which is useful for patterns such as *entity–bridge–answer* or *entity–relation–attribute*. Cross-block exchange then propagates these HO features across layers, enabling local triplet evidence to contribute to longer reasoning chains. The improvements are modest, which we attribute to HOBA's structured and efficiency-oriented design rather than unrestricted global 3D attention, Thus, these results support improved HO relational modeling without claiming that triplet attention alone solves long-range reasoning.

Comparison with Recent 2D Attention. To the best of our knowledge, no prior work has explored 3D attention mechanisms due to their high computational cost. Thus, we compare HOBA against recent 2D-based models, WIGRAPH-RoBERTa (Sekhon et al., 2023), TransEvolve-fullFF-1 (Dutta et al., 2021), and E2LN(R) (Zhang et al., 2024) which closely align with our task objectives. We ensure fair comparison by utilizing identical configurations for all models, training each for five epochs with sequence length 128 for AG News and 512 for IMDB. We find that HOBA reaches higher accuracy(see Table 4) in all cases by demonstrating the benefits of modeling triplet interactions. We exclude popular methods such as **FlashAttention** (Dao et al., 2022), **Mamba** (Zhu et al., 2024) from direct benchmarking, as they focus on kernel-level optimizations or replace attention entirely with state-space operators. Our work instead targets structural improvements to attention through HO block interactions, making these approaches complementary rather than directly comparable.

Model	Dataset	Theirs/HOBA	Focus	Interaction
WIGRAPH (AAAI '23)	AG News	91.52% / **95.22%**	Word interaction learning for 2D interaction	2D
TransEvolve (NIPS '21)	AG News	90.10% / **91.63%**	Parameter reduction for quadratic attention	2D
E2LN(R) (ACL '24)	IMDB	59.23% / **90.41%**	Efficient LayerNorm personalization	2D

Table 4: Accuracy comparison between our HOBA model and recent 2D attention methods.

Comparison with Different Sparse Attention and Long-Context Benchmarks. We test HOBA on long-sequence sentiment classification tasks using the IMDB, LRA-Text and Yelp Polarity datasets against several sparse mechanisms such as Longformer (Beltagy et al., 2020), BigBird (Zaheer et al., 2020), Local (Parmar et al., 2018), Block-diagonal (like ours approach), and Dilated attention (Hassani and Shi, 2022). These methods are widely adopted because their sparsity patterns reduce quadratic costs while preserving context. We choose these patterns for a fair comparison as they capture different trade-offs between locality, global context, and efficiency. The sparse baselines are configured with their typical hyperparameters. For instance, Longformer uses a sliding window of 512, BigBird combines 10% random

+ 10% global + 80% local links, Local attention uses a fixed window of 128, and Dilated attention uses a dilation rate of 2, ensuring settings align with our experimental context. As shown in Table 5, HOBA consistently achieves the best accuracy at L=512, reaching 93.1% on IMDB and 97.2% on Yelp, while also providing faster training than Longformer and Local attention.

Across long context settings, HOBA remains stable and competitive even as sequence lengths increase to 1024 and 2048 tokens. On Yelp Polarity, HOBA consistently outperforms Longformer and remains close to BigBird even at 2k tokens, while maintaining significantly lower inference time. On the more challenging LRA–Text benchmark, HOBA again matches or exceeds Longformer and stays within a narrow margin of BigBird, despite using a simpler triplet formulation and without any specialized long-range architectural tuning. The main benefit comes from HOBA's ability to model richer token relationships using structured 3D attention over triplets. Unlike sparse models, which reduce complexity by restricting which tokens attend to each other, our HOBA model captures detailed local interactions through block-diagonal attention. This allows better context modeling without requiring long sequences or extensive computation. Therefore, even without support for sequences longer than 512, HOBA exceeds both Longformer and BigBird within their optimal settings (i.e., longer sequence length = 2048).

Efficiency Gains with HOBA. We sweep hidden size (model width) and plot accuracy against total FLOPs while holding all other hyperparameters fixed: 2 layers, 4 heads, FFN ratio $4H$, block size = 32, sequence length L=128, identical data, optimizer, epochs, and training schedule for a fair comparison of HOBA to standard 2D attention. FLOPs include attention, QKV/OUT projections, and FFN. For the smallest width setting (H=64), both HOBA and 2D RoBERTa have roughly 3.3M parameters, with memory usage of 13.2 MB for HOBA and 14.1 MB for 2D. The throughput is about 1.2k tokens/s for 2D and 1.6k tokens/s for HOBA. Figure 2(a) shows that **HOBA achieves better accuracy at lower compute across all widths**. Concretely, HOBA reduces FLOPs by $\approx 8-12\%$ relative to 2D (e.g., 33.6M→30.2M at width 64; 337.6M→304.0M at width 224) while improving accuracy by $+1.5$–3.3 points (e.g., $74.9\% \rightarrow 76.4\%$, $85.1\% \rightarrow 88.4\%$). Because projections and the FFN dominate at short contexts (L=128), the observed savings arise solely from the attention mechanism; the gap is expected to widen for longer sequences where attention costs dominate. The global Pareto frontier therefore shifts upward with HOBA, implying a better efficiency–performance trade-off than standard 2D attention at matched training conditions.

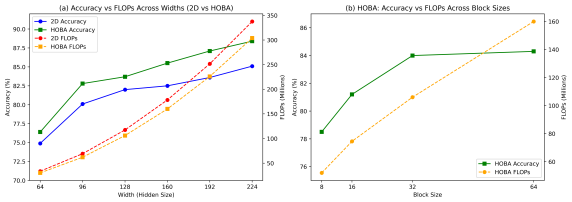


Figure 2: Computation vs. accuracy on AG News. (a) Accuracy and FLOPs across hidden widths for 2D vs. HOBA, showing that HOBA consistently achieves higher accuracy with lower compute. (b) Accuracy improves as block size increases, but saturates beyond b=32, identifying it as the best trade-off point.

