DIVINE: SIMPLE CROSS-MARKET STOCK PRETRAINING VIA DIVERSE INDICATOR RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Financial time-series pretraining typically learns from masked observations, contrastive relations, or future outcomes—yet existing objectives struggle to simultaneously avoid future-supervision uncertainty and maintain return-prediction alignment. We propose DIVINE (DIVerse INdicator rEconstruction), a simple cross-market pretraining framework that reconstructs technical indicators from raw OHLCV history. Computed from observed price-volume history, technical indicators provide consistently defined supervision across markets while summarizing diverse market dynamics with established relevance to return prediction. Pretrained jointly on six-equity market datasets, DIVINE reconstructs 77 targets derived from 16 standard indicators and transfers only the learned encoder to downstream stock ranking. Across all six markets, DIVINE achieves the strongest average portfolio performance with a lightweight 0.05M-parameter encoder, outperforming pretraining baselines and matching or exceeding substantially larger financial foundation models, while remaining robust and data-efficient. Systematic analyses show that indicator diversity and market diversity provide complementary gains in transfer. Together, these results suggest that supervision design and cross-market diversity—rather than model scale—are the key drivers of strong, transferable financial representations.

1 INTRODUCTION

Stock return prediction is a long-standing challenge in quantitative finance, particularly in cross-sectional settings where models must rank assets by expected future returns under highly non-stationary market dynamics (Azariadis & Smith, 1998; Avramov, 2002; Hu et al., 2025) and low signal-to-noise conditions (Rapach et al., 2010; Gu et al., 2020; Caner & Daniele, 2025). A key difficulty is that equity markets differ in their cross-sectional and temporal structure, which strongly shapes return predictability (Feng et al., 2019; Hsu et al., 2021; Li et al., 2024). Models trained on a single market may therefore specialize to its market-specific regularities, limiting their ability to transfer across regimes and markets. Learning reusable representations across markets is thus appealing: it can capture recurring price-volume structure across financial systems while reducing dependence on labeled data from any single market.

Pretraining offers a natural route toward such transferable representations, but its effectiveness depends critically on the supervision objective. Existing approaches largely reconstruct masked prices (He et al., 2022; Nie et al., 2022; Xia et al., 2024; Chen et al., 2025; Wang et al., 2025), learn contrastive relations between sequences (Yue et al., 2022; Liu & Chen, 2024; Lee et al., 2026), predict future prices autoregressively (Ding et al., 2025; Zhu et al., 2025; Shi et al., 2025), or solve stock-specific pretext tasks (Wang et al., 2025). While effective, existing objectives often expose a tension between *future-supervision uncertainty* and *return-prediction alignment*. Objectives based on future prices or returns are naturally aligned with prediction, but their targets depend on *stochastic* future market realizations. By contrast, objectives constructed solely from past observations avoid this source of uncertainty, but offer no explicit link between the pretraining objective and future-return prediction. This raises a central question: ***what form of past-only supervision can be directly aligned with future-return prediction?***

We explore technical indicators as such a supervision signal. Computed purely from past price-volume observations, they summarize diverse aspects of market dynamics—including trend,

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.

volatility, momentum, relative price positioning, directional movement, and price-volume interaction (Wilder, 1978; Lambert, 1980; Murphy, 1999; Appel, 2005)—with definitions that remain consistent across markets. Such price-volume transformations have long served as building blocks for predictive alpha factors in quantitative finance (Kakushadze, 2016; Yang et al., 2020; Fang et al., 2020; Feng et al., 2020), grounding their relevance to future-return prediction and making them natural candidates for return-aligned supervision. We therefore ask whether reconstructing a diverse indicator space can exploit this predictive structure to learn representations that transfer beyond the indicator targets themselves.

We propose **DIVINE** (DIVerse INdicator rEconstruction), a simple cross-market pretraining framework using diverse technical-indicator reconstruction. Given raw OHLCV sequences, DIVINE jointly reconstructs 77 targets derived from 16 standard technical indicators across multiple period configurations. All targets are reconstructed from a shared latent representation, encouraging the encoder to capture price-volume structure that supports diverse market summaries. We jointly pretrain the encoder on six international equity markets, discard the reconstruction head, and transfer only the learned encoder to cross-sectional return ranking in each target market. Technical indicators are thus used only during pretraining and are not required as downstream inputs. We summarize our contributions as follows:

- **Technical-indicator reconstruction for representation learning.** We introduce DIVINE, a cross-market pretraining framework that reconstructs a diverse space of technical indicators computed from past OHLCV history to learn transferable financial representations. Rather than using technical indicators directly as downstream features, DIVINE uses them only as pretraining targets to shape a latent representation that transfers to stock ranking.
- **Strong performance with lightweight and data-efficient pretraining.** Across six equity markets, DIVINE achieves the strongest average portfolio performance with a lightweight 0.05M-parameter encoder, outperforming pretraining methods spanning diverse supervision paradigms, and matching or exceeding financial foundation models pretrained on substantially larger external corpora with approximately $490\times$ fewer parameters. It further remains robust across encoder architectures and downstream settings. Beyond strong portfolio performance, DIVINE is remarkably efficient: it surpasses full-data pretraining baselines using only 10% of the pretraining corpus, and exceeds full-data training from scratch with only 5% of downstream labels.
- **Indicator and market diversity jointly drive transfer.** Through systematic ablations and transfer analyses, we show that the learned representation outperforms both direct indicator features and the reconstructed indicators themselves. Broader indicator type coverage contributes substantially more than adding period variants, while even pretraining without the target market still recovers 64% of the single-market-to-joint improvement. These results establish indicator diversity and market diversity as complementary drivers of transferable financial representations.

2 Related Work

Supervised Stock Ranking and Prediction. Deep models for stock ranking span recurrent architectures that optimize investment returns directly rather than price regression (Feng et al., 2018; 2019), models that exploit inter-stock structure (Yoo et al., 2021; Hsu et al., 2021; Li et al., 2024), mixture-of-experts frameworks (Sun et al., 2023; Chen & Wang, 2025), and lightweight architectures that mix information across indicator, temporal, and stock dimensions (Fan & Shen, 2024). Most of these methods are trained from scratch on individual markets. QuantNet (Koshiyama et al., 2021) instead couples market-specific encoders through a shared bottleneck, but optimizes it end-to-end for trading signals rather than explicitly pretraining a transferable representation.

Financial Time-Series Pretraining. Existing pretraining methods can be broadly organized by how their supervision relates historical observations to future returns. **(1) Masked price reconstruction** adapts masked autoencoding (He et al., 2022; Nie et al., 2022) to stock sequences, including variants using dynamic hypergraphs (Sawhney et al., 2021; Xia et al., 2024) or retrieval-augmented expert routing (Chen et al., 2025). Such objectives avoid future-outcome supervision, but reconstruction of raw missing observations is only indirectly related to future-return prediction and can be sensitive to short-term variation. SSPT (Wang et al., 2025) instead reconstructs moving averages over masked windows, partially mitigating such variation. **(2) Contrastive learning** defines posi-

tive and negative relations through temporal augmentation (Yue et al., 2022; Liu & Chen, 2024) or cross-stock future-return correlation (Lee et al., 2026). The former is only indirectly aligned with future returns, whereas the latter relies on future outcomes and thus inherits uncertainty from their stochastic realizations. **(3) Autoregressive prediction** predicts future prices over large multi-market or multi-asset corpora (Ding et al., 2025; Shi et al., 2025; Zhu et al., 2025). Although naturally aligned with forecasting, its supervision is inherently uncertain because realized future prices contain stochastic variation not determined by observed history. **(4) Stock-specific pretext tasks** use auxiliary metadata such as stock identity and sector as targets (Wang et al., 2025). Such objectives do not rely on future outcomes, but their connection to future-return prediction remains indirect. Overall, existing objectives often trade off future-outcome independence against downstream alignment. DIVINE instead reconstructs technical indicators from observed history, avoiding future-outcome supervision while retaining established relevance to future-return prediction.

Technical Indicators as Supervision Signals. Technical indicators summarize diverse price-volume dynamics and are widely used in alpha-factor construction (Kakushadze, 2016; Yang et al., 2020). NNAFC (Fang et al., 2020), the closest prior work using indicators as neural supervision targets, further shows that indicator-derived expressions can carry cross-sectional predictive signal. However, it trains a separate network for each indicator to generate tradable factors within a single market rather than learning a reusable shared representation. In contrast, DIVINE jointly reconstructs a diverse indicator space, discards the indicator predictions after pretraining, and transfers only the shared encoder representation across markets.

