From Text to Forecasts

Bridging Modality Gap with Temporal Evolution Semantic Space (TESS)

A Simple Question to Start With

Imagine you're a stock trader. This morning, two inputs land on your desk:

A 2-page news article *"Tensions escalated overnight as the central bank hinted at an unexpected rate hike. Analysts warn of cascading effects across tech-sector equities, though opinions diverge on magnitude and timing. . . "*

≈ 500 words

One line from a senior analyst

"Mean ↓, volatility ↑↑, effect: next 2 days, then fades."

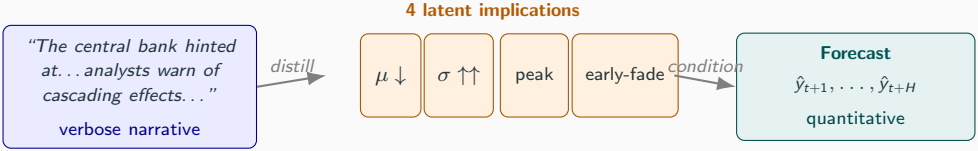
≈ 12 words

Which one helps you forecast better?

The article has *more* information. The one-liner has *more usable* information.

This is Exactly What TESS Does

The trader's secret: she never reads the prose as prose — she silently extracts four statistical implications, and *that* is what drives her forecast.



Three reasons why this works — and each has a theorem behind it:

Nothing lost Those 4 implications *are* the predictive content.
← **T4.1**

Robust to mistakes If you're unsure about one, down-weight it.
← **TA.5**

Learns from few 4 discrete axes $\ll$ vocabulary of all prose.
← **TA.6**

★ TESS formalises the *trader's mental shortcut* — and proves it's the right one.

Motivation

Why fuse Text into Time-Series Forecasting?

Unimodal forecasting captures temporal dependencies well under *stationarity*.

But real-world series are **event-driven & non-stationary**:

- Accidents, extreme weather, public-sentiment shocks $\Rightarrow$ *regime shifts*
- Trends, mean, variance change abruptly within short windows
- Numerical history alone cannot foresee these breaks

Textual exogenous signals (news, announcements, social media) carry the *cause* of these shifts $\rightarrow$ promising to fuse with numerical series.

The Core Obstacle: the Modality Gap

Time Series	**Text**
Chronologically ordered	Weakly temporal
Quantitative, compact	Qualitative, diffuse
Explicit measurements	Implicit semantics
"+15.3%"	*"significant rise"*

Consequence (largely overlooked)
Predictive relevance in text is **sparsely distributed across tokens** and **rarely aligned** with the compact numerical structure forecasters need.

Diagnosis: What actually goes wrong?

Semi-Synthetic Benchmark Design

Goal: control the ground-truth predictive signal inside text.

Construction (on FNSPID):

1. Extract statistical features from the *future* window (mean shift, volatility, ...)
2. Prompt GPT to render them into natural language aligned to the sample
3. Obtain token-level labels:

$$\mathcal{T}_{\text{sig}} \text{ (signal)} \quad \text{vs.} \quad \mathcal{T}_{\text{red}} \text{ (redundant)} \qquad |\mathcal{T}_{\text{sig}}| \ll |\mathcal{T}_{\text{red}}|$$

Two controlled questions:

(Q1) Can fusion *localize* $\mathcal{T}_{\text{sig}}$?

(Q2) Once redundancy is removed, can signals be *translated* into gains?

Bottleneck #1 — Attention is Distracted

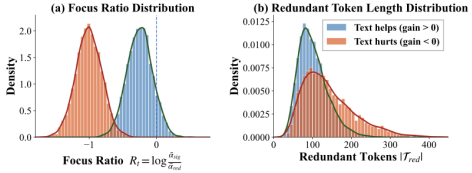


Figure 2: Analysis of attention misalignment. ***Left***: Distribution of focus ratio R_t on test samples. ***Right***: Relationship between redundant token count and predictive performance.

Figure 2 of paper

Focus Ratio:

$$R_t = \log \frac{\bar{\alpha}_{\text{sig}}}{\bar{\alpha}_{\text{red}}}$$

$R_t > 0$: attention on signal ✓

$R_t < 0$: attention on noise ×

Finding: Even when text *helps*, most samples show $R_t < 0$.

