COMPOSITIONAL DISCRETE FORECASTING WITH ORTHOGONAL FREQUENCY-BAND CODEBOOKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Long-term time-series forecasting must extrapolate recurring local structures whose frequency components reappear in different combinations. Continuous representations compose these factors flexibly but do not expose a finite vocabulary of reusable patterns. Existing discrete methods typically bind complete patches to single codes, tokenize transform coefficients, or retain continuous residual paths, leaving compositional codebook-based forecasting unresolved. We propose *PatchVQFormer*, which factorizes each patch into orthogonal frequency bands and represents it as a composition of frozen, band-specific waveform prototypes. Training-set energy and predictability determine the band partition, while band-dependent codebook capacities are selected on validation data. A spatial-then-temporal backbone uses masked future queries to predict all future band distributions in parallel. Training combines hard token classification with differentiable sequence reconstruction, whereas inference uses parameter-free soft expectations over frozen waveform prototypes. Across eight multivariate benchmarks and four horizons, PatchVQFormer achieves an average MSE/MAE of **0.291/0.342** and ranks first in **11/32** MSE and **8/32** MAE comparisons. On the four ETT benchmarks at $H \in \{336, 720\}$, it improves average MSE over the strongest listed competitor by **1.8%/1.0%**. In controlled ETT ablations, frequency-band factorization reduces average MSE over the no-Fourier variant by **4.5%**. These results establish compositional frequency-band vocabularies as an accurate and structured approach to long-horizon forecasting without a continuous residual decoder.

1 INTRODUCTION

Long-term time-series forecasting (LSTF) supports planning and decision making in complex systems where predictions must remain reliable across extended horizons. Representative applications include traffic forecasting over road networks, rainfall runoff prediction for hydrological management, and global medium-range weather forecasting (Li et al., 2018; Kratzert et al., 2019; Lam et al., 2023). Across these domains, local segments combine slowly varying trends, periodic oscillations, and faster fluctuations, with the same components recurring in different configurations. Long-horizon prediction must extrapolate these recurring structures beyond the observed context without losing local detail. Effective forecasting should therefore capture long-range dependencies while reusing shared components rather than treating every complete pattern as unrelated. We show that forecasting compositions of reusable frequency-band waveforms provides this structure and delivers its strongest gains at long horizons.

Continuous LSTF models flexibly superpose temporal factors in distributed representations, and linear, patch-based, and variate-centric designs achieve strong accuracy (Zeng et al., 2023; Nie et al., 2023; Liu et al., 2024b). Discrete models offer complementary structure by mapping recurring patterns to a finite vocabulary with stable semantics. TOTEM learns transferable time-series tokens, WaveToken predicts quantized wavelet coefficients, and ReCast combines whole-patch codes with a continuous residual path (Talukder et al., 2024; Masserano et al., 2025; Ma et al., 2026). Continuous vectors can express new combinations through superposition. A whole-patch code instead represents the joint configuration of all frequency components and does not explicitly reuse them independently across new combinations. Coefficient tokens do not form complete band-limited waveform prototypes, while residual decoding leaves forecasts only partially determined by discrete codes. This

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

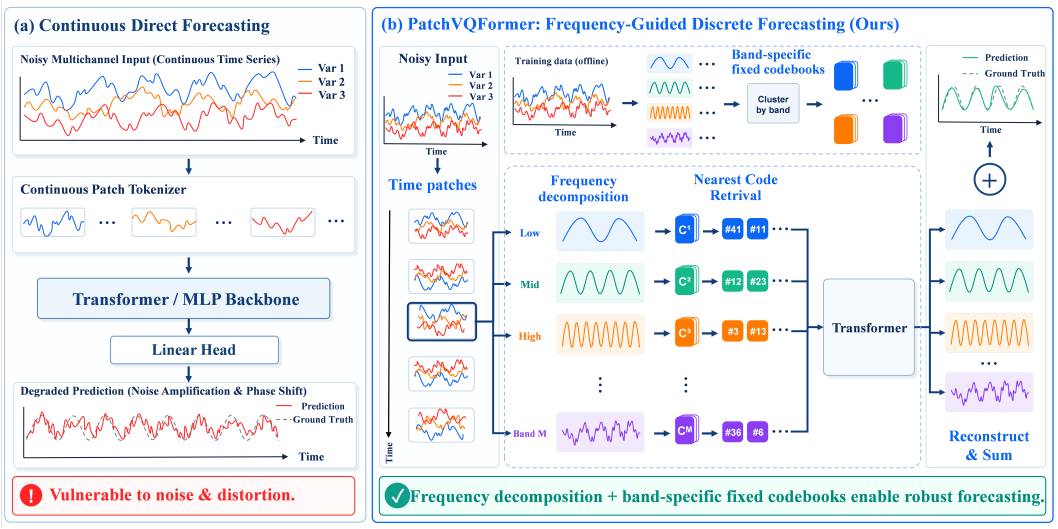


Figure 1: Overview of continuous direct forecasting and the proposed frequency-guided discrete forecasting framework. The left panel shows a conventional continuous pipeline, while the right panel illustrates band-wise decomposition, discrete code prediction, and prototype reconstruction.

raises our central question: *can a discrete forecaster preserve compositional expressiveness and accurate long-horizon prediction through a fully codebook-based output path?*

We propose **PatchVQFormer**, a compositional discrete forecaster that represents each patch as a sum of prototypes selected from orthogonal frequency-band codebooks. Figure 1 contrasts this compositional discrete pipeline with conventional continuous forecasting. Training-set energy and autocorrelation-based predictability determine fixed bands that partition the complete spectrum. Their inverse-transformed components sum exactly to the original patch before quantization. Orthogonality separates band-wise reconstruction errors, allowing each codebook to be allocated and diagnosed independently. Each band is clustered into a frozen waveform prototype table, with its capacity selected on validation data. This unequal allocation concentrates vocabulary resolution in bands where additional capacity is most useful. The selected codes retrieve trainable prototype-indexed embeddings that are summed into one patch token. The backbone first contextualizes each token across variables and then models temporal evolution with masked future queries, predicting all future band distributions in parallel. Soft expected-prototype decoding supplies reconstruction supervision during training and forms the primary inference path. Every forecast is a parameter-free weighted composition of frozen waveform prototypes, without a continuous residual decoder.

We evaluate PatchVQFormer on eight multivariate benchmarks across horizons of 96, 192, 336, and 720, with aggregate claims computed over all reported settings. Across the 32 dataset and horizon settings, PatchVQFormer achieves an average MSE/MAE of **0.291/0.342** and ranks first in **11** MSE and **8** MAE comparisons. At $H = 720$ and 336, it improves average ETT MSE over the strongest listed competitor by **1.0%/1.8%**, confirming its advantage for long-horizon prediction. Every prediction is reconstructed from frozen waveform prototypes, demonstrating that the discrete path alone carries the full output. In controlled ETT ablations, frequency-band factorization reduces average forecasting MSE and MAE over the no-Fourier variant by **4.5%** and **3.2%**. These results establish compositional frequency-band vocabularies as an accurate and structured alternative to continuous forecasting. Our contributions are summarized as follows:

- We introduce a statistics-guided tokenizer that represents patches with band-specific waveform prototypes from orthogonal frequency subspaces.
- We develop a multivariate forecaster that predicts future band distributions in parallel and reconstructs outputs by parameter-free soft expectations over frozen waveform prototypes.
- Across eight benchmarks, PatchVQFormer delivers its strongest gains at long horizons, while matched comparisons validate frequency-band factorization.

2 Related Work

Continuous Long-Term Forecasting. Continuous LSTF methods differ in temporal and cross-variable modeling. Informer uses sparse attention and one-pass decoding for long sequences, while DLinear establishes direct linear mappings as strong baselines (Zhou et al., 2021; Zeng et al., 2023). PatchTST maps subsequences to patch tokens and applies channel-independent processing, whereas ModernTCN uses large convolutional kernels for temporal patterns (Nie et al., 2023; Luo & Wang, 2024). Both retain continuous representations and regression outputs. Multivariate models differ in their variable interactions. Crossformer explicitly couples temporal and cross-dimension dependencies, while iTransformer treats variable histories as tokens and applies attention across variables (Zhang & Yan, 2023; Liu et al., 2024b). DUET combines temporal clustering with soft channel clustering to accommodate heterogeneous patterns and changing variable relationships (Qiu et al., 2025). PatchVQFormer performs cross-variable interaction before temporal patch modeling so that multivariate context can help disambiguate compressed codes before temporal extrapolation.

Frequency-Structured Forecasting. Signal decomposition and transformed-domain modeling provide LSTF representations. Autoformer embeds trend and seasonal decomposition, while FEDformer combines decomposition with frequency-domain modeling (Wu et al., 2021; Zhou et al., 2022b). TimesNet reorganizes a series using multiple periods for two-dimensional modeling, PDF decouples inter-period and intra-period variation, and TimeMixer mixes decomposed patterns across temporal scales (Wu et al., 2023; Dai et al., 2024; Wang et al., 2024). These methods expose multiscale structure while retaining continuous representations and outputs. FiLM augments Legendre projections with frequency information, and FITS interpolates complex Fourier coefficients (Zhou et al., 2022a; Xu et al., 2024). WFTNet combines Fourier and wavelet representations to capture global and local periodicity, while WPMixer applies patch-wise MLP mixing after multiresolution wavelet decomposition (Liu et al., 2024a; Murad et al., 2025). Other approaches address distribution change through reversible normalization and de-stationary factors, or model periodic dynamics through explicit phase representations (Kim et al., 2022; Liu et al., 2022; Niu et al., 2026; Liu et al., 2026). PatchVQFormer instead uses training-set energy and predictability to define an orthogonal partition with validation-selected band capacities, so frequency structure determines vocabulary semantics rather than shaping continuous features or computation.

Discrete and Compositional Time-Series Forecasting. Vector quantization replaces continuous observations or latent features with indices from a finite vocabulary. VQ-VAE provides the learned-codebook framework, and VQ-AR applies vector quantization to autoregressive probabilistic time-series forecasting (van den Oord et al., 2017; Rasul et al., 2022). Forecasting methods differ in what they discretize and how the resulting codes determine future values. Chronos scales and quantizes individual observations into a fixed vocabulary for language-model-style probabilistic forecasting, whereas TOTEM learns reusable tokens for local time-series segments and transfers them across analysis tasks (Ansari et al., 2024; Talukder et al., 2024). Sparse-VQ combines sparse vector quantization with reversible normalization in an FFN-free forecasting Transformer (Zhao et al., 2024). TimeVQVAE learns frequency-separated discrete latents for generation, while WaveToken autoregressively predicts quantized wavelet coefficients after decomposition (Lee et al., 2023; Masserano et al., 2025). ReCast learns reliability-aware patch codes and restores omitted information through a continuous residual path (Ma et al., 2026). PatchVQFormer instead composes a patch from one time-domain waveform prototype per band of an exhaustive spectral partition. It predicts future band distributions in parallel and reconstructs forecasts by summing soft expectations of frozen prototypes, without autoregressive feedback or continuous decoders.

