Why Financial Event Sequences Require Temporal Transformers: Necessity Theorems for Default Prediction

Victor Zhorin
vzhorin@uchicago.edu

January 28, 2026

Abstract

Credit risk models typically aggregate transaction histories into tabular features, discarding temporal structure. We prove this approach is fundamentally limited. Three architectural components are mathematically necessary for optimal default prediction from financial event sequences: discrete event embeddings preserving transaction identity, spectral encoding of timestamps, and learned attention over account history. Each component is individually irreducible. Removing any one provably reduces predictive performance.

The central result establishes that standard positional encoding, which assumes uniform event spacing, is categorically incorrect for financial data where transactions carry intrinsic timestamps. A payment at day 15 of the billing cycle differs predictively from the same payment at day 45, independent of sequence position. Ordinal encoding discards this temporal geometry.

We further prove that default outcomes shape the entire account trajectory representation through gradient flow during training, creating second-order structure where early transactions modify interpretation of subsequent events. This formalizes the credit intuition that payment history must be read as a whole.

The theoretical framework identifies minimal requirements for financial sequence modeling and explains observed performance gaps between sequential and tabular approaches in credit scoring.

1 Introduction

Financial institutions maintain transaction-level records: payments, purchases, transfers, credit draws, delinquencies. Each event carries an intrinsic timestamp. Predicting default from these sequences is central to credit risk management [6, 3].

Two modeling paradigms dominate. Tabular approaches aggregate transaction statistics into feature vectors: average payment amount, utilization ratio, days past due. Gradient boosting and logistic regression operate on these aggregates [11]. Sequential approaches treat account histories as token sequences, applying recurrent networks or transformers [2, 10]. Sequential methods increasingly outperform tabular baselines, but theoretical understanding of why remains incomplete.

This paper proves three results explaining the performance gap:

1. Tabular aggregation provably discards predictive information encoded in transaction ordering.
2. Standard positional encoding provably discards predictive information encoded in timestamp geometry.
3. Fixed-weight aggregation provably cannot capture context-dependent transaction interpretation.

These are mathematical necessities, not empirical observations. Any architecture violating them operates below the achievable performance bound.

1.1 Motivating Example

Consider two borrowers with identical summary statistics: same average payment, same utilization, same account age. Borrower A made consistent payments on the 15th of each month for 24 months. Borrower B made erratic payments clustering around due dates, with three instances of 45-day gaps followed by catch-up payments.

Tabular models see identical feature vectors. They cannot distinguish these borrowers. Sequential models with ordinal positional encoding see similar sequences (24 payment events at positions 1 through 24). The temporal geometry differs, but ordinal encoding discards it.

Only sequential models encoding actual timestamps capture the behavioral difference. We prove this intuition is mathematically precise.

1.2 Contributions

Necessity Theorems. We prove three architectural requirements for optimal default prediction from financial event sequences (Section 3).

Positional Encoding Critique. Standard transformer positional encoding is provably suboptimal for timestamped financial data (Corollary 3.8).

Boundary Determination. Default outcomes shape learned attention weights throughout account history via gradient flow (Section 4). Early transactions modify interpretation of later transactions through training dynamics.

Methodological Guidance. The theorems constrain model selection and explain when sequential approaches provide genuine advantage over tabular baselines.

2 Problem Formulation

2.1 Financial Event Sequences

Definition 2.1 (Account Trajectory)**.** *An* account trajectory *is a tuple* $\mathcal{A} = (E, \prec, \tau, \kappa)$ *where:*

- $E = \{e_1, \ldots, e_n\}$ *is a finite set of financial events*
- $\prec$ *is temporal precedence (partial order)*
- $\tau : E \to \mathbb{R}^+$ *maps events to timestamps*
- $\kappa : E \to \mathcal{C}$ *assigns event types from vocabulary* $\mathcal{C}$

Financial events include payments, purchases, cash advances, balance transfers, credit limit changes, and delinquency flags. Each carries an intrinsic timestamp from transaction processing.

Definition 2.2 (Observation Window)**.** *A* prediction horizon T *defines observable history* $J^-(T) = \{e : \tau(e) < T\}$*. The* outcome $Y \in \{0, 1\}$ *indicates default within a specified period after* T*.*

2.2 Prediction Task

Given account trajectory $\mathcal{A}$ observed up to horizon T, estimate $P(Y = 1|\mathcal{A}|_T)$.