- Attention systematically over-attends to redundant tokens
- More redundancy ⇒ more harm

Bottleneck #2 — Representational Mismatch

Three input variants:

- **Full**: all text
- **Signal-Only**: keep $\mathcal{T}_{\text{sig}}$ only
- **Numerical**: raw statistical features

Result (MSE): Numerical $<$ Signal-Only $<$ Full

Key takeaway
Even after *completely* removing redundancy, textual signals cannot match their numerical counterparts.

$\Rightarrow$ The gap is not just noise—it is **representational**.

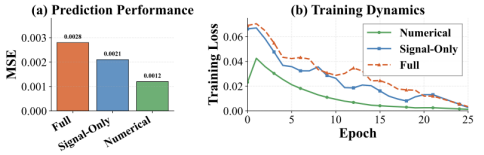


Figure 3: Comparison of three input variants. ***Left***: Prediction performance (MSE) across Full, Signal-Only, and Numerical inputs. ***Right***: Training loss curves showing convergence dynamics.

Figure 3 of paper

Intuition: Why an Intermediate Space?

From Diagnosis to Design Intuition

The two bottlenecks point to a single conclusion:

- Direct text→numeric fusion is *under-constrained*.
- The fusion layer must reason with qualitative verbs ("rise", "surge") while outputting quantitative values.

Human expert analogy: A domain expert reading a news article does *not* memorise prose — she extracts a small set of **temporal implications**:

> *"Will the mean rise or fall? Will volatility spike? Over what horizon? With what decay?"*

Design principle: build an **information bottleneck** between modalities that is (i) *numerically verifiable*, (ii) *interpretable*, and (iii) *expressible in text*.

TESS at a Glance

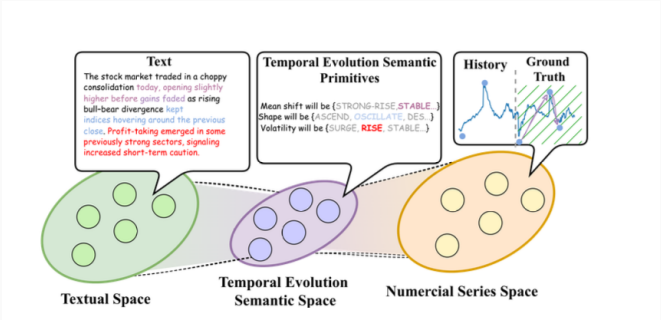
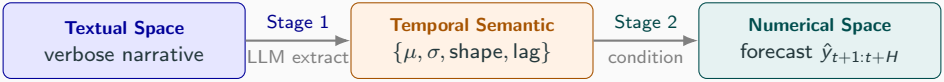


Figure 1 — Text space → Temporal Evolution Semantic Space → Numerical space



★ **Key idea:** an interpretable, numerically verifiable *bottleneck* between modalities → filter noise, preserve predictive info, tighten generalisation.

Method

Architecture

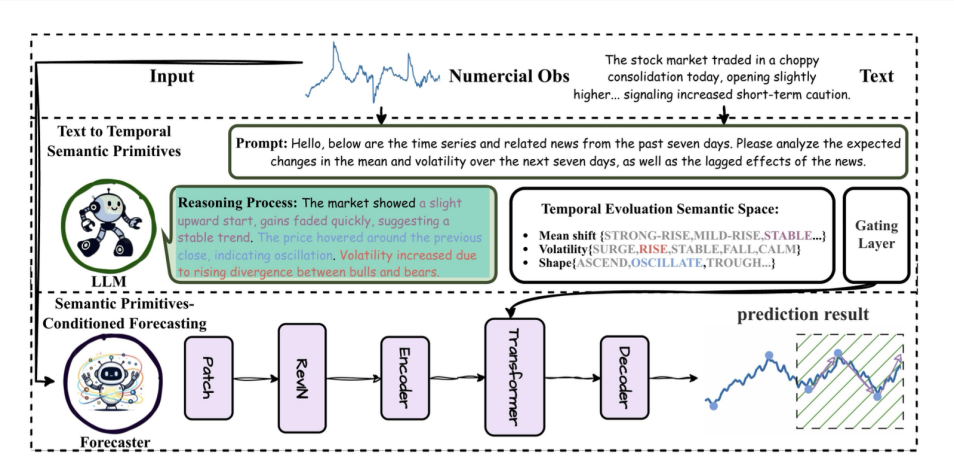


Figure 4 — structured prompting + gating + PatchTST prefix fusion

Temporal Semantic Primitives (TSPs)

Four primitives $\mathcal{P} = \{p_\mu,\ p_\sigma,\ p_{\text{shape}},\ p_{\text{lag}}\}$, each over a *finite discrete vocabulary* $\mathcal{V}_k$.