3 Problem Formulation

Multivariate Long-Term Forecasting. Given a history $X \in \mathbb{R}^{L\times C}$ with C variables and L observations, the task is to predict the next H observations $Y \in \mathbb{R}^{H\times C}$. Each variable is normalized using the mean and standard deviation of its history, producing $\bar{X}$ and $\bar{Y}$, and the same statistics restore the final forecast. We consider valid configurations with $18 \le P \le \min(L, H)$ and use overlapping patches of length P with stride $S = \lfloor P/2 \rfloor$. For $T \in \{L, H\}$, define $\mathcal{I}(T) = \{0, S, \ldots, \lfloor (T-P)/S \rfloor S\} \cup \{T-P\}$, where the terminal start ensures complete coverage

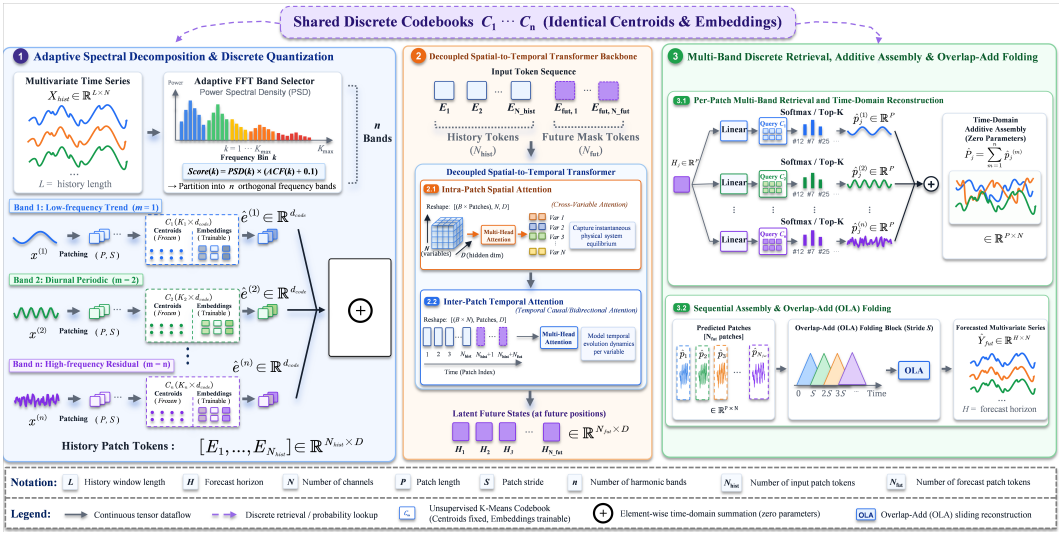


Figure 2: PatchVQFormer decomposes historical patches into band indices, models cross-variable temporal context, and reconstructs forecasts from frozen waveform-prototype distributions.

without padding or truncation. Writing the ordered starts as $r_1^{(T)} < \cdots < r_{|\mathcal{I}(T)|}^{(T)}$, set $N = |\mathcal{I}(L)|$, $J = |\mathcal{I}(H)|$, $p_{i,c} = (\bar{X}_{r_i^{(L)}+t,c})_{t=0}^{P-1}$, and $y_{j,c} = (\bar{Y}_{r_j^{(H)}+t,c})_{t=0}^{P-1}$.

Compositional Representation and Band-Specific Resolution. We represent each patch with M orthogonal frequency bands and K_m reusable waveform prototypes in band m, assigning band-specific resolution instead of one whole-patch code. The capacity vector $\mathbf{K} = (K_1, \ldots, K_M)$ is selected on validation data and fixed before final fitting and training.

Contextual Disambiguation of Compressed Patches. The same local code composition can correspond to different future dynamics under different multivariate contexts. PatchVQFormer therefore contextualizes each compressed patch across variables before modeling its temporal evolution, allowing cross-variable evidence to refine the representation for future-band prediction.

4 Method

PatchVQFormer forecasts future patches by composing reusable frequency-band waveform prototypes (Figure 2). The tokenizer partitions patches into orthogonal bands and binds each index to a frozen prototype and trainable embedding (Section 4.1). The context-first backbone models cross-variable dependencies before temporal evolution and predicts categorical band distributions in parallel (Section 4.2). Expected-prototype decoding maps distributions to band-limited waveforms that sum to each future patch (Section 4.3). Token classification and sequence reconstruction jointly supervise categorical prediction and forecast accuracy (Section 4.3). Section 4.4 establishes compositional capacity, additive band-wise distortion, and historical assignment stability.

4.1 Stage 1: Orthogonal Frequency-Band Tokenization

Patch-Wise Orthogonal Band Decomposition. For $p \in \mathbb{R}^P$, let $F \in \mathbb{C}^{P\times P}$ be the unitary DFT and B_m be conjugate-symmetric binary masks satisfying $\sum_m B_m = I$ and $B_m B_n = 0$ for $m \neq n$. The band projections are

$$\Pi_m = F^* B_m F, \qquad p^{(m)} = \Pi_m p, \qquad p = \sum_{m=1}^{M} p^{(m)}. \tag{1}$$

Orthonormal rFFT / irFFT operations implement these projections with one-sided masks, producing real, mutually orthogonal components that exactly recover p before quantization. Each setting fixes M; boundaries are discovered from normalized training patches, then fixed and reported

with the experimental setup. For sampled normalized training patches $\mathcal{A}$, let $R = \lfloor P/2 \rfloor$ and $\widehat{p}_k = \mathrm{rFFT}(p)_k$, and define the multiplicity-adjusted energy profile by

$$\gamma_k = \begin{cases} 1, & k = 0 \text{ or } (P \text{ even and } k = R), \\ 2, & \text{otherwise}, \end{cases} \qquad e_k = \frac{\gamma_k}{|\mathcal{A}|} \sum_{p \in \mathcal{A}} |\widehat{p}_k|^2, \qquad \rho_k = \frac{e_k}{\sum_{r=0}^{R} e_r}, \tag{2}$$

For each $k = 1, \ldots, R$, let $\ell_k = \mathrm{round}(P/k)$ approximate the period associated with bin k, and define the nonnegative mean lagged-product and discovery scores as

$$q_k = \begin{cases} \left[\dfrac{1}{|\mathcal{A}|} \displaystyle\sum_{p \in \mathcal{A}} \frac{1}{P - \ell_k} \sum_{t=0}^{P-\ell_k-1} p_t p_{t+\ell_k} \right]_+, & \ell_k < P, \\ 0, & \ell_k \geq P, \end{cases} \qquad s_k = \begin{cases} \rho_0, & k = 0, \\ \rho_k (q_k + 0.1), & 1 \leq k \leq R, \end{cases} \tag{3}$$

Thus s_k favors bins that are both energetic and repeatable. All configurations satisfy $R \geq 9$. If $\sum_{r=1}^{R} \rho_r = 0$, set $h = R$; otherwise, set $h = \min\left\{R, 1 + \min\left\{k \geq 8 : \frac{\sum_{r=1}^{k} \rho_r}{\sum_{r=1}^{R} \rho_r} \geq 0.85\right\}\right\}$. Bins $h, \ldots, R$ form a nonempty final band. Below h, bins are scanned in ascending frequency, and up to three local maxima satisfying $s_k > 0.02$ are retained as singleton bands when the resulting runs can form an M-band partition. Unassigned bins form contiguous runs split at their midpoints from longest to shortest until the partition contains M bands, with lower-frequency runs taking precedence under ties. DC remains in the first band and the even-P Nyquist bin in the final band, yielding deterministic, disjoint, exhaustive masks that preserve Equation 1.

Offline Waveform Prototypes and Codebooks. For each band m with fixed capacity K_m, fit the prototype table $\mathcal{C}_m = \{c_k^{(m)}\}_{k=1}^{K_m} \subset \mathrm{range}(\Pi_m)$ using training-only patches $\mathcal{P}_{\mathrm{train}}$:

$$\min_{\mathcal{C}_m} \sum_{p \in \mathcal{P}_{\mathrm{train}}} \min_{1 \leq k \leq K_m} \left\| \Pi_m p - c_k^{(m)} \right\|_2^2. \tag{4}$$

Each shared codeword binds index k, frozen prototype $c_k^{(m)}$, and trainable embedding $e_k^{(m)} \in \mathbb{R}^{d_{\mathrm{model}}}$. Encoding selects $k^{(m)}(p) = \arg\min_{1 \leq k \leq K_m} \|\Pi_m p - c_k^{(m)}\|_2^2$, breaking ties by index order. Training freezes $\mathcal{C}_m$ and optimizes $\mathcal{E}_m = \{e_k^{(m)}\}_{k=1}^{K_m}$, preserving waveform semantics while adapting embeddings. Prototype fitting pools training variables, windows, and positions using sampled coordinates shared across bands. Each table uses Lloyd k-means with squared Euclidean distance. Sampling, initialization, convergence, and empty-cluster handling are detailed in the appendix.

Band-Dependent Capacity and Compositional Reconstruction. With hard assignments, the codec reconstructs $\widetilde{p} = \sum_m c_{k^{(m)}(p)}^{(m)}$. Storing $\sum_m K_m$ prototypes supports up to $\prod_m K_m$ hard cross-band compositions, while categorical distributions form convex combinations spanning a continuous compositional space. For each dataset-horizon setting, candidate vectors are jointly ranked by validation forecasting MSE using training-only fits; the selected $\mathbf{K}$ is fixed for final prototype fitting and predictor training. The spectral score determines boundaries, whereas K_m determines band-specific resolution.

4.2 Stage 2: Context-First Discrete Forecasting

Compositional Historical Tokens. Each historical patch retrieves one trainable embedding per band through its fixed prototype indices. These embeddings are waveform-initialized and optimized with the backbone while preserving the index-to-prototype correspondence. The resulting token is $h_{i,c} = \sum_{m=1}^{M} e_{k^{(m)}(p_{i,c})}^{(m)} + a_i + v_c + g(\tau_i)$, where a_i and v_c encode patch position and variable identity. The covariates τ_i contain the calendar and sampling-time features available at patch position i, and g maps them to the model dimension. Stacking all tokens gives $H^{\mathrm{hist}} \in \mathbb{R}^{N \times C \times d_{\mathrm{model}}}$. No continuous raw-patch projection is added, so waveform content enters the backbone only through codebook indices. Waveform-based initialization details are provided in the appendix.

Spatial Attention. For future patch j and variable c, define $f_{j,c} = b + a_{N+j} + v_c + g(\tau_{N+j})$, where b is a learned future-start embedding and τ_{N+j} contains only known temporal covariates. They contain no future values or target codes. Stacking them gives $H^{\mathrm{fut}} \in \mathbb{R}^{J \times C \times d_{\mathrm{model}}}$, and

patch-axis concatenation gives $Z = [H^{\text{hist}}; H^{\text{fut}}] \in \mathbb{R}^{(N+J)\times C\times d_{\text{model}}}$. At patch position t, the shared operator $\mathcal{S}_\theta : \mathbb{R}^{C\times d_{\text{model}}} \to \mathbb{R}^{C\times d_{\text{model}}}$ maps $Z_{t,:,:}$ to $U_{t,:,:}$, yielding $U \in \mathbb{R}^{(N+J)\times C\times d_{\text{model}}}$. Applying spatial attention first allows cross-variable evidence to refine compressed historical tokens and initialize future queries before temporal extrapolation.