Definition 2.3 (Predictor)**.** *A* predictor *is a measurable function* $\mu : \mathcal{A}|_T \to [0, 1]$*.*

Goal: characterize optimal predictors minimizing expected loss over account distribution.

3 Necessity Theorems

We prove three architectural components are individually necessary.

3.1 Information Bound

Axiom 1 (Finite Information per Event)**.** *Predictive mutual information satisfies:*

$$I(\mu(\mathcal{A}); Y) \leq C \cdot |J^-(T)| \tag{1}$$

where C bounds bits per event and $|J^-(T)|$ is event count.

Information scales with transaction count, not calendar time. An account with 200 transactions over 6 months contains more signal than an account with 10 transactions over 5 years.

3.2 Necessity of Discrete Event Representation

Tabular methods aggregate events into summary statistics: payment counts, average amounts, utilization ratios. We prove this discards predictive information.

Theorem 3.1 (Transaction Order Matters)**.** *Let f depend only on marginal distributions of event types and timestamps. There exist trajectories $\mathcal{A}_1, \mathcal{A}_2$ with identical marginals but different optimal predictions.*

Proof. Construct:

$$\mathcal{A}_1 = \{(\text{large\_purchase}, t_1) \prec (\text{full\_payment}, t_2)\} \tag{2}$$

$$\mathcal{A}_2 = \{(\text{full\_payment}, t_1) \prec (\text{large\_purchase}, t_2)\} \tag{3}$$

Both have event set {large_purchase, full_payment} and timestamp set $\{t_1, t_2\}$. Marginals are identical.

But purchase-then-payment indicates responsible credit use: buy, then pay. Payment-then-purchase after paying down indicates potential cash flow stress or changed spending behavior.

If $P(Y = 1|\text{purchase} \prec \text{payment}) \neq P(Y = 1|\text{payment} \prec \text{purchase})$, optimal predictions differ, but marginal-based methods cannot distinguish. □

Corollary 3.2. *Tabular feature engineering (transaction counts, average amounts) discards predictive information in temporal ordering.*

This explains performance gaps between XGBoost on aggregated features and sequential models [2, 3].

3.3 Necessity of Spectral Temporal Encoding

Standard transformers encode sequence position, not timestamp. Token k receives $\text{PE}(k)$ regardless of when the transaction occurred [9]. We prove this is wrong for financial data.

Definition 3.3 (Ordinal Encoding)**.** *Encoding $\phi(e_k) = PE(k)$ depends only on sequence position.*

Theorem 3.4 (Timestamp Geometry Matters)**.** *Let g use ordinal encoding. There exist trajectories with identical event sequences but different temporal geometry requiring different predictions.*

Proof. Construct:

$$\mathcal{A}_1 = \{(\text{payment}, \text{day } 15), (\text{payment}, \text{day } 45), (\text{payment}, \text{day } 75)\} \tag{4}$$

$$\mathcal{A}_2 = \{(\text{payment}, \text{day } 15), (\text{payment}, \text{day } 16), (\text{payment}, \text{day } 90)\} \tag{5}$$

Both have event sequence (payment, payment, payment) at positions $(1, 2, 3)$. Ordinal encoding cannot distinguish them.

But $\mathcal{A}_1$ shows regular 30-day payment discipline. $\mathcal{A}_2$ shows two rapid payments (possible overpayment correction or cash windfall) then long gap. These behavioral patterns differ [4].

If default risk differs between regular and irregular timing, optimal predictions differ, but ordinal encoding fails. □

Definition 3.5 (Spectral Temporal Encoding)**.**

$$\Phi_\theta(t) = (\omega_0 t + \phi_0, \sin(\omega_1 t + \phi_1), \ldots, \sin(\omega_{d-1} t + \phi_{d-1})) \tag{6}$$

with learnable frequencies $\{\omega_k\}$ *and phases* $\{\phi_k\}$*.*

Theorem 3.6 (Universal Approximation)**.** *Spectral encoding with sufficient frequencies can approximate any continuous function of time.*

Proof. Stone-Weierstrass: $\{1, t, \sin(\omega t + \phi)\}$ generates a dense subalgebra of $C([0, T])$. □

Remark 3.7. *Learned frequencies discover financially meaningful timescales: billing cycles (30 days), pay periods (biweekly, monthly), quarterly patterns, and annual seasonality.*

Corollary 3.8 (Positional Encoding is Suboptimal)**.** *For financial event sequences, ordinal positional encoding provably limits achievable performance.*

3.4 Necessity of Learned Attention

Simple aggregation assigns fixed weights to event types. We prove context-dependent weighting is necessary.