(1) Distribution Shift — what changes

Mean shift

$$\Delta\mu = \frac{\bar{Y}_t - \bar{X}_t}{\sigma(X_t)}$$

Log-volatility ratio

$$r_\sigma = \log \frac{\sigma_Y + \epsilon}{\sigma_X + \epsilon}$$

5 levels: `strong-rise` | `mild-rise` | `stable` | `mild-drop` | `strong-drop`

TSPs — Shape and Lag

(2) Shape p_{shape} — how it evolves Partition forecast into N_{fcst} patches; inter-patch trend signs:

$$s_i = \operatorname{sgn}_\tau(\bar{u}_{i+1} - \bar{u}_i)$$

`ascend` | `descend` | `peak` | `trough` | `oscillate`

(3) Lag & Decay p_{lag} — when & how long Influence distribution $\pi_i \propto a_i$ over forecast patches, with three indicators:

Centroid c	**Tail mass** d	**Prominence** q
timing of onset	persistence	concentration

`early-fade` | `early-persist` | `mid-fade` | `mid-persist` | `late` | `diffuse`

★ **Key property: numerical verifiability** ★

each primitive has a deterministic extractor ψ_k on (X_t, Y_t)

Stage 1 — Text → Primitives

LLM as a frozen classifier over $\mathcal{V}_k$

Log-likelihood scoring

$$\ell_{t,k}(v) = \log P_{\text{LLM}}(v \mid s_t, \mathcal{D}_k)$$

Temperature softmax

$$q_{t,k}(v) = \text{softmax}_\tau\big(\ell_{t,k}\big)$$

$$\hat{v}_{t,k} = \arg\max_{v \in \mathcal{V}_k} \; q_{t,k}(v)$$

Why this works

- **Verbal specification.** Prompt $\mathcal{D}_k$ defines each label in natural language — the LLM's native domain.
- **Expert-style judgement.** LLM reasons *inside* the primitive space, not from scratch.
- **Modular.** The forecaster is relieved from re-discovering event semantics end-to-end.

Stage 1 — Confidence-Aware Gating

LLM labels are **not always reliable** — ambiguous news, insufficient info, hallucination.
Without filtering, bad primitives propagate directly into the forecast.

Uncertainty signal — top-1 / top-2 margin

$$m_{t,k} = \log q_{t,k}(v^{(1)}) - \log q_{t,k}(v^{(2)})$$

large margin $\Rightarrow$ confident | small margin $\Rightarrow$ ambiguous

Gating network

Embed & fuse

$$h_{t,k} = E_k[\hat{v}_{t,k}]$$

$$g_{t,k} = \sigma\big(w_k^\top [h_{t,k};\ W_m m_{t,k}] + b_k\big)$$

Free supervision (via ψ_k)

$$y_{t,k} = \mathbb{1}[\hat{v}_{t,k} = \psi_k(Y_t)]$$

$$\mathcal{L}_{\text{gate}} = \mathrm{BCE}(g_{t,k}, y_{t,k})$$

Stage 2 — Primitive-Conditioned Forecasting

Backbone: PatchTST with instance normalisation.

Prefix fusion — primitives enter inside self-attention

$$Z^{(0)} = \big[\underbrace{P}_{K\times d}\ ;\ \underbrace{E_{\text{patch}}}_{N\times d}\big] \in \mathbb{R}^{(K+N)\times d}$$

every patch attends to every primitive → semantic signals reach *every layer*, not just concatenated once.

Joint objective

$$\mathcal{L} = \underbrace{\tfrac{1}{H}\|\hat{y} - y\|_2^2}_{\mathcal{L}_{\text{fcst}}} + \lambda\ \underbrace{\mathcal{L}_{\text{gate}}}_{\text{confidence}}$$

end-to-end training (LLM frozen) — forecaster and gate learn together.

