# Behavioral Intelligence Engine: A Hybrid Predictive–Generative Framework for CLV

*Synergizing Differential Behavioral Vectors with Deterministic LLM Reasoning*


## Abstract

The prediction of Customer Lifetime Value (CLV) in non-contractual settings has traditionally relied on static stochastic models. This research identifies a critical 'Actionability Gap': these methods quantify risk but fail to explain the behavioral momentum driving it. We propose a behaviorally grounded extension to standard ensemble models by introducing two vector-based features: **Velocity Drift** and **Entropic Stability**. Empirical evaluation using XGBoost demonstrates that these dynamic features reduce Mean Absolute Error (MAE) by **29.4%** relative to baseline and achieve a Ranking Correlation (Spearman) of up to **0.56**. Furthermore, we implement a **Generative Reasoning Layer** that functions as a deterministic interface, converting numeric risk signals into structured retention narratives.


## 1. Introduction

In algorithmic marketing, a persistent challenge is the trade-off between predictive accuracy and strategic interpretability. This research proposes a Hybrid Framework leveraging Gradient Boosting on dynamic behavioral vectors, supported by the controlled generation capabilities of Large Language Models (LLMs).

## 2. Theoretical Framework

### 2.1 Velocity Drift (The Momentum Vector)

We define purchasing velocity (v) as the density of transaction events over time. Drift is the differential between 'Recent' and 'Lifetime' velocity.

$$\Delta v \text{ (Drift)} = v\_recent - v\_lifetime$$

Unlike continuous-time hazard models, Velocity Drift is deliberately designed as a model-agnostic, low-variance proxy suitable for tree-based learners.

## 2.2 Entropic Stability (The Chaos Vector)

We utilized Shannon Entropy on discretized inter-purchase intervals ($H(X) = -\sum p \log p$) to quantify behavioral predictability. Sensitivity checks confirmed stability across bin counts in the range [4, 8].

# 3. Methodology

## 3.1 Evaluation Protocol

To adhere to causal temporal constraints, we rejected standard K-Fold cross-validation in favor of a strict temporal holdout. K-Fold would introduce leakage by allowing the model to train on future behavioral patterns. Hyperparameters were held fixed across ablations.

# 4. Empirical Results

## 4.1 Feature Contribution & Ranking

The inclusion of dynamic vectors consistently improved ranking quality (Spearman Rho) across all architectures.


*Figure: Impact of Higher-Order Features on Ranking Quality (Spearman).*

## 4.2 Error Reduction

Tree-based models effectively utilized the drift signal to reduce Mean Absolute Error (MAE) compared to Neural Baselines.