Generalization Bound via Rademacher Complexity. We analyze why HOBA yields stronger generalization than standard 2D attention using Rademacher complexity (Bartlett and Mendelson, 2002; Mohri et al., 2018). For a loss ℓ bounded in $[0, 1]$ and L_ℓ-Lipschitz, the standard bound states that with probability at least $1-\delta$, for any $h \in \mathcal{H}$,

$$\mathcal{L}(h) \leq \hat{\mathcal{L}}(h) + 2L_\ell \hat{\mathfrak{R}}_n(\mathcal{H}) + 3\sqrt{\frac{\ln(2/\delta)}{2n}}. \quad (12)$$

where $\hat{\mathfrak{R}}_n(\mathcal{H})$ is the empirical Rademacher complexity. Since $\mathcal{H}_{\text{HOBA}} \subseteq \mathcal{H}_{\text{Full}}$ by construction (block-diagonal masking strictly restricts token interactions), monotonicity of Rademacher complexity implies

$$\hat{\mathfrak{R}}_n(\mathcal{H}_{\text{HOBA}}) \leq \hat{\mathfrak{R}}_n(\mathcal{H}_{\text{Full}}). \quad (13)$$

Therefore, whenever the empirical risk of HOBA is comparable to or lower than that of standard 2D attention, the reduced hypothesis class size guarantees a strictly tighter generalization bound. This progress stems from enforcing structured block-diagonal and overlapping attention masks, which limit the effective hypothesis space without sacrificing expressivity. Empirical evidence supports this theoretical claim (see Table 5, 6, 1, and 2). HOBA not only improves accuracy over 2D but also yields consistently lower training loss across datasets, confirming that the restricted hypothesis class translates into stronger generalization in practice (see **Appendix A.1** for the supporting lemma).

Dataset	L	HOBA (Ours)	Longformer	BigBird	2D Block-Diagonal	2D Local	2D Dilated
		Acc/Loss/Time	Acc/Loss/Time	Acc/Loss/Time	Acc/Loss/Time	Acc/Loss/Time	Acc/Loss/Time
IMDB	512	**93.1/0.23/10m**	92.1/0.37/26m	90.4/0.22/15m	77.7/0.51/37m	83.2/0.43/39m	85.0/0.40/35m
	2048	–	92.0/0.24/54m	92.1/0.28/38m	–	–	–
Yelp Polarity	512	**97.2/0.09/280m**	92.0/0.68/690m	95.0/0.08/**166m**	88.8/0.30/693m	93.4/0.23/711m	91.5/0.26/705m
	1024	**96.4/0.12/420m**	92.1/0.70/1120m	95.3/0.10/240m	–	–	–
	2048	**95.9/0.16/610m**	93.2/0.61/1409m	96.1/0.08/240m	–	–	–
LRA–Text	1024	**64.1/1.21/95m**	63.8/1.27/210m	65.2/1.19/160m	57.8/1.40/260m	60.3/1.32/275m	62.1/1.29/250m
	2048	**63.4/1.25/180m**	63.1/1.32/380m	65.0/1.22/300m	–	–	–

Table 5: Comparison of HOBA (ours), Longformer, BigBird, and 2D attention baselines (2D Block-Diagonal, 2D Local, 2D Dilated; all without HO) on IMDB, Yelp Polarity, and LRA–Text across sequence lengths $L = 512$–2048.

Analysis of Higher-Order Token Interactions. To better understand how HOBA models HO token relationships, we visualize representative triplet interactions discovered by the model. Figure 3 illustrates three example triplets extracted from the sentence "*Global markets showed uncertainty amid economic volatility*." Each panel highlights a higher-order interaction (i, k, j), where the intermediate token k acts as a mediator that strengthens the relationship between tokens i and j. In standard 2D attention, interactions are computed directly between token pairs, which can dilute relationships when contextual dependencies involve multiple tokens. In contrast, HOBA introduces a triplet-mediated mechanism where intermediate tokens reinforce semantically related pairs. For example, in the triplet (markets, showed, uncertainty), the token "showed" acts as a mediator that strengthens the interaction between "markets" and "uncertainty." Similarly, the triplet (economic, markets, volatility) captures a meaningful contextual dependency that is reinforced through the mediator token "markets." These examples show how HOBA uses mediator tokens to strengthen semantically coherent relationships, enabling richer higher-order dependencies beyond pairwise attention. Additional visualizations are provided in **Appendix A.4**.

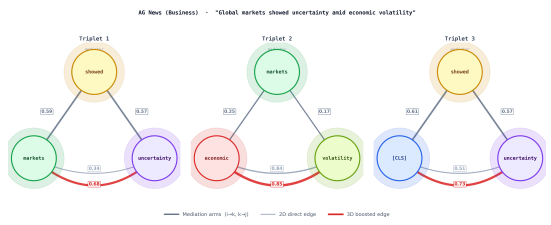


Figure 3: HOBA strengthens token relations through mediator-based triplets (i, k, j), where red edges indicate enhanced interactions compared to standard 2D attention.

4.2 Ablation Study

Impact of Cross-Block Interaction. Cross-block (CB) interaction in HOBA allows the model to connect information between distant token blocks. We find that enabling CB interaction consistently improves accuracy for both 2D and HOBA, but the gains are far more pronounced in HOBA (see Table 6). MNLI involves reasoning across sentence pairs; therefore, performance depends on connecting information that might sit far apart. Without CB interaction, those links break, and attention becomes limited. HOBA with CB forms triplets spanning the full sequence by allowing signals to combine across distant blocks. This strengthens reasoning, lessens noise, and promotes consistent patterns across layers. These results show that CB strengthens long-range reasoning and that the combination of HOBA + CB provides the most consistent performance improvements.