★ **Three-way decomposition:** LLM reasons | gate filters | Transformer forecasts

Theory: Why the bottleneck is principled

The Theoretical Puzzle

TESS replaces rich token embeddings with **only $K=4$ discrete labels**.

Three uncomfortable questions immediately arise:

Q1 Information loss?
"A hand-crafted bottleneck must discard predictive information."

Q2 LLM errors?
"If the LLM mislabels a primitive, the forecaster is misled."

Q3 Why should so few labels generalise better than rich features?

Answer: three theorems, one for each question
T4.1 (Sufficiency) **TA.5** (Error attenuation) **TA.6** (Sample complexity)

Each theorem nails down *one* concern; together they justify the entire design.

The Key Assumption — Semantic Sufficiency

Assumption A.1 (Semantic Sufficiency)

$$\hat{Y}_t \perp\!\!\!\perp X_{\text{text}} \,\big|\, (P_t, X_{\text{time}})$$

Once you know the primitives P_t and the numerical history X_{time}, the raw text X_{text} carries no further predictive information about $\hat{Y}_t$.

Is this realistic?

- The four primitives span the *statistical properties* a forecast depends on: mean level, volatility, morphology, timing.
- Any residual text content (sentiment tone, named entities, phrasing) is *epiphenomenal* w.r.t. the forecast once those four are fixed.
- Empirically validated by Fig. 3: *Numerical* input $\leq$ *Signal-Only* input.

Under A.1, the primitives are a sufficient statistic for the forecast. This is the linchpin everything else stands on.

Theorem 4.1 — Lossless Bottleneck + Tighter Gap

Setup. Any text encoder $f : X_{\text{text}} \mapsto Z$. We construct a *primitive-only* encoder $\tilde{f} : P_t \mapsto \tilde{Z}$.

Claim 1 — Predictive information is preserved

$$I(\tilde{Z}; \hat{Y}_t \mid X_{\text{time}}) \;=\; I(Z; \hat{Y}_t \mid X_{\text{time}})$$

The bottleneck keeps *exactly* the information useful for forecasting.

Claim 2 — Token dependence is reduced

$$I(\tilde{Z}; X_{\text{text}}) \;\leq\; I(Z; X_{\text{text}})$$

The bottleneck discards everything that's merely correlated with tokens.

Claim 3 — Generalisation gap shrinks
Under sub-Gaussian loss (A.2) + iid (A.3):

$$\text{Gen}(\tilde{Z}) \;\leq\; \text{Gen}(Z) \qquad \text{where} \qquad \text{Gen}(Z) \;\leq\; \sqrt{\tfrac{2\sigma^2}{n}\, I(Z; X_{\text{text}})}$$

Theorem 4.1 — Proof Strategy (Constructive)

Step 1. Construct the primitive encoder. Group text samples by primitive value $v = \phi(x)$ and average:

$$\tilde{p}(z \mid x) := p(z \mid v, s) = \sum_{x' : \phi(x') = v} \alpha_{x' \mid v, s}\, p(z \mid x')$$

Step 2. Predictive joint unchanged. Using $p(y|x, s) = p(y|v, s)$ (A.1):

$$p_{\tilde{p}}(z, y, s) = \sum_{v} p(z|v, s)\, p(y|v, s)\, p(v, s) = p(z, y, s) \quad \Rightarrow \quad I(\tilde{Z}; Y|S) = I(Z; Y|S)$$

Step 3. Token MI decreases by KL convexity. $\tilde{p}(z \mid v, s)$ is a convex combination of $\{p(z \mid x)\}_{\phi(x)=v}$:

$$D_{\mathrm{KL}}\Big(\textstyle\sum_{x'} \alpha_{x'} p(\cdot|x') \,\Big\|\, p(\cdot) \Big) \le \textstyle\sum_{x'} \alpha_{x'}\, D_{\mathrm{KL}}\big(p(\cdot|x') \,\|\, p(\cdot)\big)$$

Summing over (v, s) gives $I(\tilde{Z}; X_{\text{text}}) \le I(Z; X_{\text{text}})$. ■