Masked-Future Temporal Attention. The shared temporal operator acts along each variable's patch axis as $Q_{:,c,:} = \mathcal{T}_\phi(U_{:,c,:}; \mathcal{M})$, where $\mathcal{T}_\phi : \mathbb{R}^{(N+J)\times d_{\text{model}}} \to \mathbb{R}^{(N+J)\times d_{\text{model}}}$. The block mask $\mathcal{M}$ restricts historical queries to historical keys, while future queries attend to history and all future queries. With only known metadata, bidirectional future-query interaction captures horizon-wide dependencies without target leakage or autoregressive feedback. The model produces future states $q_{j,c} = Q_{N+j,c,:} \in \mathbb{R}^{d_{\text{model}}}$ in one parallel pass.

4.3 Output Distributions, Soft Decoding, and Training

Independent Classification Heads. Each band maps $q_{j,c}$ to logits $\ell_{j,c,m} = W_m \,\mathrm{LN}(q_{j,c}) + b_m \in \mathbb{R}^{K_m}$, defining a categorical distribution over its prototype indices. The heads are independent across bands and untied from the trainable input embeddings.

Parameter-Free Expected-Prototype Decoding. A fixed temperature $T > 0$, shared across bands and used unchanged in training and inference, gives $\pi^{(T)}_{j,c,m,k} = \mathrm{Softmax}(\ell_{j,c,m}/T)_k$. Each band yields the expected frozen prototype $\bar{c}^{(m)}_{j,c} = \sum_{k=1}^{K_m} \pi^{(T)}_{j,c,m,k} c^{(m)}_k$, and summing bands reconstructs $\widehat{p}_{j,c} = \sum_{m=1}^{M} \bar{c}^{(m)}_{j,c}$. Coverage-normalized overlap-add over $\mathcal{I}(H)$ forms the normalized sequence, which historical statistics restore to the original scale. With no trainable reconstruction parameters or continuous residual path, the frozen tables $\mathcal{C}_m$ fully determine output waveforms.

Token and Sequence Supervision. The frozen tokenizer assigns each future target patch the indices $k^*_{j,c,m} = k^{(m)}(y_{j,c})$. Token supervision averages cross-entropy over all future patches, variables, and bands as $\mathcal{L}_{\text{CE}} = -(JCM)^{-1} \sum_{j,c,m} \log \mathrm{Softmax}(\ell_{j,c,m})_{k^*_{j,c,m}}$. It aligns every categorical head with the corresponding hard codebook target. The inference decoder also produces $\widehat{\bar{Y}}$ during training, giving the normalized sequence loss $\mathcal{L}_{\text{seq}} = (HC)^{-1}\|\widehat{\bar{Y}} - \bar{Y}\|_F^2$ after overlap-add and before inverse normalization. This term coordinates bands and overlapping patches around final forecast accuracy. The total objective is $\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{seq}}\mathcal{L}_{\text{seq}}$, with $\lambda_{\text{seq}} = 1$ by default.

4.4 Analysis: Representation Capacity and Stability

Additive Distortion in Orthogonal Subspaces. For any patch p and band-wise reconstruction $v_m \in \mathrm{range}(\Pi_m)$, orthogonality gives $\|p - \sum_{m=1}^{M} v_m\|_2^2 = \sum_{m=1}^{M} \|\Pi_m p - v_m\|_2^2$. It applies to hard prototypes and soft expectations. It decomposes total patch distortion into exact band-wise contributions, supporting per-codebook capacity allocation and reconstruction diagnosis.

Stability of Historical Discrete Assignments. For a normalized band component $u = \Pi_m p$ with unique nearest prototype $c^{(m)}_a$, define the margin $\Delta_m(u) = \min_{b\neq a} \|u - c^{(m)}_b\|_2 - \|u - c^{(m)}_a\|_2$. For $K_m \geq 2$, any perturbation satisfying $\|\Pi_m \epsilon\|_2 < \Delta_m(u)/2$ preserves the assigned index. If this condition holds across all historical patches and bands while metadata remains fixed, the deterministic backbone receives the same discrete history. This gives a local stability certificate for historical tokenizer inputs, with proof details in App. C.

Monotonic Capacity-Distortion Relation. For band m, define the optimal empirical quantization distortion at capacity K as

$$D_m(K) = \min_{\substack{\mathcal{C}\subseteq \mathrm{range}(\Pi_m) \\ |\mathcal{C}|\leq K}} \sum_{p\in\mathcal{P}_{\text{train}}} \min_{c\in\mathcal{C}} \|\Pi_m p - c\|_2^2. \tag{5}$$

The nested feasible codebook families imply $D_m(K+1) \leq D_m(K)$, so additional prototypes cannot worsen optimal band reconstruction. Combined with additive distortion, this relation supports band-wise reconstruction diagnosis, while validation jointly selects capacities for forecasting. Freezing the fitted prototypes then preserves the selected vocabulary throughout predictor training.

Table 1: Multivariate long-term forecasting results for $H \in \{96, 192, 336, 720\}$. Lower is better; bold and underlined entries mark the best and second-best result in each row and metric.

Models		PatchVQFormer (ours)		PhaseFormer (2026)		DUET (2025)		WPMixer (2025)		PDF (2024)		iTransformer (2024)		PatchTST (2023)		DLinear (2023)	
Data	H	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.372	0.413	0.362	**0.389**	0.377	0.394	0.374	0.404	**0.359**	0.393	0.394	0.408	0.368	0.402	0.369	0.392
	192	0.416	0.440	0.407	**0.414**	0.407	0.428	0.412	0.427	**0.392**	0.414	0.444	0.436	0.410	0.419	0.405	0.414
	336	**0.410**	0.430	0.419	**0.417**	0.422	0.441	0.437	0.436	0.413	0.434	0.470	0.465	0.426	0.439	0.441	0.446
	720	**0.418**	0.452	0.425	**0.442**	0.433	0.467	0.550	0.513	0.461	0.472	0.493	0.483	0.459	0.473	0.490	0.505
ETTh2	96	0.283	0.346	0.285	0.346	**0.275**	0.342	0.281	0.344	0.279	**0.340**	0.293	0.344	0.282	0.342	0.282	0.351
	192	0.348	0.392	0.351	0.385	**0.339**	0.384	0.344	0.390	0.339	**0.380**	0.379	0.399	0.350	0.387	0.361	0.405
	336	**0.340**	**0.395**	0.362	0.404	0.368	0.405	0.354	0.397	0.353	0.401	0.406	0.428	0.360	0.399	0.445	0.459
	720	**0.388**	0.433	0.399	**0.431**	0.403	0.439	0.420	0.449	0.400	0.437	0.446	0.455	0.398	0.435	0.689	0.591
ETTm1	96	**0.281**	**0.340**	0.302	0.350	0.292	0.346	0.294	0.344	0.300	0.348	0.325	0.370	0.283	0.344	0.301	0.345
	192	0.328	0.370	0.331	**0.364**	0.330	0.375	0.331	0.371	0.335	0.372	0.364	0.399	**0.325**	0.371	0.336	0.375
	336	0.361	0.394	0.362	**0.384**	0.362	0.394	0.370	0.391	0.366	0.388	0.391	0.411	**0.361**	0.397	0.366	0.387
	720	0.419	0.428	**0.414**	**0.413**	0.419	0.422	0.425	0.419	0.419	0.416	0.445	0.442	0.418	0.424	0.423	0.425
ETTm2	96	**0.161**	**0.250**	0.171	0.259	0.169	0.263	0.168	0.255	0.168	0.259	0.179	0.266	0.165	0.255	0.165	0.258
	192	0.219	0.290	0.227	0.298	0.230	0.300	0.225	**0.290**	0.225	0.298	0.252	0.311	0.222	0.294	**0.218**	0.297
	336	0.272	**0.324**	0.277	0.331	**0.271**	0.330	0.276	0.326	0.277	0.333	0.316	0.352	0.275	0.327	0.280	0.343
	720	0.355	**0.377**	0.358	0.385	0.363	0.388	0.356	0.380	**0.350**	0.380	0.393	0.411	0.356	0.383	0.365	0.399
Exchange	96	0.106	0.233	0.094	0.216	0.085	0.203	0.082	**0.199**	0.082	0.202	0.094	0.218	0.083	0.201	**0.078**	0.201
	192	0.193	0.318	0.192	0.313	0.180	0.301	0.173	0.298	0.173	0.295	0.187	0.312	0.177	0.299	**0.154**	**0.291**
	336	0.343	0.428	0.350	0.430	0.347	0.427	0.335	0.420	0.322	0.409	0.353	0.430	0.331	0.416	**0.277**	**0.406**
	720	0.933	0.731	0.914	0.724	0.853	0.698	0.890	0.706	0.842	0.690	0.857	0.702	0.856	0.695	**0.834**	**0.686**
Weather	96	**0.146**	0.196	0.153	0.202	0.150	0.198	0.147	**0.191**	0.164	0.216	0.161	0.214	0.148	0.197	0.168	0.224
	192	0.201	0.250	0.198	0.246	0.194	0.240	0.189	**0.235**	0.207	0.254	0.206	0.253	**0.189**	0.238	0.212	0.270
	336	0.259	0.298	0.246	0.283	0.244	0.278	**0.238**	**0.272**	0.253	0.289	0.256	0.290	0.240	0.279	0.257	0.307
	720	0.331	0.350	0.316	0.337	0.315	0.332	0.320	**0.328**	0.316	0.336	0.328	0.339	**0.312**	0.331	0.317	0.358
Electricity	96	0.132	0.229	0.142	0.236	0.159	0.278	**0.128**	**0.222**	0.146	0.245	0.133	0.231	0.141	0.245	0.133	0.230
	192	**0.145**	0.251	0.165	0.256	0.176	0.292	0.147	**0.237**	0.162	0.261	0.149	0.246	0.155	0.258	0.151	0.244
	336	0.177	0.278	0.184	0.276	0.192	0.306	0.161	**0.255**	0.178	0.278	0.165	0.264	0.171	0.274	**0.160**	0.261
	720	0.242	0.339	0.236	0.315	0.231	0.336	0.199	**0.290**	0.220	0.311	0.201	0.294	0.210	0.306	**0.195**	0.294
Solar	96	**0.163**	**0.227**	0.180	0.241	0.201	0.276	0.172	0.228	0.192	0.253	0.172	0.245	0.167	0.235	0.190	0.255
	192	**0.176**	**0.233**	0.211	0.260	0.216	0.285	0.185	0.243	0.218	0.272	0.193	0.258	0.190	0.269	0.211	0.272
	336	**0.190**	**0.247**	0.231	0.267	0.218	0.290	0.191	0.250	0.239	0.285	0.213	0.274	0.199	0.264	0.228	0.285
	720	0.208	0.278	0.235	0.276	0.224	0.291	**0.199**	**0.258**	0.245	0.289	0.208	0.271	0.208	0.280	0.236	0.294
1st Count		**11**	8	1	8	3	0	3	**11**	3	2	0	0	4	0	7	3

5 EXPERIMENTS

Datasets and Forecasting Protocol. We evaluate PatchVQFormer on eight multivariate forecasting benchmarks: ETTh1, ETTh2, ETTm1, ETTm2, Exchange, Weather, Electricity, and Solar. Each model predicts $H \in \{96, 192, 336, 720\}$ future steps from a historical context. Normalization statistics and frequency-band codebooks are fitted on training data, hyperparameters are selected on validation data, and the test split is reserved for final evaluation. The ETT datasets use chronological train/validation/test splits of 60%/20%/20%; the others use 70%/10%/20%.