Dataset	Classes	2D w/o CB	2D w/ CB	HOBA w/ CB	HOBA w/o CB
AG News	4	88% / 0.83	89% / 0.52	**92% / 0.51**	90% / 0.53
SST-2	2	76% / 0.50	77% / 0.49	**79% / 0.49**	78% / 0.50
MNLI	3	51% / 0.73	53% / 0.71	**64% / 0.69**	57% / 0.73

Table 6: Ablation study comparing 2D (with and without CB) and HOBA variants (with and without CB). Results are accuracy/loss (block size 16, 3 layers).

Effect of Block Size. At fixed context (L=128) and capacity (2 layers, width 128, 4 heads, FFN $4H$) increasing the HOBA block size from b=8, 16, 32, 64 steadily improves accuracy from 78.5% to 84.3%, while FLOPs grow roughly linearly in b (52M to 160M)(see Figure 2(b)). Gains saturate beyond b=32 (only +0.3 points at b=64 for $\sim$1.5$\times$ more compute), identifying b=32 as the best accuracy–compute trade-off under this setting. We therefore suggest using b=32 unless longer-range within-block context is required.

Effect of Sequence Length. Table 7 indicates that performance steadily improves as sequence

length increases from 128 to 384. This trend is evident in tasks requiring deeper contextual understanding like MNLI, where accuracy increases from 51% to 59% and loss drops notably. This trend also reveals HOBA's capacity to capture broader dependencies with its HO structure. Even on SST-2, which has shorter inputs, increasing the sequence length to 256 increases accuracy.

Dataset	SL=128	SL=256	SL=384
AG News	90% / 0.46	91% / 0.48	**93% / 0.43**
MNLI	51% / 0.72	51% / 0.69	**59% / 0.53**
SST-2	77% / 0.43	**78% / 0.34**	N/A

Table 7: Sequence length ablation for HOBA (block=32, 6 layers). Accuracy / loss shown.

Comparison with a 2D Block-Local Baseline. We construct a controlled experiment (block size $b = 32$, overlap $\lambda = 0.25$).) comparing a 2D block-local attention model with HOBA, using the same block size, overlap ratio, and cross-block routing. The only difference is that the 2D baseline performs pairwise (2D) attention, whereas HOBA performs triplet-wise (3D) attention. The results indicate that HOBA consistently outperforms the matched 2D baseline, demonstrating that the gains arise from HO (3D) attention rather than the block-local structure alone (Table 8).

Depth	Model	Acc. (%)	Loss
3-layer	2D Block	88.6	0.23
3-layer	HOBA	**91.9**	**0.21**
6-layer	2D Block	90.1	0.34
6-layer	HOBA	**93.3**	**0.34**

Table 8: 2D block-local attention vs. HOBA.

Additional ablations on the overlap ratio are reported in **Appendix A.5**.

4.3 Key Difference with SOTA Methods

We find that all these prior higher-order attention models (Table 11 in **Appendix A.7**) operate globally (spectral kernels, full-sequence graphs, dense tensor contractions), resulting in fundamentally different computational objectives and scaling regimes. We find that their complexity is much higher than HOBA's block-local 3D attention, making direct comparison neither methodologically fair nor computationally feasible under matched FLOPs. However, Sparse-attention models (Longformer, BigBird, etc.) are the appropriate baselines because, like our HOBA, they are specifically designed to reduce the quadratic cost of self-attention while preserving long-range dependencies. They target the same efficiency–accuracy trade-off, operate under similar computational budgets, and scale to long sequences. Thus, it makes them conceptually and empirically aligned with HOBA's goals.

5 Conclusion

We propose HOBA, a higher-order block-diagonal attention method that models token triplet interactions to improve contextual representations in transformer models with computational efficiency. By using block-local 3D attention, HOBA captures richer dependencies than pairwise attention while remaining efficient. Cross-block interactions further enable global reasoning with minimal overhead. Experiments on ten NLP benchmarks show that HOBA achieves competitive or superior performance with lower resource usage than dense and sparse attention baselines. Our analysis highlights the interpretability benefits of triplet interactions and their role in modeling long-range semantic dependencies. Future work will explore adaptive HO attention with dynamic block structures and more stable training strategies.

6 Limitations

While block-diagonal unrolling reduces the computational cost of HO attention, HOBA can still be less efficient than some sparse attention mechanisms under strict latency or memory constraints, particularly for very long sequences. Our experiments primarily focus on moderate to long sequence lengths, and we do not claim that HOBA is optimized for significantly longer contexts. Instead, the primary contribution of HOBA lies in improved expressiveness through modeling triplet interactions beyond pairwise attention; enhanced long-range dependency modeling should be viewed as an additional benefit rather than the central claim.

7 Acknowledgement

This research work is supported by the US NSF awards 2205891, 2428761, and the US Army Research Office under Cooperative Agreement W911NF-24-2-0133.

8 Ethical Considerations

This work focuses on improving the efficiency and expressiveness of transformer architectures and does not introduce new datasets or involve human subjects. HOBA is evaluated only on standard, publicly available NLP benchmarks. As with other transformer-based models, HOBA may inherit biases present in the training data or teacher models used for knowledge distillation. We do not claim that HO attention mitigates such biases, and caution should be exercised when deploying HOBA in high-stakes or sensitive applications. Additionally, while HOBA enhances computational efficiency relative to full HO, large-scale deployment may still incur nontrivial energy and resource costs, underscoring the importance of responsible model usage.