Geometric Picture of Theorem 4.1

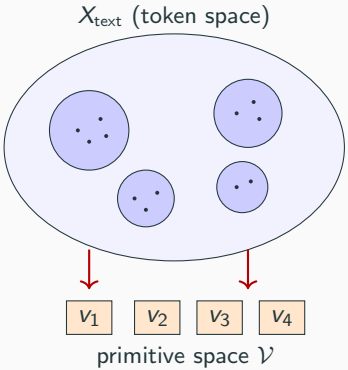
Think of it as a **quotient space**:

- Partition text space by primitive value v:

$$X_{\text{text}} = \bigsqcup_{v} \{x : \phi(x) = v\}$$

- Inside each cell, texts are predictively *equivalent* (thanks to A.1).
- The bottleneck **collapses each cell to a single point**.

What is lost? Intra-cell variation — prose style, paraphrases, irrelevant entities.
What is kept? Inter-cell variation — the only signal the forecast depends on.



The MI inequality is a direct consequence of this quotient: averaging distributions *cannot increase* KL divergence to a fixed reference (Jensen).

Theorem A.5 — Gating Quadratically Attenuates Errors

Setup. Forecaster F is coordinate-wise Lipschitz: L_k in the k-th primitive slot.
Let $\Delta_k = \|h_k^{\text{err}} - h_k^{\text{true}}\|$ be the embedding error of primitive k.

Error bound

$$\left(\hat{Y}_t^{\text{err}} - \hat{Y}_t^{\text{true}}\right)^2 \;\le\; K \sum_{k=1}^{K} L_k^2 \, \mathbf{g}_{t,k}^2 \mathbf{g}_{t,k}^2 \mathbf{g}_{t,k}^2 \mathbf{g}_{t,k}^2 \, \Delta_k^2$$

Why is this a big deal?

- Naive fusion has an effective $g \equiv 1$: error scales as $L_k^2 \Delta_k^2$.
- TESS trains $g_{t,k} \to 0$ for wrong primitives $\Rightarrow$ **quadratic suppression**.
- A 0.3 gate value dampens the error term by $\sim 11\times$; a 0.1 gate by $100\times$.

This is why gating is soft, not hard: the theorem rewards continuous down-weighting proportionally to confidence, not binary filtering.

Theorem A.5 — Proof in One Picture

Telescoping across primitives: Flip them one at a time from "true" to "err":

$$z^{(0)} = (h_1^{\text{true}}, \dots, h_K^{\text{true}}) \;\to\; z^{(1)} \;\to\; \cdots \;\to\; z^{(K)} = (h_1^{\text{err}}, \dots, h_K^{\text{err}})$$

Triangle inequality + coordinate-Lipschitz:

$$|F(E, z^{(K)}) - F(E, z^{(0)})| \le \sum_{k=1}^{K} |F(E, z^{(k)}) - F(E, z^{(k-1)})| \le \sum_{k=1}^{K} L_k \underbrace{\|g_{t,k}(h_k^{\text{err}} - h_k^{\text{true}})\|}_{=g_{t,k}\Delta_k}$$

Square both sides and apply Cauchy–Schwarz $(\sum a_k)^2 \le K \sum a_k^2$:

$$(\hat{Y}_t^{\text{err}} - \hat{Y}_t^{\text{true}})^2 \le K \sum_{k=1}^{K} L_k^2 g_{t,k}^2 \Delta_k^2. \qquad \blacksquare$$

Moral: the g^2 factor is structural — it comes from squaring the propagated error, not from a clever choice.

Theorem A.6 — Why Few Labels Generalise So Well

With $M = \prod_k M_k$ (size of primitive space) and n samples:

$$\sup_{g:\mathcal{V}\to[-1,1]} \left|R(g) - \hat{R}_n(g)\right| \;\le\; C\left(\sqrt{\frac{M}{n}} + \sqrt{\frac{\log(1/\delta)}{n}}\right)$$

Proof ingredients. Bound Rademacher complexity via partition counting:

$$\widehat{\mathfrak{R}}_n(\mathcal{G}) = \mathbb{E}_\sigma \sup_g \tfrac{1}{n}\sum_i \sigma_i g(V_i) = \tfrac{1}{n}\sum_v \mathbb{E}\left|\textstyle\sum_{i:V_i=v}\sigma_i\right| \le \tfrac{1}{n}\sum_v \sqrt{N_v} \overset{\text{CS}}{\le} \sqrt{M/n}$$