3 Problem Formulation

Notation. Let $\mathcal{M} = \{m_1, \ldots, m_M\}$ denote a set of equity markets, where market m contains T_m trading days and a stock universe $\mathcal{S}^m$. For stock $i \in \mathcal{S}^m$, let $\mathbf{X}_i^m \in \mathbb{R}^{T_m \times 5}$ denote its open, high, low, close, and volume (OHLCV) history. At day t, let $\mathbf{X}_{i,t-L+1:t}^m \in \mathbb{R}^{L \times 5}$ be the length-L lookback window and $r_{i,t\to t+h}^m = (c_{i,t+h}^m - c_{i,t}^m)/c_{i,t}^m \in \mathbb{R}$ the realized return over the next h days, where $c_{i,t}^m$ denotes the closing price.

Cross-Sectional Return Ranking. For each market m, we learn a scoring function $s^m(\cdot)$ that ranks stocks from their recent OHLCV histories such that higher-scored stocks achieve higher future returns. Following prior work (Fang et al., 2019; Den et al., 2026), we maximize the daily Spearman rank correlation between predicted scores and realized returns:

$$\max_{s^m} \mathbb{E}_t\left[\rho\left(\{s^m(\mathbf{X}_{i,t-L+1:t}^m)\}_{i\in\mathcal{S}^m}, \{r_{i,t\to t+h}^m\}_{i\in\mathcal{S}^m}\right)\right]. \tag{1}$$

We optimize equation 1 using a differentiable rank surrogate (Fang et al., 2019) and evaluate the resulting rankings through portfolio backtesting. Implementation details of backtesting and evaluation metrics are provided in Section B.3.

Pretraining and Downstream Transfer. We parameterize the scoring function as $s^m = \phi^m \circ g_\theta$, where $g_\theta : \mathbb{R}^{L\times 5} \to \mathbb{R}^{d_{\text{enc}}}$ is a shared encoder and $\phi^m : \mathbb{R}^{d_{\text{enc}}} \to \mathbb{R}$ is a lightweight market-specific prediction head. The encoder is first pretrained on a corpus $\mathcal{D}_{\text{pre}}$ drawn from markets $\mathcal{M}_{\text{pre}} \subseteq \mathcal{M}$, then transferred to market m, where ϕ^m is optimized under equation 1 with θ either frozen or jointly updated (*full fine-tuning*). We call $\mathcal{M}_{\text{pre}} = \{m\}$ as *single-market pretraining* and $|\mathcal{M}_{\text{pre}}| > 1$ as *cross-market pretraining*. Additionally, when $(|\mathcal{M}_{\text{pre}}| > 1) \wedge (m \notin \mathcal{M}_{\text{pre}})$, we refer to it as *target-excluded cross-market pretraining*. As a reference baseline, training from scratch omits pretraining and optimizes both θ and ϕ^m directly on the downstream objective.

4 Cross-Market Pretraining via Indicator Reconstruction

DIVINE pretrains an encoder to reconstruct technical indicators from raw OHLCV windows. Figure 1 provides an overview of DIVINE, while Sections 4.1 and 4.2 detail the indicator targets and pretraining/transfer protocol, respectively.

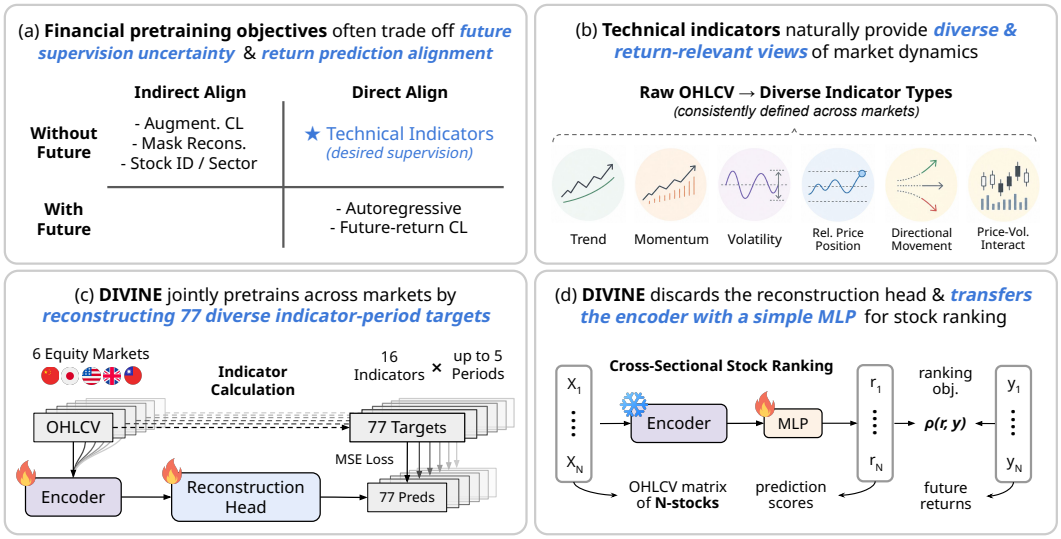

Figure 1: Overview of DIVINE, which jointly reconstructs diverse technical indicators across markets and transfers the pretrained encoder to downstream stock ranking.