Baselines and Metrics. We compare against representative continuous forecasters, including DLinear, PatchTST, iTransformer, PDF, WPMixer, DUET, and PhaseFormer. Forecasts are evaluated by mean squared error (MSE) and mean absolute error (MAE), where lower values indicate better performance. The baseline values in Table 1 follow their recorded configurations.

Implementation Details. Across four ETT benchmarks, ablations remove the Fourier decomposition, codebook, or patching module, while hyperparameter studies vary band count and codebook capacity. All variants retain the full model's data splits, backbone, optimization schedule, and evaluation procedure, changing only the specified factor. More reproduction configurations are in App. B.

5.1 MAIN RESULTS

Table 1 reports all 32 dataset–horizon settings. PatchVQFormer achieves an average MSE/MAE of $0.291/0.342$ and ranks first in 11 MSE and 8 MAE comparisons, giving it the highest MSE first-

Table 2: Ablation results averaged over four forecasting horizons on the ETT benchmarks. The final row reports the four-dataset average; lower MSE and MAE indicate better forecasting accuracy. Bold and underlined values denote the best and second-best results.

Dataset	Full		w/o Fourier		w/o Codebook		w/o Patch	
	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	**0.4269**	0.4453	0.4549	0.4672	0.4326	**0.4435**	0.4431	0.4574
ETTh2	**0.3560**	**0.3993**	0.3660	0.4064	0.3693	0.4107	0.3711	0.4125
ETTm1	0.3588	**0.3906**	0.3748	0.4031	0.3677	0.3977	**0.3580**	0.3921
ETTm2	**0.2574**	**0.3128**	0.2700	0.3223	0.2777	0.3342	0.2576	0.3150
Average	**0.3498**	**0.3870**	0.3664	0.3998	0.3618	0.3965	0.3575	0.3943

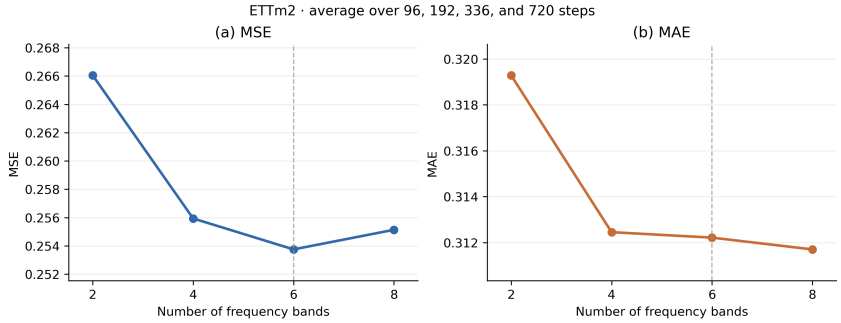


Figure 3: Frequency-band sensitivity on ETTm2, averaged over prediction lengths 96, 192, 336, and 720. The dashed line marks the selected six-band setting, and lower values are better.

place count among the listed methods. The advantage is strongest on long-horizon ETT forecasting. Averaged over ETTh1, ETTh2, ETTm1, and ETTm2, PatchVQFormer attains the lowest MSE at both $H = 336$ (0.346) and $H = 720$ (0.395), improving over the strongest listed competitor by 1.8% and 1.0%, respectively. It also achieves the best MSE on both ETTh1 and ETTh2 at these horizons and leads both metrics on ETTm1 and ETTm2 at $H = 96$, demonstrating that the fully codebook-based output path remains accurate as the forecasting horizon grows. The gains extend beyond ETT: PatchVQFormer leads both metrics on Solar at $H \in \{96, 192, 336\}$, obtains the best Weather MSE at $H = 96$, and achieves the best Electricity MSE at $H = 192$. Across all datasets, it records the lowest average MSE at $H = 336$ (0.294). Because the recorded baseline configurations use different input lengths, these results constitute a broad literature comparison, while the controlled ablations below isolate the contribution of each proposed component.

5.2 Ablation and Hyperparameter Analysis

Component Ablation. Table 2 isolates Fourier-band decomposition, discrete codebooks, and patching. Relative to the full model, removing Fourier decomposition increases average MSE and MAE by 4.7% and 3.3%, the largest degradation. Removing the codebook raises them by 3.4% and 2.5%, while removing patching raises them by 2.2% and 1.9%. The full model therefore achieves the best average MSE and MAE, validating all three components within the compositional framework.

Number of Frequency Bands. We vary the band count over $\{2, 4, 6, 8\}$ while fixing the total prototype budget and forecasting backbone. Figure 3 reports ETTm2 MSE and MAE averaged over the four forecasting horizons. Both metrics decrease from two to six bands, with six achieving the lowest average MSE. Eight bands slightly improves MAE but increases MSE, so we select six for the best MSE-MAE trade-off.

Codebook Capacity. Table 3 evaluates band capacities on ETTh2 with input length 512 and prediction length 336, holding all other settings fixed. The band-dependent allocation achieves the lowest MSE with 1248 prototypes, outperforming the uniform 256-per-band setting with 1536 prototypes. Increasing uniform capacity modestly improves MSE but not MAE consistently, with 64 prototypes per band yielding the lowest MAE. These results show that band-dependent allocation provides a better MSE–capacity trade-off than uniformly scaling the vocabulary.

Table 3: Codebook-capacity sensitivity on ETTh2 (input 512, prediction 336). Changes are relative to the main result; lower MSE and MAE indicate better forecasting accuracy.

Setting	Six band capacities	Total capacity	MSE	MAE	ΔMSE vs. main	ΔMAE vs. main
Main result	512, 256, 32, 64, 128, 256	1248	**0.3403**	0.3948	+0.00%	+0.00%
All 32	32, 32, 32, 32, 32, 32	192	0.3442	0.3948	+1.14%	+0.02%
All 64	64, 64, 64, 64, 64, 64	384	0.3438	**0.3937**	+1.02%	−0.26%
All 128	128, 128, 128, 128, 128, 128	768	0.3436	0.3960	+0.98%	+0.30%
All 256	256, 256, 256, 256, 256, 256	1536	0.3421	0.3964	+0.52%	+0.42%

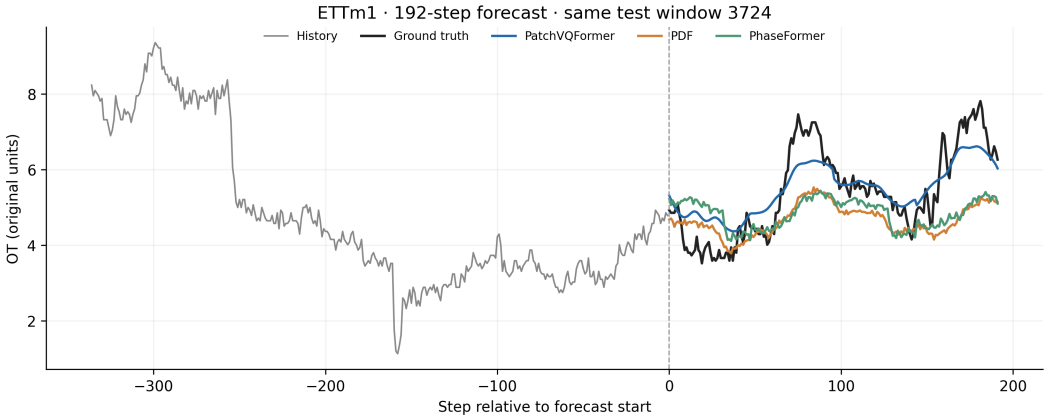


Figure 4: A 192-step forecasting case on ETTm1 (test window 3724). The plot shows the historical context and ground truth together with forecasts from PatchVQFormer, PDF, and PhaseFormer.

5.3 Visualization Analysis

Frequency-Band Prototypes. Figure 7 in App. D visualizes the most-used frozen time-domain prototypes in each band and their soft usage shares. Low-frequency codebooks contain smooth level and turning patterns, whereas higher bands contain progressively faster, lower-amplitude oscillations. Usage remains distributed across multiple prototypes, demonstrating band specialization without reducing the vocabulary to a single dominant waveform.

Forecasting Cases. Figure 4 compares the 192-step forecasts of PatchVQFormer, PDF, and PhaseFormer on an ETTm1 test window. PatchVQFormer follows the two pronounced rises in the second half and better matches their elevated levels than PDF and PhaseFormer. This case illustrates that codebook-based decoding can express horizon-scale level changes rather than repeating a fixed local template. Additional ETTh1 and ETTh2 cases appear in App. D.

6 Conclusion

We introduced PatchVQFormer, a compositional discrete forecaster that represents patches with orthogonal frequency-band prototypes. Training-only spectral statistics define the bands, validation selects their capacities, and a spatial-then-temporal backbone predicts future categorical distributions in parallel. Parameter-free prototype expectations reconstruct forecasts without a continuous residual path. Across eight benchmarks and 32 settings, PatchVQFormer achieves average MSE/MAE $0.291/0.342$ and the highest MSE first-place count. On four ETT datasets, it improves average MSE over the strongest listed competitor by $1.8\%/1.0\%$ at $H = 336/720$, while controlled ablations show that frequency-band factorization reduces MSE by 4.5%. Band-dependent allocation also outperforms a larger uniform vocabulary in MSE. These findings demonstrate that finite codebooks retain compositional expressiveness by recombining reusable band-specific waveforms, establishing fully codebook-based forecasting as an accurate, structured alternative for long-horizon prediction.

AI USE STATEMENT

Generative AI tools were used as auxiliary writing and research assistants during the preparation of this manuscript, including to improve language and organization, refine the presentation of figures and tables, and suggest potentially relevant literature and research directions. The authors manually accessed and verified every source identified with AI assistance through its original publication venue, such as conference proceedings, journal websites, OpenReview, or arXiv, before citing or discussing it. All methodological designs, mathematical formulations, experimental procedures, implementations, analyses, and conclusions were developed and verified by the authors, who take full responsibility for the accuracy, integrity, and final content of the manuscript.

ETHICS STATEMENT

This work did not involve human participants or the collection of private or personally identifiable information. All experiments used publicly available benchmark datasets in accordance with their respective licenses, terms of use, and research guidelines. Because this general-purpose forecasting approach could be applied in domains where prediction errors affect downstream decisions, its outputs should be treated as estimates rather than guaranteed future outcomes. Potential risks include error propagation, biases inherited from the training data, and inappropriate use in high-impact settings without adequate validation. We therefore recommend careful evaluation, uncertainty- aware interpretation, and domain-specific verification before practical deployment.

REPRODUCIBILITY STATEMENT

To support independent reproduction of the reported results, the paper and supplementary material describe the data preprocessing procedures, model architecture, training configurations, evaluation protocols, computational environment, key hyperparameters, and experimental settings. Upon acceptance, we will release the complete source code, configuration files, and associated experimental resources in a public repository, including the training and evaluation scripts required to reproduce the reported experiments under the documented settings and evaluation protocols.