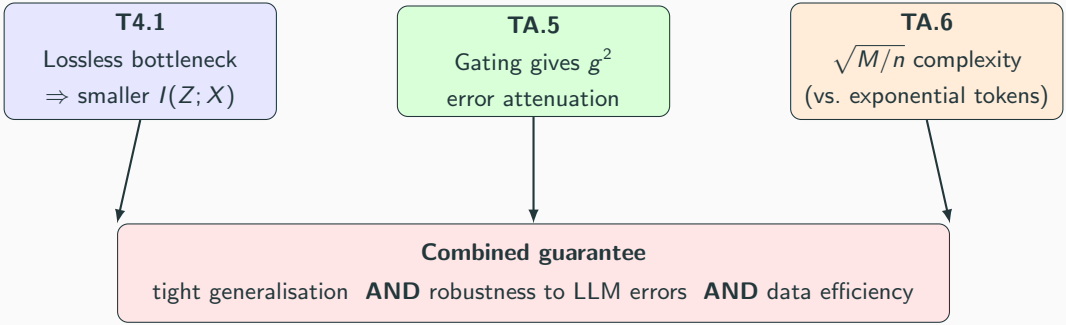
Compare hypothesis classes:

Representation	complexity rate
Raw tokens, length T, alphabet $\lvert\mathcal{A}\rvert$	$\sqrt{\lvert\mathcal{A}\rvert^T/n}$ (exponential)
Primitives, K=4 slots	$\sqrt{M/n}$ with $M = 5\times5\times5\times6 = 750$

For $n\sim10^4$: token rate $\gg 1$ (vacuous); primitive rate ≈ 0.27 (informative).

This is the **quantitative reason** TESS can be trained effectively on modest data.

How the Three Theorems Interlock



Each theorem attacks one failure mode:

- **T4.1**: "Does the bottleneck hurt?" — No, predictive info preserved.
- **TA.5**: "Does LLM error hurt?" — Only up to g^2-scaled.
- **TA.6**: "Does low data hurt?" — Far less than with tokens.

What Theory Predicts, Experiments Confirm

Theoretical prediction	Empirical confirmation
T4.1: Lower $I(Z; X_{\text{text}}) \Rightarrow$ smaller gen. gap $\Rightarrow$ *smoother, faster* convergence	Fig. 6: TESS converges faster and with smaller loss oscillation than direct text fusion.
TA.5: Wrong primitives get suppressed *continuously* by confidence	Fig. 8(b): correct primitives $g \in [0.65, 0.78]$; wrong ones $g \in [0.21, 0.40]$.
TA.6: Tiny discrete space beats token features when data is limited	Bitcoin (only 858 steps!): 29% MSE reduction vs. best baseline.
Primitives are *complementary* (each covers one axis: what / how / when)	Table ablation: removing any primitive hurts; Mean-Shift removal costs +33% MSE.

Theory is not decorative — every theorem has a matching experimental fingerprint.

Experiments

Main Results — 4 Datasets

Model	Bitcoin			FNSPID		
	MAE	MSE	RMSE	MAE	MSE	RMSE
PatchTST	1.362	3.246	1.802	0.0160	0.0016	0.0402
TimesNet	1.460	3.823	1.955	0.0152	0.0015	0.0385
TimeLLM	1.394	3.448	1.857	0.0158	0.0017	0.0412
NewsForecasting	1.359	3.204	1.790	0.0176	0.0017	0.0412
TESS (Ours)	**1.112**	**2.273**	**1.508**	**0.0147**	**0.0012**	**0.0347**
Gain vs. best	+18.2%	**+29.1%**	+15.8%	+3.3%	+20.0%	+9.9%

- Largest gains on *event-driven* financial data — aligned with the motivation.
- Competitive on general datasets (Electricity best, Environment runner-up).