4.1 Diverse Technical Indicators as Pretraining Targets

Target Definition. Given the observed price-volume history of stock i in market m up to day t, we define

$$\mathbf{z}_{i,t}^{m} = \Psi(\mathbf{X}_{i,:t}^{m}) \in \mathbb{R}^{D}, \tag{2}$$

where $\Psi = [\psi_{k_1}^{j_1}, \ldots, \psi_{k_D}^{j_D}]$ collects D indicator–period combinations, with each ψ_k^j denoting indicator j evaluated at period k. We adopt 16 standard technical indicators (e.g., SMA, RSI, MACD, ATR, and MFI) spanning diverse aspects of price-volume dynamics, including trend, volatility, momentum, relative price positioning, directional movement, and price-volume interaction. Most indicators are computed at five periods $k \in \{3, 5, 10, 14, 20\}$, with indicator-specific exceptions, yielding $D = 77$ indicator–period targets in total. The indicator set is selected for breadth of market-dynamic coverage rather than downstream performance. The full list of indicators is provided in Appendix A.

Target Normalization. To prevent scale differences across indicators and markets from dominating the reconstruction loss, we normalize inputs and targets in three stages. *First*, OHLCV inputs and price- or volume-level-dependent targets are divided by the corresponding maximum price or volume within the input window, while scale-free indicators remain unchanged. *Second*, each target dimension is clipped to $\text{median} \pm 10 \times \text{IQR}$ and standardized using training-split statistics from its market. *Third*, for cross-market pretraining, targets are re-standardized using pooled statistics across all pretraining markets. The resulting target is denoted as $\tilde{\mathbf{z}}_{i,t}^{m} \in \mathbb{R}^{D}$. Full preprocessing details and ablations are provided in Appendices C.1 and F.1, respectively.

4.2 Pretraining Objective and Downstream Transfer

Indicator Reconstruction. Given an OHLCV window $\mathbf{X}_{i,t-L+1:t}^{m} \in \mathbb{R}^{L\times 5}$, the encoder produces $\mathbf{h}_{i,t}^{m} = g_\theta(\mathbf{X}_{i,t-L+1:t}^{m}) \in \mathbb{R}^{d}$, and a linear reconstruction head $\mathbf{W} \in \mathbb{R}^{d\times D}$ predicts all D indicator targets from this shared representation, encouraging the encoder to capture price-volume structure shared across diverse indicator types and period scales. We train the encoder and reconstruction head by minimizing the mean squared reconstruction error:

$$\mathcal{L}(\theta, \mathbf{W}) = \mathbb{E}_{m,i,t}\left[\left\|\mathbf{W}^{\top} g_\theta(\mathbf{X}_{i,t-L+1:t}^{m}) - \tilde{\mathbf{z}}_{i,t}^{m}\right\|_2^2\right]. \tag{3}$$

To give all markets equal expected contribution during joint pretraining, markets are sampled inversely proportional to their training-set frequency. The reconstruction quality of each indicator–period target is reported in Appendix E.1.

Architecture and Transfer. Our default encoder g_θ is a two-layer GRU whose last hidden state forms the representation $\mathbf{h}_{i,t}^m$, with evaluations of alternative encoder architectures provided in Section 5.2. After pretraining, we discard the reconstruction head $\mathbf{W}$, attach a two-layer MLP ϕ^m for each downstream market, and optimize it using the ranking objective in equation 1. Detailed implementation and hyperparameter settings of DIVINE are provided in Appendix C.2.

5 EXPERIMENTS

5.1 EXPERIMENTAL SETUP

Datasets and Inputs We evaluate DIVINE on six equity-market datasets: CSI300 (CN), CSI500 (CN), NI225 (JP), SP500 (US), FTSE100 (UK), and TWSE300 (TW), with 2008–2019 for training, 2020 for validation, and 2021–2024 for testing. Detailed dataset statistics and the construction of stock universe are provided in Appendix B.1. Following prior work (Feng et al., 2019; Sun et al., 2023; Den et al., 2026), most baselines take OHLCV concatenated with three moving-average features as input, with preprocessing details provided in Appendix B.2. Exceptions are methods whose input formats are tied to their pretraining. For example, financial foundation models (Shi et al., 2025; Zhu et al., 2025) are pretrained on their released preprocessing pipelines, and DIVINE takes raw OHLCV as input.