Theorem 3.9 (Context Determines Meaning)**.** *Fixed-weight aggregation cannot capture context-dependent transaction interpretation required for optimal prediction.*

Proof. Let transaction c (e.g., minimum payment) appear in two contexts:

$$\mathcal{A}_A = \{(\text{overlimit}, t_1) \prec (c, t_2)\} \tag{7}$$

$$\mathcal{A}_B = \{(\text{large\_payment}, t_1) \prec (c, t_2)\} \tag{8}$$

Fixed aggregation assigns weight $w(c)$ regardless of context.

But minimum payment following overlimit signals distress: borrower cannot pay more. Minimum payment following large payment may indicate temporary cash allocation choice. Same transaction, different meaning [1].

If $P(Y = 1|\text{overlimit} \prec c) \neq P(Y = 1|\text{large\_payment} \prec c)$, optimal prediction requires context-dependent weights.

Learned attention provides this:

$$\alpha(e_i, e_j) = \text{softmax}\left(\frac{\langle Q\psi(e_i), K\psi(e_j)\rangle}{\sqrt{d}}\right) \tag{9}$$

□

3.5 Minimal Architecture

Theorem 3.10 (Necessary Components)**.** *Optimal default prediction from financial event sequences requires:*

- *Discrete event embeddings (Theorem 3.1)*
- *Spectral temporal encoding (Theorem 3.4)*
- *Learned attention (Theorem 3.9)*

Architectures missing any component are provably suboptimal.

4 Gradient Flow from Default Outcomes

Beyond architecture, we prove that default outcomes shape how models interpret entire account histories.

4.1 Path Decomposition

Theorem 4.1 (Gradient Paths)**.** *The gradient of prediction loss with respect to event embedding $\psi(e_j)$ decomposes over paths through the attention graph:*

$$\frac{\partial \mathcal{L}}{\partial \psi(e_j)} = \sum_{\gamma \in \Gamma(e_j, Y)} w_\gamma \cdot v_\gamma \tag{10}$$

The default signal propagates backward through learned attention structure.

4.2 Second-Order Effects

Theorem 4.2 (Transaction Interactions)**.** *For temporally related events $e_i \prec e_j$:*

$$\frac{\partial^2 \mathcal{L}}{\partial \psi(e_i) \partial \psi(e_j)} \neq 0 \tag{11}$$

generically. Early transactions modify the predictive contribution of later transactions.

This formalizes credit intuition. A missed payment changes how subsequent minimum payments are interpreted. The effect is multiplicative through attention, not additive through aggregation.

4.3 Boundary Determination

Theorem 4.3 (Default Shapes Representation)**.** *At training convergence, attention weights satisfy:*

$$\alpha^*(e_i, e_j) \propto \frac{\partial}{\partial \psi(e_j)} \log P(Y|\mathcal{A}) \tag{12}$$

Accounts that default train the model to attend differently than accounts that remain current. The learned attention encodes default-predictive relationships between transaction types.

5 Methodological Implications

5.1 When Sequential Models Help

The theorems identify conditions where sequential approaches provide genuine advantage:

Transaction ordering carries signal. If the sequence of payments, purchases, and credit events affects default risk, tabular aggregation loses information.

Timing geometry matters. If payment regularity, transaction clustering, or cycle-relative timing predicts outcomes, ordinal encoding loses information.

Context modifies interpretation. If the same transaction type means different things depending on recent history, fixed aggregation loses information.

When none hold, tabular methods may suffice with lower computational cost.