REFERENCES

Abdul Fatir Ansari, Lorenzo Stella, Ali Caner Turkmen, Xiyuan Zhang, Pedro Mercado, Huibin Shen, Oleksandr Shchur, Syama Sundar Rangapuram, Sebastian Pineda Arango, Shubham Kapoor, Jasper Zschiegner, Danielle C. Maddix, Hao Wang, Michael W. Mahoney, Kari Torkkola, Andrew Gordon Wilson, Michael Bohlke-Schneider, and Bernie Wang. Chronos: Learning the Language of Time Series. *Transactions on Machine Learning Research*, 2024. URL https://openreview.net/forum?id=gerNCVqqtR.

Tao Dai, Beiliang Wu, Peiyuan Liu, Naiqi Li, Jigang Bao, Yong Jiang, and Shu-Tao Xia. Periodicity Decoupling Framework for Long-term Series Forecasting. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/d298cf34e4539f9134db7f38b42f69fe-Paper-Conference.pdf.

Taesung Kim, Jinhee Kim, Yunwon Tae, Cheonbok Park, Jang-Ho Choi, and Jaegul Choo. Reversible Instance Normalization for Accurate Time-Series Forecasting against Distribution Shift. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=cGDAkQo1C0p.

Frederik Kratzert, Mathew Herrnegger, Daniel Klotz, Sepp Hochreiter, and Günter Klambauer. Neuralhydrology - interpreting lstms in hydrology. *CoRR*, abs/1903.07903, 2019. URL http://arxiv.org/abs/1903.07903.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, Alexander Merose, Stephan Hoyer, George Holland, Oriol Vinyals, Jacklynn Stott, Alexander Pritzel, Shakir Mohamed, and Peter Battaglia. Learning skillful medium-range global weather forecasting. *Science*, 382(6677):1416–1421, 2023. doi: 10.1126/science.adi2336. URL https://www.science.org/doi/abs/10.1126/science.adi2336.

Daesoo Lee, Sara Malacarne, and Erlend Aune. Vector quantized time series generation with a bidirectional prior model. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, volume 206 of *Proceedings of Machine Learning Research*, pp. 7665–7693. PMLR, 2023. URL https://proceedings.mlr.press/v206/lee23d.html.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. In *International Conference on Learning Representations*, 2018. URL https://openreview.net/forum?id=SJiHXGWAZ.

Peiyuan Liu, Beiliang Wu, Naiqi Li, Tao Dai, Fengmao Lei, Jigang Bao, Yong Jiang, and Shu-Tao Xia. WFTNet: Exploiting Global and Local Periodicity in Long-Term Time Series Forecasting. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 5960–5964, 2024a. doi: 10.1109/ICASSP48485.2024.10446883.

Yangyou Liu, Zezhi Shao, Xinyu Chen, Hu Chen, Fei Wang, and Yuankai Wu. PULSE: Generative Phase Evolution for Non-Stationary Time Series Forecasting. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=JJIqZzujgE.

Yong Liu, Haixu Wu, Jianmin Wang, and Mingsheng Long. Non-stationary Transformers: Exploring the Stationarity in Time Series Forecasting. In *Advances in Neural Information Processing Systems*, volume 35, pp. 9881–9893, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/4054556fcaa934b0bf76da52cf4f92cb-Abstract-Conference.html.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. itransformer: Inverted transformers are effective for time series forecasting. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=JePfAI8fah.

DongHao Luo and Xue Wang. ModernTCN: A Modern Pure Convolution Structure for General Time Series Analysis. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/86b1437c1e4c3b3c4debff98234a67e7-Paper-Conference.pdf.

Xiang Ma, Taihua Chen, Pengcheng Wang, Xuemei Li, and Caiming Zhang. ReCast: Reliability-aware codebook-assisted lightweight time series forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 24299–24307, 2026. doi: 10.1609/aaai.v40i29.39610. URL https://ojs.aaai.org/index.php/AAAI/article/view/39610.

Luca Masserano, Abdul Fatir Ansari, Boran Han, Xiyuan Zhang, Christos Faloutsos, Michael W. Mahoney, Andrew Gordon Wilson, Youngsuk Park, Syama Sundar Rangapuram, Danielle C. Maddix, and Bernie Wang. Enhancing foundation models for time series forecasting via wavelet-based tokenization. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=B6WalMoQJW.

Md Mahmuddun Nabi Murad, Mehmet Aktukmak, and Yasin Yilmaz. WPMixer: Efficient Multi-Resolution Mixing for Long-Term Time Series Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 19581–19588, 2025. doi: 10.1609/aaai.v39i18.34156. URL https://ojs.aaai.org/index.php/AAAI/article/view/34156.

Yuqi Nie, Nam H. Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*, 2023. URL https://arxiv.org/abs/2211.14730.

Yiming Niu, Jinliang Deng, and Yongxin Tong. PhaseFormer: From Patches to Phases for Efficient and Effective Time Series Forecasting. In *International Conference on Learning Representations*, pp. 8262–8285, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0dfa4b8a2eda4521992e2e1672001e2c-Paper-Conference.pdf.

Xiangfei Qiu, Xingjian Wu, Yan Lin, Chenjuan Guo, Jilin Hu, and Bin Yang. DUET: Dual Clustering Enhanced Multivariate Time Series Forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1185–1196. Association for Computing Machinery, 2025. doi: 10.1145/3690624.3709325.

Kashif Rasul, Young-Jin Park, Max Nihlén Ramström, and Kyung-Min Kim. VQ-AR: Vector quantized autoregressive probabilistic time series forecasting. *arXiv preprint arXiv:2205.15894*, 2022. URL https://arxiv.org/abs/2205.15894.

Sabera Talukder, Yisong Yue, and Georgia Gkioxari. TOTEM: TOkenized time series EMbeddings for general time series analysis. *Transactions on Machine Learning Research*, 2024. URL https://arxiv.org/abs/2402.16412.

Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural Discrete Representation Learning. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/7a98af17e63a0ac09ce2e96d03992fbc-Paper.pdf.

Shiyu Wang, Haixu Wu, Xiaoming Shi, Tengge Hu, Huakun Luo, Lintao Ma, James Zhang, and Jun Zhou. TimeMixer: Decomposable Multiscale Mixing for Time Series Forecasting. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/a7ac8a21e5a27e7ab31a5f42a0117bdb-Paper-Conference.pdf.

Haixu Wu, Jiehui Xu, Jianmin Wang, and Mingsheng Long. Autoformer: Decomposition Transformers with Auto-Correlation for Long-Term Series Forecasting. In *Advances in Neural Information Processing Systems*, volume 34, pp. 22419–22430, 2021. URL https://proceedings.neurips.cc/paper/2021/hash/bcc0d400288793e8bdcd7c19a8ac0c2b-Abstract.html.

Haixu Wu, Tengge Hu, Yong Liu, Hang Zhou, Jianmin Wang, and Mingsheng Long. TimesNet: Temporal 2D-Variation Modeling for General Time Series Analysis. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=ju_Uqw384Oq.

Zhijian Xu, Ailing Zeng, and Qiang Xu. FITS: Modeling Time Series with 10k Parameters. In *International Conference on Learning Representations*, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/701251e1db4a2e4dd2ef23f5265d5936-Paper-Conference.pdf.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2023. URL https://arxiv.org/abs/2205.13504.

Yunhao Zhang and Junchi Yan. Crossformer: Transformer Utilizing Cross-Dimension Dependency for Multivariate Time Series Forecasting. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=vSVLM2j9eie.

Yanjun Zhao, Tian Zhou, Chao Chen, Liang Sun, Yi Qian, and Rong Jin. Sparse-VQ transformer: An FFN-free framework with vector quantization for enhanced time series forecasting. *arXiv preprint arXiv:2402.05830*, 2024. URL https://arxiv.org/abs/2402.05830.

Haoyi Zhou, Shanghang Zhang, Jieqi Peng, Shuai Zhang, Jianxin Li, Hui Xiong, and Wancai Zhang. Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 11106–11115, 2021. doi: 10.1609/aaai.v35i12.17325. URL https://ojs.aaai.org/index.php/AAAI/article/view/17325.

Tian Zhou, Ziqing Ma, Xue Wang, Qingsong Wen, Liang Sun, Tao Yao, Wotao Yin, and Rong Jin. FiLM: Frequency improved Legendre Memory Model for Long-term Time Series Forecasting. In *Advances in Neural Information Processing Systems*, volume 35, pp. 12677–12690, 2022a. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/524ef58c2bd075775861234266e5e020-Abstract-Conference.html.

Tian Zhou, Ziqing Ma, Qingsong Wen, Xue Wang, Liang Sun, and Rong Jin. FEDformer: Frequency Enhanced Decomposed Transformer for Long-term Series Forecasting. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 27268–27286. PMLR, 2022b. URL https://proceedings.mlr.press/v162/zhou22g.html.

A Discussion, Limitations, and Future Directions

A.1 What Compositional Discretization Contributes

PatchVQFormer tests a specific alternative to both continuous regression and whole-patch quantization: a future patch is represented as a composition of band-limited waveform prototypes whose semantics remain fixed throughout predictor training. This factorization introduces a useful inductive bias when local structures recur at separable temporal scales. Orthogonal bands make the pre-quantization decomposition exact and the reconstruction error additive, while separate codebooks allow resolution to be assigned according to the needs of each spectral region. The forecaster can therefore reuse a low- frequency shape with different middle- and high-frequency components instead of requiring a distinct code for every complete waveform. Soft decoding also avoids committing to one prototype when the categorical prediction is uncertain, yet every output remains a weighted composition of frozen prototypes rather than the result of a continuous residual decoder.

The experiments support this mechanism most clearly on the ETT benchmarks. Across all 32 dataset–horizon settings, PatchVQFormer obtains an average MSE/MAE of $0.291/0.342$ and records 11 MSE and 8 MAE first-place results. More specifically, it has the lowest average ETT MSE among the listed methods at both $H = 336$ and $H = 720$. This is evidence that the discrete output constraint need not prevent accurate long-range extrapolation when the test series repeatedly recombines patterns represented in the training vocabulary. The evidence is not uniform, however: PatchVQFormer does not lead on ETTm1 for $H \geq 192$ and is worse than the strongest listed baseline at every Exchange horizon. The latter is the clearest failure case, although the experiments do not isolate whether its cause is distribution shift, insufficient codebook coverage, or another dataset-specific factor.

The ablations further qualify the interpretation. Removing Fourier decomposition or patching worsens the four-ETT average MSE from 0.3498 to 0.3664 and 0.3575, respectively, supporting the use of spectral factorization and local patch structure. Removing the codebook increases the average MSE/MAE to $0.3618/0.3965$, compared with $0.3498/0.3870$ for the full model. Thus, the complete model gives the best aggregate MSE and MAE, but its advantage is not uniform for every metric or dataset. The capacity study leads to a similarly specific conclusion: the band-dependent allocation gives the best MSE while a smaller uniform allocation gives the best MAE, and increasing the total number of prototypes does not monotonically improve forecasting. This does not conflict with the monotonic capacity–distortion result in Section 4.4, which concerns optimal training-set quantization rather than generalization of the downstream forecaster.

