Late Breaking Results: Hyperdimensional Regression with Fine-Grained and Scalable Confidence-Based Learning

Jiseung Kim[1*], Hyunsei Lee[1*], Tajana Rosing[2], Mohsen Imani[3], and Yeseong Kim[1*]
[1]DGIST, [2]University of California San Diego, [3]University of California Irvine
{js980408, wwhslee, yeseongkim}@dgist.ac.kr, tajana@ucsd.edu, m.imani@uci.edu

***Abstract*—We propose an advanced hyperdimensional computing (HDC) framework for regression tasks, addressing the limitations of existing methods through three key innovations: fine-grained feature encoding, confidence-based inference, and dimension-split boosting for scalable training. By preserving inter-feature relationships and enabling efficient computation on high-dimensional spaces, the framework achieves superior accuracy and efficiency across diverse benchmarks. Our evaluation demonstrates that HBRF achieves significant improvements in prediction quality and computational efficiency as compared to the state-of-the-art HDC-based regression by 31% and 54.8%, respectively.**

***Index Terms*—Hyperdimensional Computing, Regression, FPGA**

I. Introduction

The growing demand for lightweight, efficient machine learning solutions has driven significant interest in Hyperdimensional Computing (HDC), a brain-inspired paradigm leveraging high-dimensional vectors (hypervectors) for data representation and learning [1]. The computational simplicity, inherent parallelism, and robustness to noise positions HDC as a compelling alternative to traditional machine learning methods, particularly in resource-constrained environments. While HDC has demonstrated remarkable success in classification tasks, its application to regression remains underexplored.

Regression, as a problem domain, requires the accurate modeling of continuous relationships between input features and target variables, often across highly nonlinear surfaces. Traditional HDC methods, e.g., the state-of-the-art RegHD [2], adapt classification-inspired techniques to regression. RegHD relies on coarse clustering of inputs into discrete hypervector representations, followed by similarity-based inference. While effective for simple regression surfaces, this approach suffers from two fundamental limitations: (1) it lacks fine-grained encoding capabilities to preserve continuous inter-feature relationships, leading to reduced accuracy for complex, nonlinear functions; (2) the inherent high dimensionality of hypervectors makes training regression models computationally intensive on existing hardware platforms due to limited parallelism and memory bandwidth, and it is exacerbated in the existing HDC-based technique as they rely on multi-pass training; and (3) it does not provide a mechanism to quantify prediction confidence, leaving users without a measure of model reliability in uncertain regions of the data space. These constraints hinder the applicablity of HDC-based regression for practical applications.

In this work, we propose a novel HDC-based regression framework, called HBRF (HDC-based Boosting Regression Framework), that addresses these limitations through three key innovations. First, we introduce a fine-grained encoding scheme that captures inter-feature relationships with precision

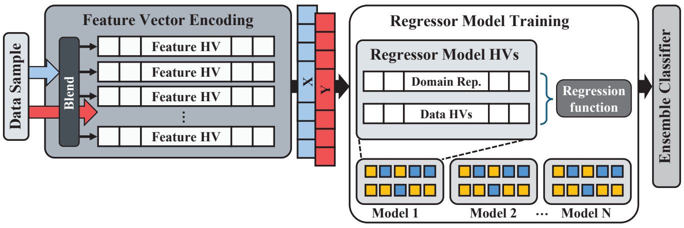

Fig. 1. Overview of HBRF

by dynamically interpolating hypervector representations between normalized bounds. This approach ensures high fidelity in encoding continuous variables, enabling accurate modeling of complex, nonlinear surfaces. Second, we incorporate a confidence estimation mechanism, leveraging similarity metrics to quantify the reliability of predictions. This feature enhances model interpretability and provides crucial insights in scenarios with sparse or noisy data. Finally, we adopt an adaptive boosting strategy tailored to high-dimensional regression. By incrementally constructing hypervectors through single-pass training on subsets of dimensions, our method overcomes the computational challenges of training full-dimensional hypervectors. Our evaluation shows that HBRF delivers substantial enhancements in both prediction accuracy and computational efficiency, outperforming the state-of-the-art HDC-based regression by 31% and 54.8%, respectively.