Compared Baselines. We compare DIVINE against four groups of methods: **(1) Financial models trained from scratch**: RankLSTM (Feng et al., 2019), FinGAT (Hsu et al., 2021), AlphaMix (Sun et al., 2023), MASTER (Li et al., 2024), StockMixer (Fan & Shen, 2024), and DHMoE (Chen & Wang, 2025); **(2) General time-series pretraining**: TS2Vec (Yue et al., 2022), PatchTST (Nie et al., 2022), and TimesURL (Liu & Chen, 2024); **(3) Financial pretraining**: CISTHPAN (Xia et al., 2024), MERA (Chen et al., 2025), SSPT (Wang et al., 2025), and FASCL (Lee et al., 2026); and **(4) Financial foundation models**: Kronos (Shi et al., 2025) and FinCast (Zhu et al., 2025). Note that the released foundation models are pretrained on substantially larger external corpora with broader temporal coverage, so we include them as external references rather than controlled comparisons, providing a benchmark against strong industry-scale models. Implementation details and hyperparameter settings for all baselines appear in Appendix D.

Downstream Evaluation Protocol. From-scratch and pretraining baselines use a 20-day input window and 5-day prediction horizon for cross-sectional return ranking. For financial foundation models, we adapt pretrained representations to the same ranking task using model-specific preprocessing and adaptation procedures, and select the configuration most aligned with the evaluation setting of other baselines, with full ablations across all configurations reported in Appendix D.2. Unless otherwise specified, all methods follow the same portfolio backtesting protocol and report long-only Sharpe Ratio (SR) and Calmar Ratio (CR), excluding transaction costs. Each setting is evaluated over 25 seed–offset combinations (five random seeds $\times$ five staggered starting offsets), with full details provided in Appendix B.3. Note that our datasets use fixed stock universes shared by all methods, which may introduce survivorship effects. Absolute portfolio performance should therefore be interpreted as a controlled comparison rather than a point-in-time investable backtest.

5.2 MAIN RESULTS

Table 1 summarizes portfolio backtest results across six equity markets. Pretraining methods generally achieve stronger average performance than models trained from scratch, although the gains vary substantially across pretraining approaches. Notably, none of the compared financial pretraining methods surpasses PatchTST on average, suggesting that financial-domain pretraining alone does not necessarily yield stronger transfer.

Among all generic and financial pretraining baselines, DIVINE achieves the strongest average performance. With a GRU encoder, it reaches an average SR/CR of 1.396/1.790, improving over the strongest pretraining baseline, PatchTST (1.007/1.202), by 39%/49%. The same pattern holds across encoder families: the Transformer and Mamba variants also outperform all compared pretraining baselines on average, showing that DIVINE is not backbone-specific and remains robust under both frozen-encoder transfer and full fine-tuning (Appendix F.2).

Table 1: Portfolio backtest results across six equity-market datasets from 2021 to 2024. For each method, we report its parameter size in millions (M) and portfolio performance in Sharpe Ratio (SR) and Calmar Ratio (CR). Best and second-best performance per column are highlighted in **bold** and underline, respectively. * denotes a statistically significant improvement over Kronos-small in the average performance across markets ($p < 0.05$, one-sided paired t-test over 25 matched seed–offset evaluations). Standard deviations of the average SR and CR results are reported in Appendix F.3.