References

Gustavo Aguilar, Yuan Ling, Yu Zhang, Benjamin Yao, Xing Fan, and Chenlei Guo. 2020. Knowledge distillation from internal representations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7350–7357.

Peter L Bartlett and Shahar Mendelson. 2002. Rademacher and gaussian complexities: Risk bounds and structural results. *Journal of machine learning research*, 3(Nov):463–482.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Charaf Eddine Benarab and Shenglin Gui. 2022. Cnn-trans-enc: A cnn-enhanced transformer-encoder on top of static bert representations for document classification. *arXiv preprint arXiv:2209.06344*.

Yuyang Chai, Zhuang Li, Jiahui Liu, Lei Chen, Fei Li, Donghong Ji, and Chong Teng. 2024. Compositional generalization for multi-label text classification: A data-augmentation approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17727–17735.

Cheng Chen, Yichun Yin, Lifeng Shang, Zhi Wang, Xin Jiang, Xiao Chen, and Qun Liu. 2021. Extract then distill: Efficient and effective task-agnostic bert distillation. In *Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14–17, 2021, Proceedings, Part III 30*, pages 570–581. Springer.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, and 1 others. 2020. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*.

Baiyun Cui, Yingming Li, Ming Chen, and Zhongfei Zhang. 2019. Fine-tune bert with sparse self-attention mechanism. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 3548–3553.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. 2022. Flashattention: Fast and memory-efficient exact attention with io-awareness. In *Advances in Neural Information Processing Systems 35*, pages 16344–16359.

Subhabrata Dutta, Tanya Gautam, Soumen Chakrabarti, and Tanmoy Chakraborty. 2021. Redesigning the transformer architecture with insights from multi-particle dynamical systems. *Advances in Neural Information Processing Systems*, 34:5531–5544.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Ali Hassani and Humphrey Shi. 2022. Dilated neighborhood attention transformer. *arXiv preprint arXiv:2209.15001*.

Giwon Hong, Jeonghwan Kim, Junmo Kang, and Sung-Hyon Myaeng. 2022. Graph-induced transformers for efficient multi-hop question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10288–10294.

Feyza Duman Keles, Pruthuvi Mahesakya Wijewardena, and Chinmay Hegde. 2023. On the computational complexity of self-attention. In *International conference on algorithmic learning theory*, pages 597–619. PMLR.

Andriy Kosar, Guy De Pauw, and Walter Daelemans. 2023. Advancing topical text classification: a novel distance-based method with contextual embeddings. In *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 586–597.

Jiagen Li, Rui Yu, Huihao Huang, and Huaicheng Yan. 2025. Unimodal-driven distillation in multimodal emotion recognition with dynamic fusion. *arXiv preprint arXiv:2503.23721*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Valerii Likhosherstov, Krzysztof Choromanski, and Adrian Weller. 2023. On the expressive flexibility of self-attention matrices. In *Proceedings of*

the AAAI Conference on Artificial Intelligence, volume 37, pages 8773–8781.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2023. Roberta: A robustly optimized bert pretraining approach. *Journal of Machine Learning Research*, 24:1–48.

Michael Livanos, Ian Davidson, and Stephen Wong. 2024. Cooperative knowledge distillation: A learner agnostic approach. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14124–14131.

Chengqiang Lu, Jianwei Zhang, Yunfei Chu, Zhengyu Chen, Jingren Zhou, Fei Wu, Haiqing Chen, and Hongxia Yang. 2022. Knowledge distillation of transformer-based language models revisited. *arXiv preprint arXiv:2206.14366*.

Hui Ma, Jian Wang, Hongfei Lin, Bo Zhang, Yijia Zhang, and Bo Xu. 2023. A transformer-based model with self-distillation for multimodal emotion recognition in conversations. *IEEE Transactions on Multimedia*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. 2018. *Foundations of machine learning*. MIT press.

Krenare Pireva Nuci, Paul Landes, and Barbara Di Eugenio. 2024. Roberta low resource fine tuning for sentiment analysis in albanian. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14146–14151.

Soroush Omranpour, Guillaume Rabusseau, and Reihaneh Rabbany. 2024. Higher order transformers: Efficient attention mechanism for tensor structured data. *arXiv preprint arXiv:2412.02919*.

Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. 2018. Image transformer. In *International conference on machine learning*, pages 4055–4064. PMLR.

Avinash Patil and Aryan Jadon. 2025. Advancing reasoning in large language models: Promising methods and approaches. *arXiv preprint arXiv:2502.03671*.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. Adapterhub: A framework for adapting transformers. *arXiv preprint arXiv:2007.07779*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Sharath Chandra Raparthy, Eric Hambro, Robert Kirk, Mikael Henaff, and Roberta Raileanu. 2023. Learning to solve new sequential decision-making tasks with in-context learning. In *NeurIPS 2023 Foundation Models for Decision Making Workshop*.

Aurko Roy, Mohammad Saffar, Ashish Vaswani, and David Grangier. 2021. Efficient content-based sparse attention with routing transformers. *Transactions of the Association for Computational Linguistics*, 9:53–68.

Arshdeep Sekhon, Hanjie Chen, Aman Shrivastava, Zhe Wang, Yangfeng Ji, and Yanjun Qi. 2023. Improving interpretability via explicit word interaction graph layer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 13528–13537.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Yi Tay, Mostafa Dehghani, Samira Abnar, Yikang Shen, Dara Bahri, Philip Pham, Jinfeng Rao, Liu Yang, Sebastian Ruder, and Donald Metzler. 2020. Long range arena: A benchmark for efficient transformers. *arXiv preprint arXiv:2011.04006*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*, pages 5998–6008.