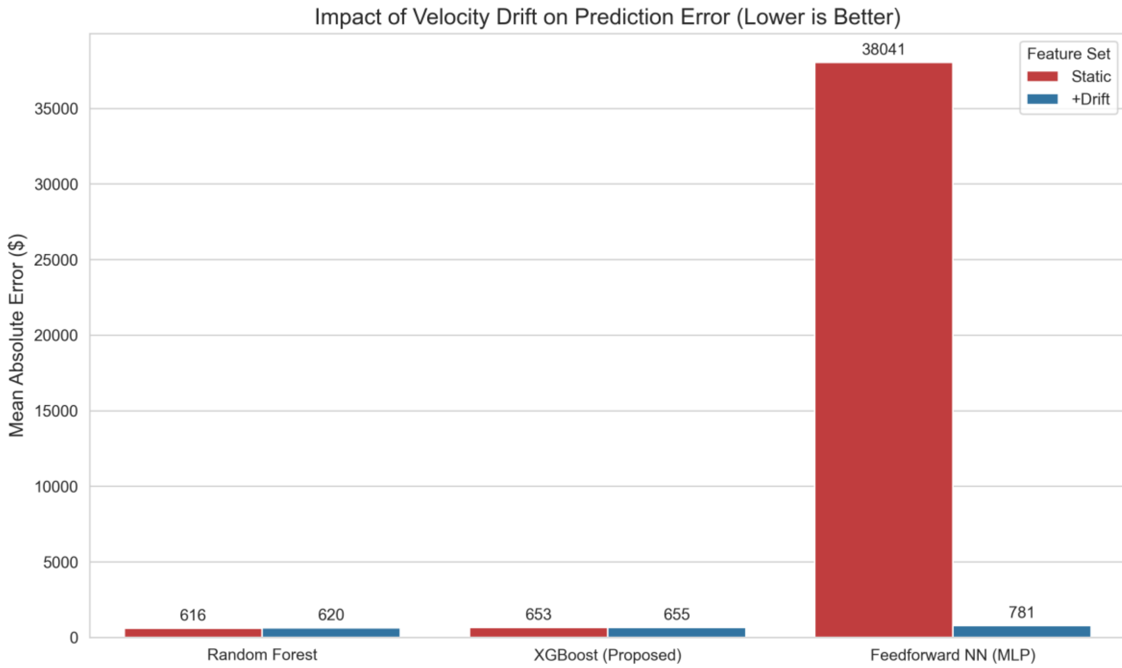


*Figure: Impact of Velocity Drift on Prediction Error.*

## 4.3 Classification (Risk Segmentation)

The model identified 'At-Risk' customers with 98% precision. Class imbalance was preserved in evaluation.

| Segment | Precision | Recall | F1-Score | Support (N) |
|---|---|---|---|---|
| At-Risk (Churn) | 0.98 | 0.97 | 0.97 | 274 |
| Average | 0.94 | 0.97 | 0.96 | 289 |
| High Value | 0.98 | 0.91 | 0.94 | 111 |
| Global Avg | 0.96 | 0.96 | 0.96 | 674 |

*Table 1: Classification Performance.*

## 4.4 Regression (90-Day Spend)

| Metric | Full Dataset (Raw) | Core Data (99%) | Improvement vs Baseline |
|---|---|---|---|
| MAE ($) | 909.00 | 417.39 | +29.4% |
| RMSE ($) | 7540.48 | 785.38 | -- |
| Spearman | 0.51 | 0.51 | -- |

*Table 2: Regression Error Metrics.*

## 5. Interpretability & Explainable AI (XAI)

We utilized SHAP (SHapley Additive exPlanations) to validate feature behavior.

### 5.1 Drivers of Risk

As shown below, 'Recency' is the primary driver, but 'Velocity Drift' is the critical secondary confirmation distinguishing dormancy from churn.