II. Proposed Methodology

The HBRF framework extends HDC by introducing a novel learning algorithm tailored for continuous regression tasks. Unlike existing methods such as RegHD, which rely on coarse quantization and clustering, our approach leverages fine-grained encoding and a robust training mechanism to model complex nonlinear relationships. Figure 1 illustrates the overall workflow, encompassing three core components: feature vector encoding, regressor model training, and adaptive boosting. The system encodes feature vectors into high-dimensional hypervectors (HVs), constructs regressor models using distributed representations of input-output relationships, and incrementally refines predictions through dimension-split boosting.

In the encoding phase, feature vectors are transformed into hypervectors via a precision-preserving operation termed *blend*, which ensures an accurate representation of inter-feature relationships. The regressor model training involves generating hypervector pairs that aggregate input-output associations across the dataset. Finally, boosting iteratively constructs ensembles of weak regressors, allowing the system to scale to high-dimensional spaces while improving accuracy.

Feature Encoding To encode feature vectors $\mathbf{v} = \langle e_1, \ldots, e_n \rangle$, the proposed method partitions the range of each feature into P segments. For each segment boundary, a random hypervector $\rho^p(\mathbf{B}_i)$ is assigned, where $\mathbf{B}_i$ corresponds to the i^{th} feature. A value e_i residing in segment p is encoded by blending two adjacent boundary hypervectors using the *blend* operation:

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korean government (MSIT) (No.2022-0-00991, 1T-1C DRAM Array Based High-Bandwidth, Ultra-High Efficiency Processing-in-Memory Accelerator). This work was also supported by Cogarobotics.

*These authors contributed equally to this work

978-3-9826741-0-0/DATE25/©2025 EDAA

$$\beta(\rho^p(\mathbf{B}_i), \rho^{p+1}(\mathbf{B}_i), (e_iP - \lfloor e_iP \rfloor) \cdot D),$$

where D is the dimensionality of hypervectors. The *blend* function linearly interpolates between hypervectors, ensuring the encoded hypervector reflects the precise relative position of e_i within its segment. This encoding preserves inter-feature relationships and allows accurate modeling of continuous variable distances in the hyperdimensional space. Subsequently, feature hypervectors are aggregated to form a unified representation using element-wise multiplication to capture feature interactions.

Regressor Model Training Regression training constructs two hypervector models: the domain representation hypervector $\mathbf{M}$ and the data hypervector $\mathbf{W}$. For each training sample i, input features and target values are encoded as hypervectors $\mathbf{X}^i$ and $\mathbf{Y}^i$, respectively. The models are then updated as:

$$\mathbf{M} = \sum_i \mathbf{X}^i \times \mathbf{Y}^i, \quad \mathbf{W} = \sum_i \mathbf{X}^i.$$

Then, the regression output for an input $\overline{\mathbf{X}}$ can be calculated using a similarity-based function:

$$f(\overline{\mathbf{X}}) = \frac{\delta(\mathbf{M}, \overline{\mathbf{X}} \times \mathbf{Y}_{one})}{\delta(\mathbf{W}, \overline{\mathbf{X}})},$$

where $\mathbf{Y}_{one}$ is the hypervector representing $y = 1.0$.

Note that, unlike existing techniques such as RegHD, the regression training can be done in a single pass, which greatly improves efficiency. In addition, in HBRF, the integration of confidence estimation through $\mathbf{W}$ significantly enhances the robustness of the regression results. During inference, the confidence for a query $\overline{\mathbf{X}}$ is determined by measuring its similarity to training data points. Lower similarity indicates sparse representation in the training set, signaling lower prediction confidence. This feature not only provides valuable insights in regions with limited data coverage but also enables more reliable decision-making compared to RegHD, which lacks a mechanism to quantify prediction reliability.

Dimension-Split Boosting The high dimensionality of hypervectors poses computational challenges on existing hardware platforms. To address this, we introduce a boosting strategy that splits the hypervector space into manageable subsets, training weak regressors on these subsets. Each iteration creates a new model, refining predictions based on errors from the previous iteration. The final prediction is the weighted sum of individual models $f_{\text{ensemble}}(\overline{\mathbf{X}}) = \sum_k \alpha_k f_k(\overline{\mathbf{X}})$ where α_k is the weight for the k^{th} model. We can make the full dimension model by concatenating the weak learners.