Markets →		CSI300		CSI500		NI225		SP500		FTSE100		TWSE300		Avg.	
Method	**Param.**	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
Financial models trained from scratch															
RankLSTM	0.05	0.753	0.663	0.869	0.699	0.610	0.447	1.414	2.138	0.314	0.226	0.645	0.461	0.768	0.772
FinGAT	0.04	0.820	0.536	1.175	1.063	0.789	0.634	0.485	0.428	0.537	0.330	0.775	0.578	0.764	0.595
MASTER	0.7	0.991	0.879	1.198	1.044	0.730	0.616	1.477	2.398	0.241	0.143	0.750	0.619	0.898	0.950
AlphaMix	0.04	0.708	0.584	1.064	0.959	0.543	0.458	1.574	2.486	0.612	0.448	0.736	0.650	0.873	0.931
StockMixer	0.01	0.726	0.612	0.615	0.421	0.450	0.361	0.919	0.899	0.407	0.263	0.763	0.554	0.647	0.518
DHMoE	10.2	0.961	0.977	1.716	**2.490**	0.679	0.585	0.980	0.680	0.612	0.384	0.464	0.292	0.902	0.901
General Time-Series Pretraining															
TS2Vec	0.6	1.014	0.893	1.522	1.640	0.780	0.664	0.952	1.008	0.493	0.344	1.140	0.989	0.983	0.923
PatchTST	0.6	0.662	0.533	1.147	1.113	0.807	0.653	1.618	3.117	0.437	0.334	1.371	1.465	1.007	1.202
TimesURL	0.6	1.022	0.972	1.302	1.277	0.576	0.448	1.057	1.191	0.412	0.268	1.117	0.958	0.914	0.852
Financial Pretraining															
MERA	1.5	1.075	1.055	0.677	0.472	0.769	0.554	1.117	1.131	0.704	0.469	1.116	0.955	0.962	0.833
CI-STHPAN	0.5	0.720	0.518	0.949	0.646	0.286	0.261	1.527	2.075	0.705	**0.553**	0.520	0.423	0.785	0.746
FASCL	14.2	0.547	0.485	0.951	0.881	0.954	0.742	1.300	1.713	**0.718**	0.504	1.246	1.275	0.953	0.933
SSPT	0.4	0.447	0.404	0.931	0.724	1.079	0.896	1.162	2.035	0.540	0.362	1.095	0.938	0.876	0.893
Financial Foundation Models															
FinCast	991.4	0.780	0.668	1.036	0.819	0.649	0.507	0.699	0.560	0.434	0.274	0.924	0.673	0.754	0.583
Kronos-mini	5.3	1.260	1.344	1.637	1.822	0.762	0.592	1.266	1.644	0.522	0.365	1.214	1.085	1.110	1.142
Kronos-small	24.5	1.459	1.516	1.916	2.229	0.903	0.783	1.523	2.329	0.524	0.378	1.209	1.186	1.256	1.404
Kronos-base	99.7	1.326	1.305	1.803	1.979	0.895	0.714	1.387	2.040	0.359	0.252	1.154	1.076	1.154	1.228
DIVINE (Ours)															
GRU encoder	0.05	**1.520**	**1.640**	**1.982**	2.207	**1.245**	**1.146**	1.645	**3.774**	0.568	0.413	1.418	1.549	**1.396***	**1.790***
Mamba encoder	0.07	1.083	1.183	1.675	1.841	1.075	0.961	**1.807**	3.519	0.645	0.425	1.273	1.336	1.259	1.544*
TFM encoder	0.1	1.383	1.371	1.811	1.813	1.193	0.970	1.456	3.586	0.646	0.446	**1.510**	**1.835**	1.333*	1.670*

Among financial foundation models, performance does not scale with model size: FinCast (991.4M parameters) underperforms even models trained from scratch, while Kronos-small (24.5M parameters) provides the strongest external reference. Kronos-small reaches an average SR/CR of 1.256/1.404, yet the default GRU-based DIVINE (0.05M parameters) achieves 11.1%/27.5% higher SR/CR with approximately $490\times$ fewer parameters, with both improvements being statistically significant across the 25 matched seed–offset evaluations. Results under realistic transaction costs are provided in Appendix F.4, where DIVINE maintains the highest average SR/CR across markets.

The main exception is FTSE100, the smallest stock universe in our evaluation, where methods that explicitly or implicitly model inter-stock relationships (e.g., AlphaMix, MERA, CI-STHPAN, FASCL) match or outperform DIVINE. This suggests that cross-stock structure may be particularly important in smaller universes, highlighting a limitation of the current per-stock encoder. While this section focuses on the default evaluation setting, Section 5.4 further evaluates robustness across prediction horizons, portfolio strategies, and data regimes.

5.3 Ablation Study of Indicator Supervision

Table 2 conducts an ablation study isolating three sources of gain in DIVINE: (i) using indicators as inputs versus reconstruction targets, (ii) single-market versus cross-market pretraining, and (iii) transferring via reconstructed indicators versus the encoder hidden state. All settings use the same indicator space and normalization, with other components matched whenever applicable.