A.2 Limitations and Future Directions

The compositional vocabulary is also an information bottleneck. For each band, expected-prototype decoding is confined to the convex hull of waveforms seen during codebook construction. Rare transients, new amplitude–shape combinations, and changing periodicities may consequently be smoothed or mapped to inappropriate prototypes. Instance normalization restores the scale of a forecast but cannot recover within-window structure discarded by quantization. Moreover, fixed Fourier partitions can suffer from patch-boundary effects and assume that a single training-derived spectral organization is appropriate throughout the dataset. The independent band heads share a contextual state and sequence loss, but they do not explicitly model a joint distribution over band combinations. Their marginal expectations can therefore produce a plausible average waveform while obscuring multimodal futures.

The theoretical results should be read within the same scope. Orthogonality guarantees exact decomposition before quantization and additive distortion after band-wise reconstruction; it does not guarantee low distortion for a finite vocabulary. The assignment-margin result certifies invariance of a nearest hard assignment only under a local, band-bounded perturbation; the implemented distance-weighted Top-K retrieval can still vary within this neighborhood, and the result does not establish robustness of the final forecast to distribution shift. Similarly, the number of possible code compositions is a representational upper bound, not evidence that all combinations are observed, learnable, or physically valid. A finite vocabulary by itself also provides no guarantee of denoising, causal identification, calibrated uncertainty, or lower computational cost.

The empirical scope remains limited despite the complete 32-setting main table. The available baseline runs use different input lengths, so the table is a literature comparison rather than a controlled state-of-the-art ranking. Component ablations cover only the four ETT datasets, codebook-capacity sensitivity is reported only for one ETTh2 setting, and the number-of-bands study covers only ETTm2. Robustness under noise, rare events, and distribution shift, as well as end-to-end time, memory, offline clustering cost, and multiple-seed variability, also remain to be measured. In addition, the test-informed input-length selection path identified in App. B must be excluded from strictly held-out comparisons. These evaluations are necessary before making robustness or efficiency claims beyond the reported benchmark settings.

Promising extensions follow directly from these limitations. Adaptive or multiresolution partitions could accommodate time-varying spectra; hierarchical or expandable discrete vocabularies could reserve capacity for

rare regimes; and structured prediction over cross-band code tuples could capture dependencies that independent heads represent only implicitly. Sampling and calibration of the predicted categorical distributions would turn the current point forecaster into an uncertainty-aware model. Finally, monitoring assignment margins and reconstruction error at inference time may provide a practical signal for detecting vocabulary mismatch and deciding when a codebook should be updated.

B Notation and Implementation Details

This appendix records the defaults of the current multi-dataset PatchVQFormer experiment scripts. Environment variables and `EXTRA_RUN_ARGS` take precedence over these defaults, so final tables must be generated from the effective arguments stored in the corresponding run logs.

B.1 Notation and Tensor Shapes

Symbol	Meaning
L, H, C	History length, forecast length, variable count
P, S, N, J	Patch length, stride, historical and future patch counts
M, K_m	Number of bands and band-specific codebook size
Π_m	Orthogonal projection onto band m
$\mathcal{C}_m$	Frozen waveform prototype table
$\mathcal{E}_m$	Prototype-indexed embedding table, frozen or trainable by configuration
$k^{(m)}(p)$	Nearest-prototype assignment in band m
$\mathcal{S}_\theta, \mathcal{T}_\phi$	Spatial and masked temporal operators

B.2 Data Processing and Frequency Masks

Splits and normalization. All experiments use multivariate inputs (`features=M`), with $L \in \{96, 336, 512, 720\}$, $H \in \{96, 192, 336, 720\}$, and `label_len`$= 48$. ETTh1 and ETTh2 use the first $8{,}640$ observations for training, the next $2{,}880$ for validation, and the following $2{,}880$ for testing. ETTm1 and ETTm2 use $34{,}560/11{,}520/11{,}520$ observations. Electricity, Exchange, Weather, and Solar use the first 70% for training, approximately the next 10% for validation, and the final 20% for testing. Validation and test dataset objects prepend the preceding L observations when constructing a historical input; their forecast targets remain entirely within the corresponding partition.

A per-variable `StandardScaler` is fitted only on the training segment. Within each globally standardized sample, the model then computes historical window statistics

$$\mu_c = \frac{1}{L}\sum_{t=1}^{L} x_{t,c}, \qquad \sigma_c = \sqrt{\frac{1}{L}\sum_{t=1}^{L}(x_{t,c}-\mu_c)^2 + 10^{-5}}, \qquad \bar{x}_{t,c} = \frac{x_{t,c}-\mu_c}{\sigma_c}. \tag{6}$$

The same historical μ_c and σ_c normalize the forecast target, and the model output is transformed back to the globally standardized space before evaluation. The default is `INVERSE=0`; consequently, the reported MSE and MAE are in globally standardized units rather than the original physical units.

Patch indexing and overlap-add. For a configured patch length P_0, the effective historical patch length is $P = \min(P_0, L)$ and the stride is $S = \max(1, \lfloor P/2 \rfloor)$. Complete patches start at $0, S, 2S, \ldots$; if the last such patch does not reach the window endpoint, an additional patch starts at $L - P$. The forecast interval is patched analogously using H. Thus, the main configurations do not require right padding, although the implementation retains a zero-padding branch for sequences shorter than a requested patch. Overlapping decoded patches are summed and divided pointwise by their coverage counts.

FFT convention and masks. The real FFT is applied independently to every instance-normalized patch. NumPy/PyTorch's default convention is used: the forward `rfft` is unnormalized and `irfft(..., n=P)` includes the inverse factor $1/P$. Each band retains a closed contiguous interval of one-sided frequency bins and zeros all other bins before the inverse transform. The negative-frequency conjugates required for a real signal are supplied implicitly by `irfft`. DC belongs to the first band, and the Nyquist bin for even P belongs to the final band. The band components therefore sum to the input patch up to floating-point error before quantization.

Frequency-mask discovery. Mask statistics use at most $10{,}000$ distinct training-only window–variable–patch coordinates sampled with seed 2026. For one-sided bin k, the normalized energy is

$$\rho_k = \frac{\mathbb{E}_p[|\,\mathrm{rFFT}(p)_k|^2]}{\sum_r \mathbb{E}_p[|\,\mathrm{rFFT}(p)_r|^2]}. \tag{7}$$

Non-endpoint bins are not multiplied by a conjugate multiplicity of two. For $k > 0$, the lag is $\ell_k = \mathrm{round}(P/k)$, and q_k is the nonnegative part of the mean lag-ℓ_k product over the complete patch. The score is $s_k = \rho_k(q_k + 0.1)$, while $s_0 = \rho_0$. The deterministic partition begins with bin 9 as the initial high-frequency-tail location, uses a 0.85 cumulative non-DC energy threshold, retains local maxima with $s_k > 0.02$ when feasible, and completes the requested number of bands with ordered contiguous groups. These boundaries are heuristic and are not optimized on the validation set.

Leakage audit and model-selection scope. The scaler, frequency-mask discovery, and prototype fitting consume only the training partition. The validation set selects the Stage 2 checkpoint, and the test set is used for final metric computation. This code-path separation does not by itself establish that every reported comparison is free from test-set selection: the repository also contains a baseline-selection script that chooses the best input length by test MSE, and some configuration comments record test-informed tuning. Any result using this "best input length" protocol must state that selection rule explicitly; a strictly held-out report must instead fix the input length without consulting test performance.

B.3 Clustering, Capacity Selection, and Search

Prototype fitting. For each dataset–horizon configuration, the implemented construction is:

1. Extract length-L windows from the globally standardized training segment, apply Equation 6, and form length-P patches.
2. Regard $(T_{\mathrm{train}} - L + 1)CN$ window–variable–patch triples as the candidate population. Draw at most 100,000 coordinates with replacement. All bands share these coordinates, after which each band receives its own band-pass time-domain component.
3. Fit one independent codebook of size K_m per band. The default `STAGE1_DEVICE=cuda` path uses full-batch Lloyd k-means with squared Euclidean distance, k-means++ initialization, three initializations, 200 maximum iterations, and seed 2026. The initialization with the lowest mean squared assignment distance is retained.
4. Compute assignments in blocks of $\max(\texttt{CLUSTER_BATCH_SIZE}, 65536)$; the default is therefore 65,536, and `CLUSTER_BATCH_SIZE=4096` does not imply mini-batch k-means. Stop when labels do not change or when the total squared center displacement is at most $\max(10^{-4}\sum_j \mathrm{Var}(x_j), 10^{-8})$. Reinitialize an empty cluster with the sample farthest from the current centers.
5. Store both samples and centers as float32 vectors in $\mathbb{R}^P$. The selected centers become float32 model buffers, assignments are integer indices, and the prototype buffers remain frozen throughout Stage 2.

The alternative CPU path invokes scikit-learn `KMeans`; its precise termination and empty-cluster behavior are version dependent and should not be conflated with the CUDA implementation above.

Capacity configurations. The main harmonic runs fit the configured capacity vector or the single automatically assigned vector; they do not sweep `CLUSTER_CAPACITY_CANDIDATES=64` on validation data.

Table 4 lists the script defaults using the notation $P/M/(K_1, \ldots, K_M)/T$, where T is the test-time soft-decoding temperature. The effective P is still truncated when $P > L$.

B.4 Waveform-Based Embedding Initialization

Each frozen codebook $\mathcal{C}_m \in \mathbb{R}^{K_m \times P}$ has a prototype-indexed embedding table $\mathcal{E}_m \in \mathbb{R}^{K_m \times d_{\mathrm{model}}}$. Initialization first zeros the table and then copies the first $\min(P, d_{\mathrm{model}})$ time-domain samples of each prototype into the corresponding embedding row. If $d_{\mathrm{model}} > P$, the remaining coordinates are sampled independently from $\mathcal{N}(0, 0.02^2)$; if $P > d_{\mathrm{model}}$, the waveform is truncated to its first d_{model} samples. There is no learned initialization projection or additional normalization.

Whether these embeddings subsequently update is configuration dependent. The default is `FREEZE_TOKEN_EMBEDDINGS=1`; only the ETTm2 configurations with $H \in \{192, 336, 720\}$ override it to zero. When trainable, the embeddings are optimized by Adam together with the backbone. No penalty keeps them close to the original prototypes, enforces unit norm, or projects them back to a waveform subspace. The frozen prototype and its embedding nevertheless retain the same row index throughout training.

B.5 Backbone, Objective, and Inference Pseudocode

Tensor flow. For batch size B, variable count C, N historical patches, J future patches, and hidden width d, a band-m codebook has shape $K_m \times P$ and its embedding table has shape $K_m \times d$. In the current main-

Table 4: Default patch, band, codebook-capacity, and decoding-temperature profiles in the current experiment scripts. "Auto" denotes the deterministic capacity rule above, not a validation search.