Sinong Wang, Belinda Z. Li, Madian Khabsa, Han Fang, and Hao Ma. 2020. Linformer: Self-attention with linear complexity. *arXiv preprint arXiv:2006.04768*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long papers)*, pages 1112–1122.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Xufeng Yao, Fanbin Lu, Yuechen Zhang, Xinyun Zhang, Wenqian Zhao, and Bei Yu. 2024. Progressively

knowledge distillation via re-parameterizing diffusion reverse process. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 16425–16432.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: Transformers for longer sequences. In *Advances in Neural Information Processing Systems 33*, pages 17283–17297.

Jianhua Zhang, Yi Gao, Ruyu Liu, Xu Cheng, Houxiang Zhang, and Shengyong Chen. 2025. Can students beyond the teacher? distilling knowledge from teacher's bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 22434–22442.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

You Zhang, Jin Wang, Liang-Chih Yu, Dan Xu, and Xuejie Zhang. 2024. Improving personalized sentiment representation with knowledge-enhanced and parameter-efficient layer normalization. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8877–8889.

Haoyi Zhou, Siyang Xiao, Shanghang Zhang, Jieqi Peng, Shuai Zhang, and Jianxin Li. 2022. Jump self-attention: Capturing high-order statistics in transformers. *Advances in Neural Information Processing Systems*, 35:17899–17910.

Lianghui Zhu, Bencheng Liao, Qian Zhang, Xinlong Wang, Wenyu Liu, and Xinggang Wang. 2024. Vision mamba: Efficient visual representation learning with bidirectional state space model. *arXiv preprint arXiv:2401.09417*.

A Appendix

A.1 Proof of Lemma 1

Lemma 1 (HOBA is a restricted sub-class of full attention)**.** *Let $\mathcal{H}_{Full}$ be the hypothesis class induced by a transformer with unconstrained (dense) self-attention, and let $\mathcal{H}_{HOBA}$ denote the same architecture with block-diagonal or overlapping attention masks. Then $\mathcal{H}_{HOBA} \subseteq \mathcal{H}_{Full}$.*

Setup. Consider a single attention layer with per-head dimension d and additive mask $M \in \mathbb{R}^{L\times L}$. The masked attention output is

$$\mathrm{Attn}(Q,K,V;M) = \mathrm{softmax}\Big(\tfrac{1}{\sqrt{d}}QK^\top + M\Big)V, \qquad (14)$$

where $Q, K, V \in \mathbb{R}^{L\times d}$ are the query, key, and value projections. Let $f(X;\Theta,M)$ denote the model function with parameters Θ and attention mask M, consistent with the general $f_\theta(X)$ formulation in the main text. Then $f(x;\Theta,M)$ specifies a depth-D transformer (with fixed depth and hidden size for both models) obtained by composing layers of the masked attention in Eq. (11).

Classes.

$$\begin{aligned}\mathcal{M}_{\text{Full}} &:= \mathbb{R}^{L\times L},\\ \mathcal{M}_{\text{HOBA}} &:= \big\{M \in \mathbb{R}^{L\times L} :\ M_{ij} = -\infty \ \text{ if } (i,j) \text{ lies outside}\\ &\qquad \text{allowed block positions}\big\},\end{aligned} \qquad (15)$$

$$\begin{aligned}\mathcal{H}_{\text{Full}} &:= \big\{\, x \mapsto f(x;\Theta,M) : \Theta \text{ arbitrary},\ M \in \mathcal{M}_{\text{Full}}\big\},\\ \mathcal{H}_{\text{HOBA}} &:= \big\{\, x \mapsto f(x;\Theta,M) : \Theta \text{ arbitrary},\ M \in \mathcal{M}_{\text{HOBA}}\big\}.\end{aligned} \qquad (16)$$

Inclusion. By construction, $\mathcal{M}_{\text{HOBA}} \subseteq \mathcal{M}_{\text{Full}}$, since block-diagonal or overlapping masks are a strict subset of all possible real-valued masks. Hence, for any (Θ, M) with $M \in \mathcal{M}_{\text{HOBA}}$, the same pair (Θ, M) is admissible in $\mathcal{H}_{\text{Full}}$. This implies $f(\cdot;\Theta,M) \in \mathcal{H}_{\text{Full}}$, and therefore every function realizable by HOBA is realizable by the full-attention model:

$$\mathcal{H}_{\text{HOBA}} \subseteq \mathcal{H}_{\text{Full}}.$$

The argument extends directly to multi-head and multi-layer transformers, since masks are applied independently per head and per layer, and compositionality preserves inclusion. □

A.2 More on Datasets

The datasets used in our experiments span multiple NLP tasks, including news categorization (AG News (Zhang et al., 2015)), natural language inference (MNLI (Williams et al., 2018)), sentiment analysis (SST (Socher et al., 2013), IMDB (Maas et al., 2011), Yelp Polarity (Zhang et al., 2015), LRA-Text (Tay et al., 2020)), question classification (TREC (Li and Roth, 2002)), and question answering (SQuAD (Rajpurkar et al., 2016)) (see Table 9). They vary widely in both scale and input length: AG News and MNLI contain medium-length sequences with median sizes of about 35–40 tokens, SST and TREC are shorter with median lengths under 20 and 10 tokens respectively, while SQuAD passages are longer with a median around 150 tokens. IMDB and Yelp Polarity consist of full-length user reviews, exhibiting much greater input size, with medians of approximately 230 and 150 tokens. LRA-Text evaluates long-range dependency modeling with sequence lengths of up

to 4,096 tokens. HotpotQA (Yang et al., 2018) is a multi-hop question answering dataset where answers require reasoning across multiple supporting documents, with an average context length of approximately 120–150 tokens. StrategyQA (Geva et al., 2021) is a compositional reasoning benchmark involving implicit multi-step reasoning over commonsense knowledge, with an average input length of roughly 45–60 tokens. This diversity in task type, size, and sequence length provides a comprehensive benchmark for evaluating both efficiency and effectiveness of attention mechanisms.