Using indicators directly as input features improves average SR/CR from 0.829/0.938 (raw OHLCV) to 1.173/1.251, confirming that the indicator space carries more predictive information than raw price-volume features alone. Single-market indicator reconstruction further reaches 1.183/1.419, showing that reconstruction-based representation learning provides additional benefit beyond direct indicator inputs. The largest gain comes from cross-market pretraining: holding the indicator target and downstream signal fixed, expanding pretraining from one market to multiple markets raises SR/CR from 1.183/1.419 to 1.396/1.790. Finally, under identical cross-market pretraining, the hid-

Table 2: Ablation of DIVINE along three factors: indicator role, pretraining scope, and downstream signal, reported in SR and CR. The upper block reports from-scratch baselines and the lower block reports pretraining variants, with the final row corresponding to the full DIVINE. Best results per column are shown in **bold**.

Ablation Factors			CSI300		CSI500		NI225		SP500		FTSE100		TWSE300		Avg.	
Indicator Role	Pretraining Scope	Downstream Signal	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
–	–	–	0.545	0.323	0.644	0.437	0.793	0.670	1.650	3.018	0.417	0.320	0.922	0.859	0.829	0.938
Input	–	–	1.411	1.475	1.587	1.374	0.856	0.749	1.485	2.388	0.532	0.321	1.169	1.200	1.173	1.251
Target	Single-Market	Hidden	1.146	1.041	1.658	1.626	1.117	0.828	1.474	3.374	0.458	0.288	1.243	1.357	1.183	1.419
Target	Cross-Market	Recon. Ind.	1.483	1.594	1.897	1.831	1.098	1.000	1.641	3.175	**0.635**	**0.467**	1.247	1.333	1.333	1.567
Target	Cross-Market	Hidden	**1.520**	**1.640**	**1.982**	**2.207**	**1.245**	**1.146**	**1.645**	**3.774**	0.568	0.413	**1.418**	**1.549**	**1.396**	**1.790**

den representation outperforms the reconstructed indicators (1.396/1.790 vs. 1.333/1.567), indicating that the encoder learns ranking-relevant structure beyond the reconstructed targets themselves.

5.4 Robustness and Data Efficiency

Robustness across downstream settings. Table 3 compares DIVINE against five best-performing methods in Table 1 covering diverse pretraining objectives: masked reconstruction (PatchTST), contrastive learning (FASCL, TS2Vec), retrieval-augmented reconstruction (MERA), and autoregressive pretraining (Kronos-small). These methods are evaluated across five prediction horizons under both long-only and long-short strategies, yielding ten downstream settings in total. While MERA, PatchTST, and Kronos-small lead in specific settings, DIVINE achieves the highest overall average SR (1.715) and best average rank (1.6), ranking first in five settings and second in four, making it the most consistent method across prediction horizons and portfolio strategies.

Table 3: Average Sharpe Ratio (SR) across six markets under ten downstream settings formed by five prediction horizons and two portfolio strategies. Summary columns report mean SR per strategy and overall, and mean rank across all settings (lower is better). Best and second-best results are shown in **bold** and underline, respectively.

Portfolio Strategy	Long-only (LO)					Long-Short (LS)					Summary (Avg.)			
Prediction Horizon	1D	3D	5D	10D	20D	1D	3D	5D	10D	20D	LO	LS	Overall	Rank
From Scratch	1.345	0.742	0.829	0.878	0.802	1.196	0.508	0.653	0.666	0.683	0.919	0.741	0.830	6.1
TS2Vec	2.280	1.171	0.983	0.766	0.684	3.026	1.430	1.062	0.745	0.559	1.177	1.364	1.271	5.8
PatchTST	2.657	1.190	1.007	0.967	**0.904**	3.403	1.513	1.246	0.989	0.760	1.345	1.582	1.464	3.3
FASCL	2.558	1.311	0.953	0.797	0.602	3.093	1.485	1.070	0.814	0.449	1.244	1.382	1.313	5.5
MERA	**3.309**	1.376	0.962	0.900	0.778	**4.233**	1.847	1.462	**1.195**	**0.923**	1.465	**1.932**	1.699	2.4
Kronos-small	3.034	1.638	1.256	0.892	0.731	3.856	1.901	1.337	0.841	0.545	1.510	1.696	1.603	3.3
DIVINE	3.158	**1.688**	**1.396**	**1.087**	0.850	3.543	**1.984**	**1.524**	1.121	0.801	**1.636**	1.795	**1.715**	**1.6**

Data efficiency across pretraining and finetuning. Figure 2 evaluates scaling with pretraining data (left) and downstream labels (right). DIVINE improves steadily with more pretraining data and, using only 10% of the corpus, already surpasses all compared pretraining baselines trained on the full corpus. It also leads across all downstream-label fractions and exceeds full-data training from scratch with only 5% of the labels. In contrast, no other pretraining method matches DIVINE at any data fraction, including Kronos-mini despite its substantially larger model size and external pretraining corpus. Similar trends hold for CR, with full results in Appendix E.2.