Dataset	H	P	M	$(K_1, \ldots, K_M)$	T
ETTh1	96	96	6	$(512, 256, 32, 64, 128, 256)$	1.1
	192	96	6	$(128, 512, 16, 32, 8, 8)$	1.1
	336	336	6	$(512, 256, 32, 64, 128, 256)$	1.1
	720	720	4	$(256, 512, 64, 16)$	1.1
ETTh2	96	96	8	$(512, 256, 16, 32, 16, 16, 64, 16)$	1.2
	192	128	6	$(512, 256, 64, 16, 32, 16)$	1.2
	336	336	6	$(512, 256, 32, 64, 128, 256)$	1.3
	720	720	8	$(256, 32, 128, 8, 8, 8, 16, 8)$	1.1
ETTm1	96	96	6	$(512, 256, 32, 64, 128, 256)$	1.2
	192	96	8	$(256, 64, 32, 8, 8, 8, 16, 8)$	1.3
	336	336	6	$(512, 256, 32, 64, 128, 256)$	1.1
	720	576	8	$(128, 32, 8, 256, 8, 16, 8, 8)$	1.1
ETTm2	96	96	6	$(512, 256, 32, 64, 128, 256)$	1.1
	192	96	6	$(128, 512, 16, 32, 8, 8)$	1.1
	336	96	6	AUTO	1.1
	720	96	6	AUTO	1.1
Exchange	96	32	6	AUTO	1.1
	192	32	6	AUTO	1.1
	336	24	6	AUTO	1.1
	720	96	4	$(256, 512, 64, 16)$	1.1
Weather	96	96	6	$(512, 256, 32, 64, 128, 256)$	1.1
	192	48	6	$(256, 32, 16, 8, 64, 8)$	1.1
	336	96	4	$(256, 64, 32, 8)$	1.1
	720	240	6	$(256, 128, 32, 8, 16, 8)$	1.1
Electricity	96	48	6	$(256, 512, 128, 128, 1024, 2048)$	1.0
	192	48	6	$(256, 512, 128, 128, 256, 512)$	1.1
	336	96	6	$(512, 256, 128, 256, 512, 2048)$	1.0
	720	336	6	$(512, 256, 128, 128, 256, 1024)$	1.0
Solar	96	96	6	$(256, 128, 16, 8, 32, 8)$	0.9
	192	96	6	$(512, 512, 64, 32, 32, 32)$	1.0
	336	144	6	$(2048, 256, 32, 16, 64, 16)$	1.1
	720	288	6	$(64, 256, 8, 128, 16, 8)$	0.9

experiment implementation, a historical band component retrieves a distance-weighted mixture of its nearest embeddings rather than only the single hard assignment. For its nearest-neighbor set $\mathcal{N}_m(p)$,

$$\alpha_{m,k}(p) = \frac{\exp(-\|\Pi_m p - c_k^{(m)}\|_2^2)}{\sum_{r \in \mathcal{N}_m(p)} \exp(-\|\Pi_m p - c_r^{(m)}\|_2^2)}, \qquad h(p) = \sum_m \sum_{k \in \mathcal{N}_m(p)} \alpha_{m,k}(p) e_k^{(m)}. \tag{8}$$

The default neighborhood size is four; selected ETTh1 cells use eight or two. Summed band embeddings form a tensor in $\mathbb{R}^{B \times C \times N \times d}$. Position and variable embeddings, together with a patch-start hour embedding when enabled, are added before concatenating $B \times C \times J \times d$ future queries.

The default backbone applies a variable-axis Transformer followed by a patch-axis temporal Transformer. In the implemented future mask, a future query can attend to all historical patches and to itself, but not to other future queries. For each band, an identity-initialized, trainable $d \to d$ output projection is followed by LayerNorm and a $d \to K_m$ classification head.

Objective. Let CE_m be the cross-entropy averaged over the batch, variables, and future patches for band m. Let C_m be the average cosine similarity between the ground-truth prototype and the untempered expected prototype obtained from the band logits. With first differences denoted by Δ, the implemented loss is

$$\mathcal{L} = \sum_{m=1}^{M} w_m \left[CE_m + 2(1 - C_m)\right] + \lambda_{\text{seq}} \operatorname{MSE}(\widehat{Y}, \bar{Y}) + \lambda_{\text{diff}} \operatorname{MSE}(\Delta\widehat{Y}, \Delta\bar{Y}). \tag{9}$$

The defaults are $w_m = 1$, $\lambda_{\text{seq}} = 8$, and $\lambda_{\text{diff}} = 2$. ETTm2 at $H = 96$ uses $\lambda_{\text{diff}} = 4$; ETTh2 at $H = 192$ and $H = 336$ uses $\lambda_{\text{seq}} = 16$ and 12, respectively. Selected ETTh2 and ETTm2 cells also use nonuniform w_m. Although the configuration exposes `RECON_LOSS_WEIGHT=2`, this value is not consumed by the harmonic loss path in Equation 9.

Training. Freeze the prototype buffers; normalize the history and target using the historical statistics; assign each target band component to its nearest prototype for the cross-entropy label; retrieve Top-K historical embeddings; and produce all future logits in one forward pass. A temperature-one softmax forms each expected prototype during training. The bands are summed, coverage-normalized overlap-add forms $\widehat{Y}$, and Equation 9 is backpropagated through only the parameters enabled by the configuration.

Inference. Only historical values and known time markers are provided. The model repeats historical normalization and Top-K retrieval and predicts all band logits in parallel. The main soft forecast applies the cell-specific temperature T in Table 4 to a full-vocabulary softmax, takes the corresponding expectation over frozen prototypes, sums the bands, applies coverage-normalized overlap-add, and restores the historical scale to the globally standardized space. The evaluation code additionally records argmax hard decoding, Top-K decoding, and an oracle reconstruction using true future indices. The oracle is diagnostic only and is not a forecast.

B.6 Training and Evaluation Configuration

The default backbone uses $d_{\text{model}} = 128$, four attention heads, one spatial layer, two temporal layers, $d_{\text{ff}} = 512$, and Transformer dropout 0.2. ETTm1 at $H = 336$ overrides the widths to $d_{\text{model}} = 384$ and $d_{\text{ff}} = 1536$. Stage 2 uses PyTorch Adam with learning rate 10^{-4} and weight decay 10^{-4} for at most 30 epochs, normally with patience 10. ETTm2 at $H = 96$ uses 60 epochs and patience 20. Gradient clipping is disabled by default, and the random seed is 2026. Patch-start hour embeddings are controlled by the dataset configuration; the full calendar embedding is disabled in all current profiles.

Default batch size is 64 for the ETT datasets, Weather, and Exchange; 32 for Electricity and Solar. ETTh1 at $H \in \{336, 720\}$ overrides the batch size to 128, while ETTm2 at $H = 96$ uses 32. All reported experiments were conducted on an NVIDIA L40S GPU with 48 GB of memory.

Checkpoint selection uses the validation-set mean band loss, comprising the cross-entropy and cosine terms in Equation 9, rather than the complete sequence objective. A checkpoint is saved only after a strict improvement; nonfinite values are ignored, and training stops after the configured number of consecutive non-improvements. The validation loader shuffles and normally uses `drop_last=True`. The test loader does not shuffle but also normally drops the final incomplete batch, except when the entire split is smaller than one batch. These loader settings are therefore part of the metric-reproduction protocol.

C Proofs and Scope of the Analysis

C.1 Orthogonal Band Projections

From unitarity $FF^* = I$ and the masks in Equation 1,

$$\Pi_m^* = \Pi_m, \qquad \Pi_m \Pi_n = F^* B_m B_n F, \qquad \sum_m \Pi_m = I. \tag{10}$$

The projections are self-adjoint and idempotent, with mutually orthogonal ranges. Conjugate-symmetric masks preserve real-valued patches. This does not imply that extracting a subwindow after a longer-window projection preserves orthogonality.

C.2 Additive Reconstruction Distortion

Let $r_m = \Pi_m p - v_m$ for any $v_m \in \text{range}(\Pi_m)$, including hard prototypes and soft expectations of prototypes from band m. Then $r_m \in \text{range}(\Pi_m)$ and $\langle r_m, r_n \rangle = 0$ for $m \neq n$. Expanding the norm gives

$$\left\| \sum_m r_m \right\|_2^2 = \sum_m \|r_m\|_2^2 + 2 \sum_{m<n} \langle r_m, r_n \rangle = \sum_m \|r_m\|_2^2, \tag{11}$$

which proves the stated additive distortion identity. Euclidean cluster means and their convex combinations remain in the corresponding linear subspace.

C.3 A Sufficient Nearest-Prototype Stability Condition

Write $\delta = \Pi_m \epsilon$. For any $b \neq a$, the triangle and reverse triangle inequalities imply

$$\begin{aligned} &\|u + \delta - c_b^{(m)}\|_2 - \|u + \delta - c_a^{(m)}\|_2 \\ &\quad \geq \|u - c_b^{(m)}\|_2 - \|u - c_a^{(m)}\|_2 - 2\|\delta\|_2 \\ &\quad \geq \Delta_m(u) - 2\|\delta\|_2. \end{aligned} \tag{12}$$

This difference is positive if $\|\delta\|_2 < \Delta_m(u)/2$, so the nearest prototype remains a. A single-codeword band is trivially invariant but can be highly lossy. Neither case proves proximity to an unobserved clean signal or improved prediction.

C.4 Assumptions, Counterexamples, and Numerical Checks

For a concrete numerical check, take a length-four patch $p = (1, 2, 3, 4)$ and partition its real-FFT bins into $B_1 = \{0, 1\}$ and $B_2 = \{2\}$. The inverse transforms give $p_1 = (1.5, 1.5, 3.5, 3.5)$ and $p_2 = (-0.5, 0.5, -0.5, 0.5)$. Thus, $p_1 + p_2 = p$ and $\langle p_1, p_2 \rangle = 0$. Choose band-valid prototypes $c_1 = (2, 2, 3, 3)$ and $c_2 = (-1, 1, -1, 1)$. Direct application of the masks gives $\Pi_1 c_1 = c_1$ and $\Pi_2 c_2 = c_2$. The residuals $r_m = p_m - c_m$ are orthogonal, with $\|r_1\|_2^2 = \|r_2\|_2^2 = 1$; correspondingly, $\|p - (c_1 + c_2)\|_2^2 = 2 = \sum_m \|r_m\|_2^2$. In exact arithmetic, Euclidean cluster means of band-projected samples remain in that band's linear subspace; finite-precision implementations should check $\|c - \Pi_m c\|_2$ against a stated numerical tolerance.

The nearest-prototype margin is sufficient, not universal. For unit band vector v, prototypes $-v$ and v, and $u = -\delta v$ with $0 < \delta < 1$, the distance margin is 2δ. A perturbation $2\delta v$ moves u across the assignment boundary; the required perturbation can be arbitrarily small as $\delta \to 0$. Conversely, a high-frequency component $x_t = (-1)^t$ lies at the Nyquist frequency yet is perfectly predictable from its previous value. High frequency therefore does not, by itself, imply noise or justify a small vocabulary. These constructions check the scope of the formulation; they do not establish numerical tolerances for a fitted codebook or forecasting robustness.

D Additional Results and Diagnostics

D.1 Full Ablation Results by Forecasting Horizon

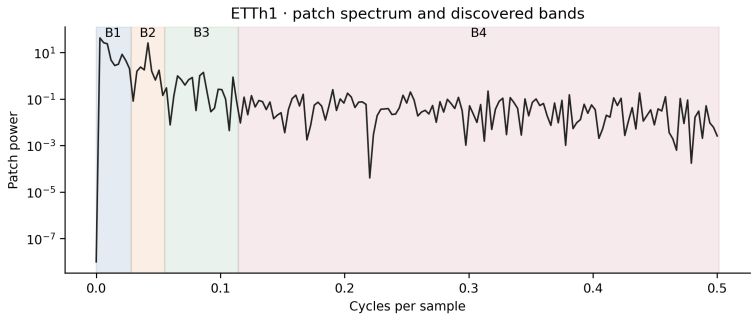
Table 5 reports the complete contents of the ablation summary across the four ETT benchmarks. All 16 configurations use an input length of 336. The four horizon-specific rows retain the individual measurements, and the final row for each dataset reports the mean across the four forecasting horizons.