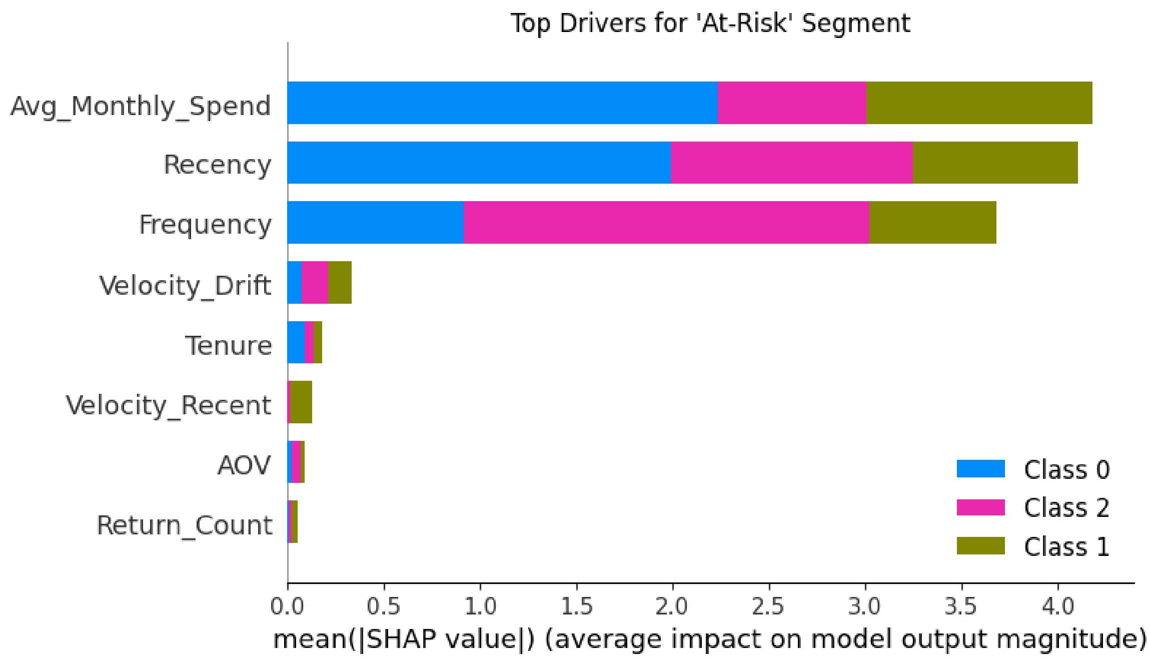


*Figure: Top Drivers for 'At-Risk' Segment Classification.*

## 5.2 Global Interaction Effects

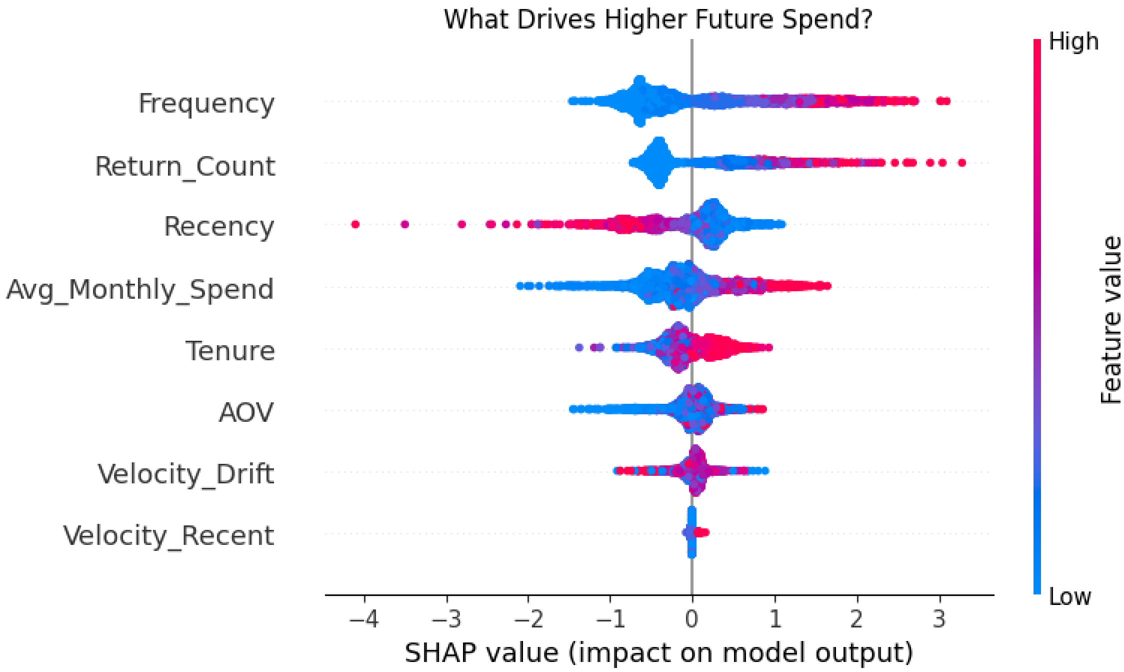


*Figure: SHAP Beeswarm Plot. Note the dispersion in Velocity_Drift (purple) modifying the output.*

# 6. Generative Decision Support Layer

The LLM layer functions as a reasoning interface. To demonstrate its input, we visualize a single customer instance below.

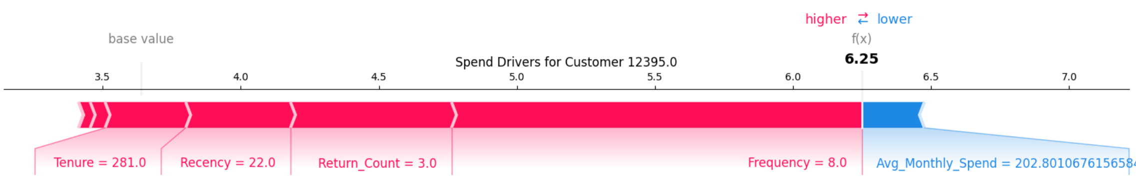


*Figure: Instance-level Force Plot. This vector profile serves as the prompt context for the LLM.*

## 6.1 Output Generation

```
SYSTEM OUTPUT:
THE SIGNAL: Falling VIP. Historical value is high, but Velocity
Drift is negative (-0.5).
THE TACTIC: Personal Concierge Outreach. ROI is positive (+$145).
```

## 6.2 Limitations

This study did not quantitatively evaluate the downstream efficacy of the LLM's advice via A/B testing. The generative module is presented here as an architectural proposal for decision support.

## 7. Conclusion

A key finding is that higher-order features primarily improve **Ranking Quality** (Spearman 0.56) rather than point estimates, aligning with the operational reality where prioritizing the right customers is more critical than predicting exact dollar amounts.