Dataset	Task	Class	Train	Test
AG News	NC	4	120K	7.6K
MNLI	NLI	3	393K	9.8K
SST	SA	2	67K	1.8K
TREC	QC	5	5.5K	500
SQuAD	QA	4	87K	10.6K
IMDB	SA	2	25K	25K
LRA–Text	SA	2	250K	25K
Yelp Polarity	SA	2	260K	20K

Table 9: Dataset Statistics and Task Descriptions

A.3 Block-wise Tensor Attention

Our HOBA takes an input sequence $X \in \mathbb{R}^{L\times d}$ and projects it into query, key, and value subspaces as $Q, K_1, K_2, V_1, V_2 \in \mathbb{R}^{L\times d}$ through learned linear transformations (Algorithm 1). The sequence is divided into overlapping blocks $\mathcal{B}$ of size b with overlap ratio λ, and for each block $(s, e) \in \mathcal{B}$ we extract the block tensors $Q^{(m)}, K_1^{(m)}, K_2^{(m)}, V_1^{(m)}, V_2^{(m)} \in \mathbb{R}^{b\times d}$.

Higher-order interaction tensor. Within each block, the 3D attention interaction is computed using the tensor contraction

$$\mathcal{T}^{(m)} = \text{einsum}(\texttt{"id,jd,kd->ijk"},$$
$$Q^{(m)}, K_1^{(m)}, K_2^{(m)}) \in \mathbb{R}^{b\times b\times b},$$

where i indexes the query token, (j, k) index paired key tokens, and the embedding dimension d is contracted out. Softmax over the (j, k) axes yields the triplet attention tensor $\mathcal{A}^{(m)} \in \mathbb{R}^{b\times b\times b}$.

Value interaction and block output. The corresponding value interaction tensor is

$$\mathcal{U}^{(m)} = \text{einsum}(\texttt{"jd,kd->jkd"},$$
$$V_1^{(m)}, V_2^{(m)}) \in \mathbb{R}^{b\times b\times d}.$$

The contextual block output is then obtained by

$$C^{(m)} = \text{einsum}(\texttt{"ijk,jkd->id"},$$
$$\mathcal{A}^{(m)}, \mathcal{U}^{(m)}) \in \mathbb{R}^{b\times d},$$

which provides one contextual representation per token inside the block.

Aggregation and multi-head computation. All $C^{(m)}$ outputs are accumulated into the global tensor O with overlap handling. In the multi-head setting, the same computation is applied independently for each head h to obtain $C_h^{(m)}$, after which the outputs are concatenated across heads and passed through the usual output projection W_O. A pooled summary representation R is obtained via BLOCKPOOLING and incorporated through CROSSBLOCKATTENTION, yielding the final block-updated representation.

A.4 Comparison of Token-Level Attention Patterns

We visualize the interaction graphs generated by HOBA and standard 2D models to understand the qualitative differences in token-level attention patterns. Comparison of 3D vs 2D attention patterns shown in Figure 4.

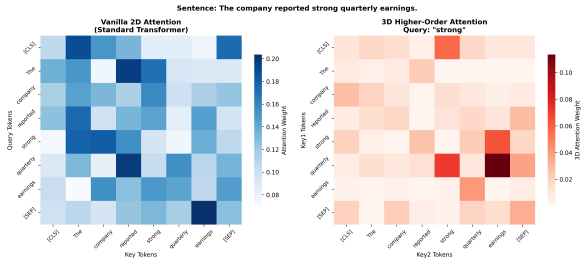


Figure 4: Comparison of attention patterns between standard 2D attention (left) and our proposed 3D higher-order attention (right). The 2D plot displays traditional pairwise query-key token interactions, while the 3D plot shows a key$_1$ $\times$ key$_2$ slice for the query token *"strong"* by capturing triplet interactions. HOBA assigns a higher weight to semantically relevant triplets such as (*"strong," "quarterly,"* and *"earnings"*) and reflects its ability to model richer contextual dependencies.

A.5 More Ablation

Impact of Overlap Ratio Table 10 analyzes the impact of different overlap ratios on HOBA performance. An overlap ratio of $\lambda = 0.25$ consistently yields the best results on AG News and MNLI, suggesting an effective balance between local block context and cross-block information flow. Larger

Algorithm 1 Block-Diagonal Higher-Order Attention

Require: Input $X \in \mathbb{R}^{L \times d}$, block size b, overlap ratio λ
Ensure: Output $O \in \mathbb{R}^{L \times d}$