Where does TESS help? Non-Stationary Scenarios

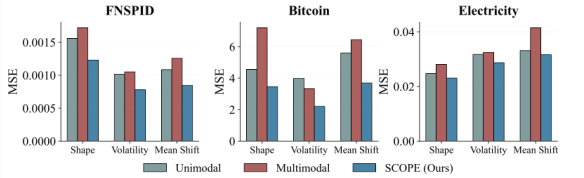


Figure 5: Performance comparison on three types of non-stationary scenarios (Shape, Volatility, Mean Shift). TESS (blue) consistently outperforms both unimodal (teal) and multimodal (red) baselines.

Figure 5 of paper

Three test subsets carved by non-stationarity type:

- Shape transition
- Volatility change
- Mean shift

TESS wins on all three:

- 21–52% MSE reduction over multimodal baselines
- 21–45% over unimodal baselines

Ablation & Gating Diagnostics

Component ablation (MSE ↑ when removed):

- w/o TESS: +46.2% (Bitcoin)
- w/o Gating: +3.7% (Bitcoin)

Primitive ablation (FNSPID):

- w/o Mean Shift: **+33%**
- w/o Volatility / Shape / Lag: smaller but non-trivial

Gating sanity check:

- Correct primitives: g median 0.65–0.78
- Incorrect primitives: g median 0.21–0.40

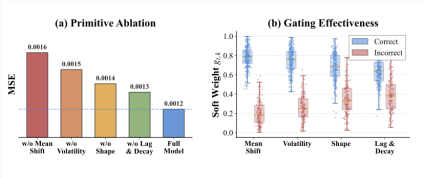


Figure 8 of paper

Optimisation Dynamics

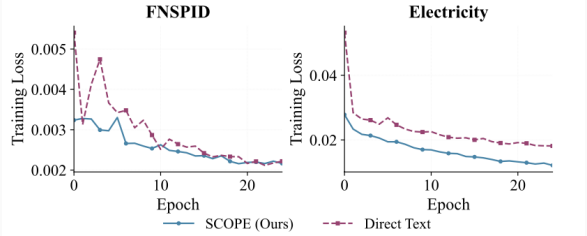


Figure 6: Training loss curves comparison. TESS (blue) converges faster with smoother dynamics than direct text fusion (purple).

Figure 6 of paper

TESS converges faster and smoother than direct text fusion.

This is the *empirical fingerprint* of Theorem 4.1:

- Lower $I(\tilde{Z}; X_{\text{text}})$
- Tighter generalisation gap
- Less noise in gradients

Case Study

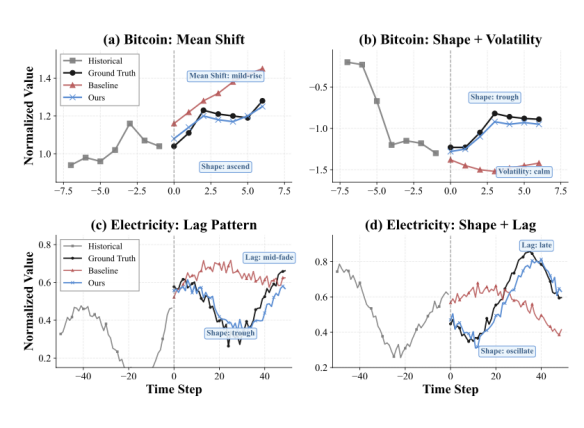


Figure 7 — TESS captures mild-rise / trough / mid-fade / late patterns

Direct text fusion fails these regime transitions; TESS aligns closely with ground truth.

Conclusion

Takeaways

1. The text–time-series **modality gap** is not a noise problem — it is a *representational* one, empirically diagnosed via focus-ratio and three-variant experiments.
2. **TESS** introduces a small, numerically verifiable *Temporal Evolution Semantic Space* as a principled bottleneck.
3. **Theory-driven design — three theorems, three guarantees**:
   - **T4.1**: sufficiency $\rightarrow$ lossless compression $\rightarrow$ tighter generalisation gap.
   - **TA.5**: Lipschitz forecaster $\rightarrow$ gating gives g^2 error attenuation.
   - **TA.6**: finite discrete class $\rightarrow$ $\sqrt{M/n}$ vs. exponential token rate.
4. Up to **29% MSE reduction** over SOTA uni- and multi-modal baselines, with each theoretical prediction matched by an empirical fingerprint.