III. Experimental Results

We implemented HBRF using Python 3.8 for GPU and an RTL implementation for FPGA. The implementation environment included an NVIDIA RTX 3090 GPU for high-performance evaluation and a Xilinx Alveo U280 FPGA for energy-efficient computation. We used a diverse set of regression datasets, including synthetic benchmarks for controlled evaluations and real-world datasets such as Boston Housing, Diabetes, Buzz in social media and Superconductor.

Accuracy Evaluation The proposed framework demonstrates significant improvements in prediction accuracy compared to RegHD across all benchmarks, as illustrated in Figure 2 (left).

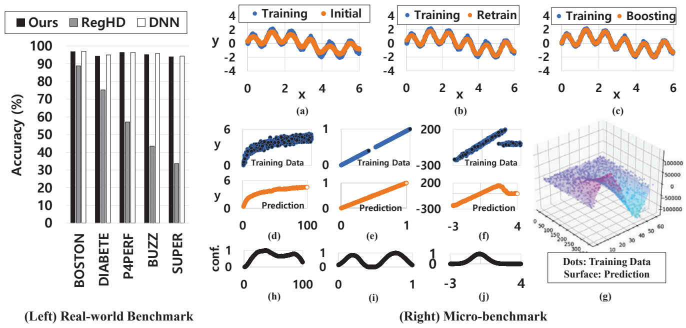

Fig. 2. Accuracy Evaluation of HBRF

TABLE I
FPGA Throughput

	Boston	Diabates	P4PERF	Buzz	Super	Average
Ours (ms)	1.87	1.81	6.43	6.64	6.82	4.71
RegHD (ms)	3.87	3.71	15.24	15.45	15.62	10.78
Speedup (%)	51.68	51.21	57.81	57.02	56.34	54.81

The fine-grained encoding mechanism enables precise modeling of complex relationships, achieving an average accuracy improvement of 31% over RegHD. The microbenchmark evaluations shown in Figure 2 (right) highlight the impact of retraining and boosting on prediction quality. Subfigures (a-c) depict the incremental improvements in prediction accuracy as the model undergoes retraining and dimension-split boosting. Subfigures (d-f) demonstrate the ability of HBRF to handle noisy and incomplete data, while (g) validates its effectiveness in multivariate regression scenarios. Confidence levels, illustrated in subfigures (h-j), show lower confidence in regions with limited training data.

Performance We also evaluate the performance improvement of HBRF as compared to the baseline algorithm, RegHD. Table I summarizes the training throughput of the FPGA implementation to process each batch of 32 samples with the comparison to RegHD. The results show that HBRF achieves a speedup of 54.8% on average. We also observed that for the GPU-based implementation, HBRF results in a speedup of 37% over RegHD. The performance gains are attributed to the proposed dimension split-based boosing technique, which combines weak learners trained only with single-pass processes.

IV. Conclusions

This paper presents a novel regression framework for HDC, addressing critical limitations in existing HDC-based regression methods. The proposed approach introduces fine-grained encoding to preserve inter-feature relationships, dimension-split boosting for scalable training, and a confidence estimation mechanism for reliable inference. It enables precise modeling of complex nonlinear relationships while maintaining efficiency and robustness of HDC. In our evaluation on diverse datasets and hardware platforms, including GPU and FPGA implementations, we demonstrate significant improvements in accuracy (31%) and energy efficiency (54.8%) compared to the state-of-the-art algorithm.

References

[1] P. Kanerva, "Hyperdimensional computing: An introduction to computing in distributed representation with high-dimensional random vectors," *Cognitive computation*, vol. 1, no. 2, pp. 139–159, 2009.

[2] A. Hernández-Cano, C. Zhuo, X. Yin, and M. Imani, "Reghd: Robust and efficient regression in hyper-dimensional learning system," in *2021 58th ACM/IEEE Design Automation Conference (DAC)*, 2021, pp. 7–12.