// Project to attention subspaces
$Q, K_1, K_2, V_1, V_2 \leftarrow XW_Q, XW_{K_1}, XW_{K_2}, XW_{V_1}, XW_{V_2}$
// Initialize output and generate overlapping blocks
$O \leftarrow 0_{L \times d}$, $\mathcal{B} \leftarrow \textsc{GenerateBlocks}(L, b, \lambda)$
for each block $(s, e) \in \mathcal{B}$ **do**
  // Extract block tensors
  $Q^{(m)}, K_1^{(m)}, K_2^{(m)}, V_1^{(m)}, V_2^{(m)} \leftarrow Q[s:e], K_1[s:e], K_2[s:e], V_1[s:e], V_2[s:e]$
  // Compute 3D attention tensor
  $\mathcal{T}^{(m)} \leftarrow \textsc{Einsum}(\texttt{"id, jd, kd->ijk"}, Q^{(m)}, K_1^{(m)}, K_2^{(m)})/\sqrt{d}$
  $\mathcal{A}^{(m)} \leftarrow \textsc{Softmax}_{(j,k)}(\mathcal{T}^{(m)})$
  // Value interaction tensor
  $\mathcal{U}^{(m)} \leftarrow \textsc{Einsum}(\texttt{"jd, kd->jkd"}, V_1^{(m)}, V_2^{(m)})$
  // Contextual block output
  $C^{(m)} \leftarrow \textsc{Einsum}(\texttt{"ijk, jkd->id"}, \mathcal{A}^{(m)}, \mathcal{U}^{(m)})$
  // Accumulate with overlap
  $O[s:e] \leftarrow O[s:e] + C^{(m)}$
end for
// Cross-block interaction
$R \leftarrow \textsc{BlockPooling}(O, \mathcal{B})$
$O \leftarrow O + \textsc{CrossBlockAttention}(O, R)$
// Normalize overlaps and project
$O \leftarrow \textsc{NormalizeOverlaps}(O)W_O$
return O

overlaps provide diminishing returns while increasing redundancy.

Dataset	$\lambda = 0.0$	$\lambda = 0.1$	$\lambda = 0.25$	$\lambda = 0.5$
AG News	89% / 0.48	91% / 0.46	**93% / 0.43**	92% / 0.45
MNLI	56% / 0.61	58% / 0.57	**59% / 0.53**	58% / 0.55

Table 10: Overlap ratio ablation for HOBA (block=32, 6 layers). Accuracy / loss shown.

A.6 Accounting for Overlap and Cross-Block Costs

This section provides the full derivation of the additional computational terms introduced by overlapping blocks (Eq. 4) and cross-block attention (Eqs. 5–7), complementing the complexity discussion in Section 3.2.

(a) Overlap cost (referencing Eq. 4). From Eq. 4, the final token output is obtained by averaging the block-specific outputs $o_i^{(m)}$ over the set of overlapping blocks $\mathcal{B}_i$ that contain token x_i:

$$o_i = \frac{1}{|\mathcal{B}_i|} \sum_{m \in \mathcal{B}_i} o_i^{(m)}.$$

Since each token appears in a constant number of overlapping blocks determined by the fixed overlap ratio λ, this introduces only a constant-factor overhead and therefore does not change the asymptotic complexity of the model.

(b) Cross-block attention cost (referencing Eqs. 5–7). As defined in Eqs. 5–7, each block summary representation $r^{(m)}$ is computed by pooling over its b token-level outputs:

$$r^{(m)} = \frac{1}{b} \sum_{i=mb}^{(m+1)b-1} h_i,$$

where h_i denotes the intermediate contextual output for token x_i. Cross-block attention is then performed across all $B = L/b$ blocks. The attention operation over these B block representatives requires

$$\mathcal{O}(B^2 d) = \mathcal{O}\left(\frac{L^2}{b^2} d\right),$$

which captures the full cost of global block-to-block interactions.

(c) Final combined complexity. Combining the local 3D block cost $\mathcal{O}(Lb^2d)$ with the overlap (constant factor) and the cross-block attention term above, the overall complexity of HOBA is

$$\text{HOBA}(L) = \mathcal{O}(Lb^2 d) + \mathcal{O}\left(\frac{L^2}{b^2} d\right),$$

which correctly includes all contributions from local higher-order interactions, overlap aggregation, and cross-block attention.

A.7 Key Difference with SOTA Methods

We find that all these prior higher-order attention models (Table 11) operate globally (spectral kernels, full-sequence graphs, dense tensor contractions), resulting in fundamentally different computational objectives and scaling regimes.

Standard Attention and Knowledge Distillation. Transformers like BERT, RoBERTa, and T5 have become foundational in core NLP tasks like sentiment analysis (Nuci et al., 2024), natural language inference (Raparthy et al., 2023), and topic classification (Kosar et al., 2023). Each task requires different modeling strengths; sentiment analysis needs polarity sensitivity (Nuci et al., 2024), NLI demands contextual reasoning (Raparthy et al., 2023), and news classification favors topical abstraction. To address this diversity, researchers have proposed task-aware attention mechanisms (Benarab and Gui, 2022; Roy et al., 2021), modular tuning methods like adapters (Pfeiffer et al., 2020), and data-centric augmentation techniques (Chai et al., 2024). However, challenges persist in domain generalization, compositional reasoning, and inference efficiency (Patil and Jadon, 2025). Knowledge distillation processes have been widely adopted to mitigate these challenges. These methods compress large transformer models by training compact students on softened outputs from teacher models using temperature scaling and Kullback-Leibler(KL) divergence (Lu et al., 2022). In practice, teacher-student setups align final logits or internal attention patterns through KL or MSE losses, often blending hard and soft supervision (Aguilar et al., 2020; Chen et al., 2021). Recent methods use progressive stages to gradually shrink the model (Yao et al., 2024), bias-aware filtering (Zhang et al., 2025), or cooperative schemes (Livanos et al., 2024). Specialized distillation has also been explored in sentiment and emotion-aware models (Li et al., 2025; Ma et al., 2023).