D.2 Frequency-Band Prototype Visualizations and Code Usage

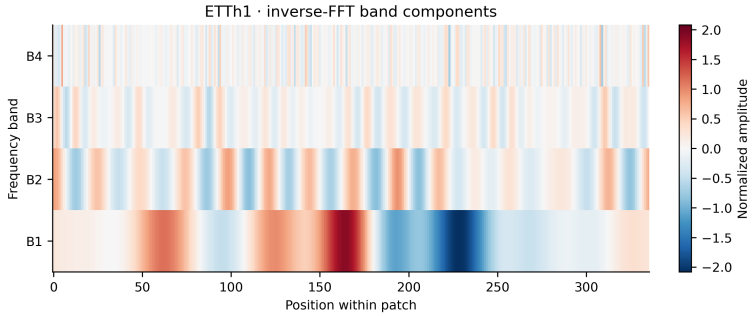
Visualization protocol. We analyze the ETTh1 main-experiment checkpoint for the 720-step forecasting task without retraining the model. The Fourier example uses the last length-336 historical patch of the OT variable in a representative test window. The patch is normalized by the mean and standard deviation of its input window before applying the real FFT. The four contiguous frequency bands, discovered using training data only, contain bins 0–9, 10–18, 19–38, and 39 through the Nyquist bin. For the prototype analysis, every test window–variable pair contributes its middle historical patch. Within each band, we accumulate the distance-weighted Top-4 encoding probabilities and display the three prototypes with the largest total weights. The percentages in the legends are therefore soft usage shares rather than hard nearest-neighbor assignment frequencies.

Frequency partition and reconstruction. Figure 5 traces one patch from its spectrum to its band-wise time-domain representation. The training-derived boundaries separate the slowly varying B1 component from progressively faster oscillations in B2–B4. Because the masks are complementary, summing the four inverse-FFT components recovers the normalized input up to numerical precision; this equality verifies the decomposition rather than forecasting accuracy. Replacing each component by its nearest frozen prototype produces the orange curve in Figure 5(c). The resulting codebook reconstruction preserves the patch's principal low-frequency shape while expressing it entirely through four discrete, band-specific waveforms, illustrating the distinction between lossless spectral decomposition and the subsequent finite-vocabulary approximation.

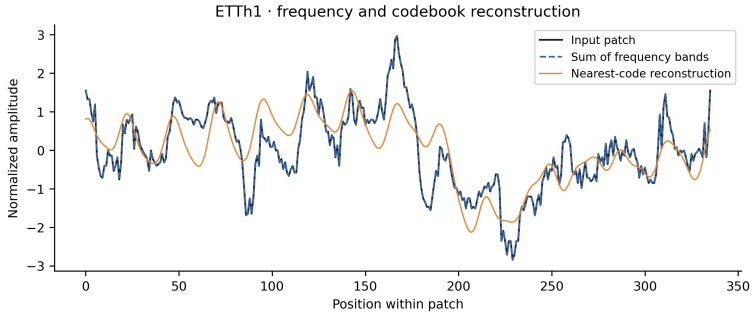
Band-specific waveform vocabularies. Figure 7 shows that the frozen codebooks specialize to distinct temporal scales. The B1 prototypes capture smooth level changes and broad turning patterns. B2 contains medium-frequency oscillations with different phases and amplitude envelopes, while B3 provides denser, more regular periodic patterns. B4 represents the fastest and lowest-amplitude fluctuations, including a frequently used near-zero high-frequency component. Thus, a forecast patch is not restricted to one whole-patch template: it can combine one waveform from each specialized vocabulary to form a new cross-band composition.



(a) Patch spectrum and training-derived frequency bands.

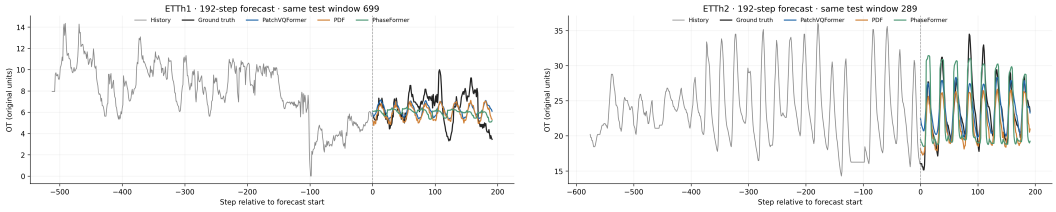


(b) Inverse-FFT components of the four bands.



(c) Exact band sum and nearest-prototype reconstruction.

Figure 5: Fourier decomposition and codebook reconstruction of an ETTh1 patch. The bands sum exactly before quantization; the nearest-prototype curve shows the discrete approximation.

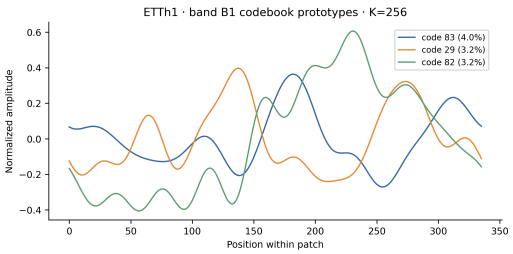


(a) ETTh1, test window 699. **(b) ETTh2, test window 289.**

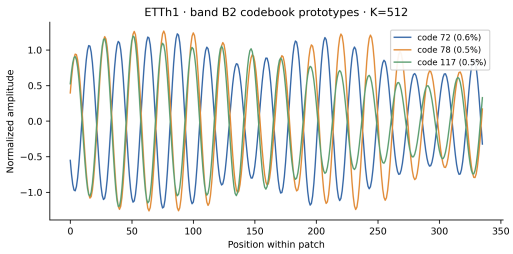
Figure 6: 192-step OT forecasts on ETTh1 and ETTh2, showing shared history, ground truth, and predictions from PatchVQFormer, PDF, and PhaseFormer. Dashed lines mark the forecast start.

Table 5: Complete ETT ablation results for $H \in \{96, 192, 336, 720\}$ using input length 336. Lower is better; best and second-best entries are shown in bold and underline.

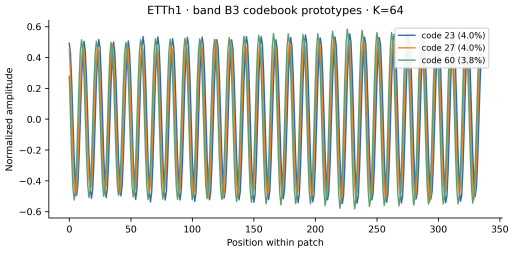
Models		Full		w/o Fourier		w/o Codebook		w/o Patch	
Data	H	MSE	MAE	MSE	MAE	MSE	MAE	MSE	MAE
ETTh1	96	0.4074	0.4234	0.4362	0.4498	**0.4002**	**0.4199**	0.4134	0.4261
	192	0.4323	0.4457	0.4610	0.4666	**0.4319**	**0.4358**	0.4419	0.4504
	336	**0.4283**	**0.4439**	0.4699	0.4748	0.4386	0.4465	0.4321	0.4482
	720	**0.4395**	**0.4682**	0.4524	0.4777	0.4598	0.4717	0.4849	0.5048
	Avg.	**0.4269**	0.4453	0.4549	0.4672	0.4326	**0.4435**	0.4431	0.4574
ETTh2	96	0.3162	0.3677	0.3298	0.3777	**0.3034**	**0.3612**	0.3212	0.3741
	192	0.3637	0.3979	0.3730	0.4050	0.3869	0.4199	**0.3607**	**0.3967**
	336	**0.3496**	**0.4010**	0.3591	0.4074	0.3693	0.4128	0.3593	0.4117
	720	**0.3947**	**0.4304**	0.4022	0.4356	0.4177	0.4490	0.4433	0.4673
	Avg.	**0.3560**	**0.3993**	0.3660	0.4064	0.3693	0.4107	0.3711	0.4125
ETTm1	96	**0.2907**	**0.3487**	0.3112	0.3655	0.2983	0.3560	0.3037	0.3600
	192	**0.3326**	**0.3732**	0.3541	0.3900	0.3450	0.3846	0.3359	0.3801
	336	0.3747	0.3998	0.3947	0.4151	0.3812	0.4050	**0.3592**	**0.3919**
	720	0.4371	0.4404	0.4390	0.4419	0.4463	0.4450	**0.4332**	**0.4365**
	Avg.	0.3588	**0.3906**	0.3748	0.4031	0.3677	0.3977	**0.3580**	0.3921
ETTm2	96	**0.1658**	**0.2531**	0.1812	0.2657	0.1780	0.2681	0.1720	0.2588
	192	**0.2226**	**0.2922**	0.2305	0.2972	0.2392	0.3107	0.2300	0.2979
	336	**0.2749**	**0.3245**	0.2913	0.3369	0.3004	0.3500	0.2770	0.3274
	720	0.3661	0.3814	0.3770	0.3893	0.3933	0.4081	**0.3515**	**0.3761**
	Avg.	**0.2574**	**0.3128**	0.2700	0.3223	0.2777	0.3342	0.2576	0.3150



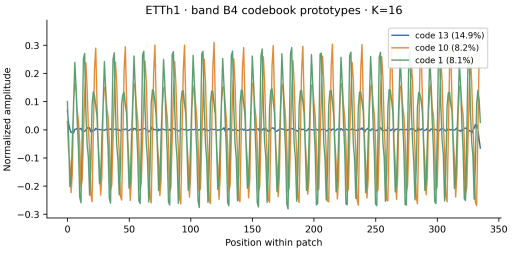
(a) B1, $K_1 = 256$**.**



(b) B2, $K_2 = 512$**.**



(c) B3, $K_3 = 64$**.**



(d) B4, $K_4 = 16$**.**

Figure 7: Most-used prototypes in each ETTh1 frequency band. Axes show patch position and normalized amplitude; legend percentages report distance-weighted Top-4 usage over the test windows.

Code-usage concentration. The displayed Top-3 prototypes account for 10.4%, 1.6%, 11.8%, and 31.2% of the weighted encoding mass in B1 through B4, respectively. Usage is therefore most diffuse in the large B2 vocabulary and most concentrated in the compact B4 vocabulary. This band-dependent pattern is consistent with allocating different capacities to different spectral regions instead of forcing a uniform vocabulary size. At the same time, even the concentrated B4 band uses multiple distinct high-frequency shapes, showing that the smallest codebook does not reduce the band to a single fixed residual pattern.

Additional 192-step forecasting cases. Figure 6 complements the ETTm1 case in Figure 4 with ETTh1 and ETTh2 examples using the same historical window and OT values restored to their original units. PatchVQ-Former captures the recurring short-period variation on ETTh1 and the timing of pronounced periodic cycles on ETTh2, illustrating how the methods extrapolate different combinations of local oscillation and level variation.