5.2 Regulatory Considerations

Credit models face interpretability requirements [8]. The necessity theorems clarify the tradeoff:

- Tabular models are interpretable but provably limited.
- Sequential models capture more signal but require attention visualization or other explanation methods.
- The performance gap is not a modeling choice but a mathematical constraint.

Regulators evaluating model risk should recognize that simpler models may underperform not due to poor engineering but due to architectural limitations.

5.3 Feature Engineering Guidance

Effective features should preserve:

- Transaction ordering (not just counts)
- Inter-event timing (not just averages)
- Context windows (not just marginal frequencies)

Hand-crafted sequence features can partially close the gap with learned representations.

6 Related Work

Credit Scoring. Traditional scorecards use logistic regression on aggregated features [6]. Gradient boosting improves performance but maintains tabular structure [11]. Our results explain why sequential approaches outperform these baselines.

Deep Learning for Credit. RNNs for transaction sequences showed early promise [2]. Transformer-based credit models have emerged recently [10, 7]. These architectures use standard positional encoding; our results suggest timestamp encoding would improve performance.

Behavioral Finance. Payment timing reflects household financial constraints [1, 4]. Our theorems formalize why this behavioral signal requires temporal encoding to extract.

Time Representation. Time2Vec introduced learnable periodic representations [5]. Applications to finance remain limited. Our contribution: proving spectral encoding is necessary for timestamped financial sequences.

7 Limitations

The theorems establish *what* is required, not sample complexity or computational cost. Practical deployment involves additional constraints.

The framework assumes discrete timestamped events with binary outcomes. It does not directly address:

- Continuous account balances (time series rather than events)
- Multi-class outcomes (e.g., delinquency stages)
- Portfolio-level correlations

The proofs are existential: architectures missing components *can* fail, not that they *always* fail on every dataset.

8 Conclusion

Default prediction from financial event sequences requires discrete event embeddings, spectral temporal encoding, and learned attention. These components are mathematically necessary. Tabular aggregation, ordinal positional encoding, and fixed-weight pooling each provably limit performance.

The results explain observed gaps between sequential and tabular credit models and provide methodological guidance: encode actual timestamps, preserve transaction order, allow context-dependent interpretation.

For credit risk practitioners, the theorems identify when sophisticated sequential models justify their complexity and when simpler approaches suffice.

References

[1] S. Agarwal, C. Liu, and N. S. Souleles. The reaction of consumer spending and debt to tax rebates: Evidence from consumer credit data. *Journal of Political Economy*, 115(6):986–1019, 2007.

[2] D. Babaev, M. Savchenko, A. Tuzhilin, and D. Umerenkov. E.T.-RNN: Applying deep learning to credit loan applications. In *KDD*, 2019.

[3] B. R. Gunnarsson, S. vanden Broucke, B. Baesens, M. Óskarsdóttir, and W. Lemahieu. Deep learning for credit scoring: Do or don't? *European Journal of Operational Research*, 295(1):292–305, 2021.

[4] A. E. Khandani, A. J. Kim, and A. W. Lo. Consumer credit-risk models via machine-learning algorithms. *Journal of Banking & Finance*, 34(11):2767–2787, 2010.

[5] S. M. Kazemi, R. Goel, S. Eghbali, J. Ramber, J. Sahota, and others. Time2Vec: Learning a general time representation. *arXiv:1907.05321*, 2019.

[6] S. Lessmann, B. Baesens, H.-V. Seow, and L. C. Thomas. Benchmarking state-of-the-art classification algorithms for credit scoring: An update of research. *European Journal of Operational Research*, 247(1):124–136, 2015.

[7] I. Padhi, Y. Schiff, I. Melnyk, M. Rigotti, and others. Tabular transformers for modeling multivariate time series. In *ICASSP*, 2021.

[8] Board of Governors of the Federal Reserve System. SR 11-7: Guidance on model risk management. 2011 (revised 2020).

[9] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, and others. Attention is all you need. In *NeurIPS*, 2017.

[10] Z. Wang, J. Wohlwend, and T. Lei. Structured prediction for conditional meta-learning. In *NeurIPS*, 2021.

[11] Y. Xia, C. Liu, Y. Li, and N. Liu. A boosted decision tree approach using Bayesian hyper-parameter optimization for credit scoring. *Expert Systems with Applications*, 78:225–241, 2017.