Method	Interaction	How HO is Modeled	Comp. Structure	Limitation	How HOBA Differs
Jump Attention (JAT) (Zhou et al., 2022)	Spectral interactions	FFT-based spectral convs for multi-token relations	Global conv kernels	No explicit triplets; heavy compute	HOBA computes *explicit* $((i, j, k))$ triplets but only inside blocks
WIGRAPH (Sekhon et al., 2023)	Graph-based HO	Graph propagation for global relations	Global graph layer	Needs full-sequence graph; not scalable	HOBA uses local 3D blocks + lightweight cross-block routing
Graph-Induced Attn (Hong et al., 2022)	Graph-biased dot-prod	Injects syntactic graph into 2D attention	Standard attention + graph bias	Still pairwise; not 3D	HOBA directly models triplets; no external graph needed
Kernelized HO Attn (Choromanski et al., 2020)	Kernel approx.	Tensor kernels for HO interactions	Sequence-wide tensor ops	High memory, expensive	HOBA uses block-diagonal factorization; linear in L for fixed block size
Sparse HO Self-Attn (Cui et al., 2019)	Sparse structured	Sparse softmax patterns	Standard 2D attention	No triadic modeling	HOBA performs true 3D attention over triplets within blocks

Table 11: Conceptual Comparison of HO Attention Methods and How HOBA Differs.

A.8 Why Triplets Over Pairs?

HOBA differs from standard 2D attention not because it uses a 3D tensor, but because the 3D tensor explicitly models third-order token interactions that cannot be captured by pairwise dot-product attention. In conventional Transformers, each attention score depends only on the relation between two tokens, limiting the model to binary dependencies (e.g., subject→verb or adjective→noun). In contrast, HOBA's triplet attention tensor $A_{ijk} = Q_i^\top (K_j \circ K_k)$ incorporates the joint context formed by a pair of keys (j, k), enabling the model to compute attention scores based on interactions of two context tokens simultaneously. This provides access to richer relational structures that naturally arise in language, such as scope resolution and negation ("not really good"), multi-modifier composition ("very highly unlikely"), and compound noun cohesion ("interest rate policy"), all of which depend on three-way configurations rather than isolated token pairs. Whereas 2D attention must approximate these higher-order effects across multiple layers, HOBA can express them within a single layer, offering a stronger inductive bias for structured multi-token reasoning.

Here, we formalize what is gained by 3D attention and what is inherently lacking in pairwise (2D) attention mechanisms.

Setup. Let an input sequence be $X = [x_1, x_2, x_3]$, where $x_i \in \mathbb{R}^d$. In standard 2D self-attention, contextualized representations are formed by aggregating pairwise interactions,

$$h_i = \sum_j \alpha_{i,j}\, v(x_j), \qquad \alpha_{i,j} = \mathrm{softmax}_j(q(x_i)^\top k(x_j)), \quad (17)$$

so that the computation relies only on pairwise affinities $q(x_i)^\top k(x_j)$.

More generally, any pairwise-attention-based model yields a function of the form

$$f(x_1, x_2, x_3) = F(\{\langle x_i, x_j\rangle\}_{i,j}), \quad (18)$$

for some function F. This formulation characterizes models whose representations are identifiable only from pairwise affinities under fixed-depth attention primitives.

Theorem. There exists a triplet-dependent function that cannot be represented by any model whose computation depends only on pairwise interactions, but can be represented by a single 3D attention layer.

Proof. Define a triplet classification function by

$$y(x_1, x_2, x_3) = \mathrm{sign}(\det[x_1\ x_2\ x_3]). \quad (19)$$

Let $x_1 = e_1$, $x_2 = e_2$, $x_3 = e_3$ be the standard basis in $\mathbb{R}^3$, yielding $y = +1$. Now define $x_3' = -e_3$, yielding $y = -1$. In both cases, all pairwise dot products satisfy

$$\langle x_i, x_j\rangle = \langle x_i, x_j'\rangle \quad \forall i, j, \quad (20)$$

so the pairwise affinity set $\{\langle x_i, x_j\rangle\}$ is identical. Therefore, any pairwise-only model must output the same value for both inputs, contradicting the labels. □

Expressivity of 3D attention. In contrast, HOBA constructs a 3D interaction tensor

$$\mathcal{A}_{i,j,k} \propto q(x_i)^\top g(x_j, x_k), \tag{21}$$

and aggregates contextual information via triplet interactions as in Eq. (2). The determinant admits the decomposition

$$\det[x_1\ x_2\ x_3] = \langle \mathcal{E},\ x_1 \otimes x_2 \otimes x_3 \rangle, \tag{22}$$

where $\mathcal{E}$ is the Levi–Civita tensor. Thus, a single 3D attention head can represent this function exactly.

Implication. While deep 2D Transformers may approximate HO relations indirectly, pairwise attention lacks a primitive for genuine third-order interactions. HOBA introduces this primitive explicitly, enabling stable and efficient modeling of triplet-dependent structures.

A.9 Experimental Setup

We compare vanilla RoBERTa models with our proposed HOBA variants under identical training configurations to ensure fairness. Both models use 3-, 6-, and 12-layer transformer backbones with hidden size 768, 12 attention heads, and a feed-forward dimension of 3072. Specifically, we use a small batch size of 8 and a maximum input sequence length of 64 tokens for 3-layer models, and 128 tokens for 6-layer and 12-layer models. In HOBA, standard 2D multi-head self-attention is replaced by block-diagonal higher-order attention, with block sizes $b \in \{16, 64\}$ and an overlap ratio of 25%. Embeddings and classification heads are initialized from a pretrained RoBERTa-base model to guarantee architectural compatibility. All models are trained with AdamW (learning rate 1×10^{-5}, weight decay 0.01), a linear decay schedule with 10% warmup, dropout rate 0.1, batch size 8, and early stopping after 3 epochs without validation improvement. We train for 5 epochs depending on the task and average results across three random seeds (seed = 42) to report mean and standard deviation. Finally, we report model performance to highlight the effectiveness of HOBA compared to the vanilla baseline (see Table 2). We use NVIDIA A100 GPUs for all training, including Colab A100 instances where applicable. The source code and implementation will be available at `github.com/mafizurrahman/HOBA`.