Having established DIVINE's strong performance across horizons, strategies, and data regimes, we next examine what drives these gains: which properties of the indicator target space matter (Section 6.1), and how the composition of pretraining markets shapes cross-market transfer (Section 6.2).

6 Analysis

6.1 What Makes a Good Indicator Set for Pretraining?

Indicator diversity. We examine how the breadth of the supervision space affects transfer performance by randomly sampling $n \in \{2, 4, 8, 12\}$ of the 16 indicators while retaining all period

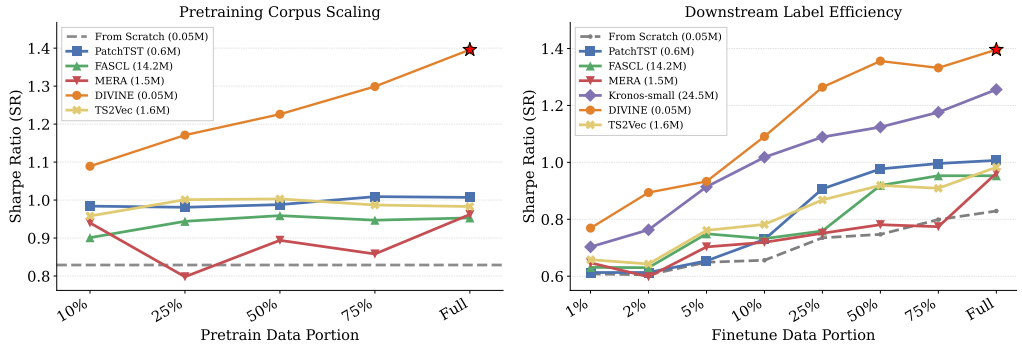

Figure 2: Pretraining-corpus scaling (left) and downstream-label efficiency (right) in average SR across six markets. Kronos-small is excluded from the left panel as an externally pretrained model.

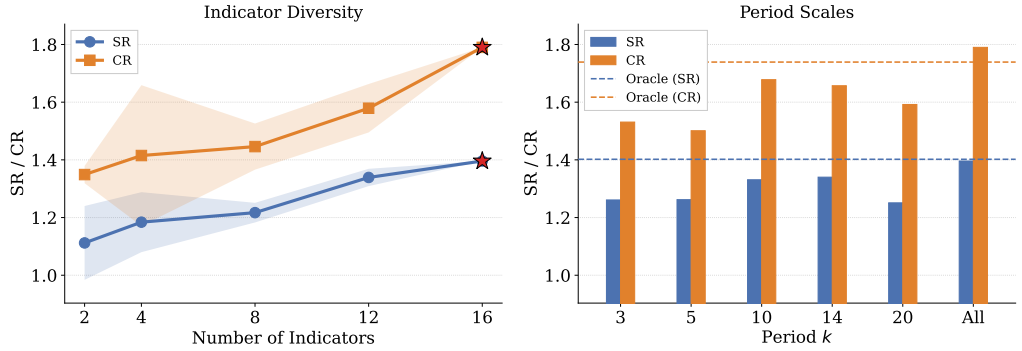

Figure 3: Average SR and CR across six markets when varying the number of pretraining indicators n (left; mean and std across three random draws for each n) and the period scale k (right).

scales (Figure 3, left). Both SR and CR improve monotonically with n, while standard deviation across random draws decreases, showing that broader indicator coverage yields stronger and more consistent downstream performance.

Temporal-scale diversity. Holding all 16 indicators fixed, we then vary their period scale k (Figure 3, right). No single period scale consistently outperforms others, while combining all scales performs comparably to the best single scale selected per market in hindsight (Oracle; see Appendix E.3). This is consistent with the observation that indicators computed over nearby period scales are inherently correlated, providing limited additional signal to the encoder beyond what a single scale already captures.

Together, these results reveal a clear design principle: ***indicator-type diversity matters substantially more than period-scale diversity***. Importantly, this pattern cannot be attributed simply to having more reconstruction targets: the 4-indicator setting already yields roughly 20 targets (via multiple period scales) yet still underperforms a configuration using all indicator types at a single period scale (i.e., 16 targets). The gain therefore stems from covering a broader range of distinct market dynamics, not from increasing the number of correlated targets.

6.2 How Does Pretraining Market Composition Affect Transfer?

Building on the cross-market gains established in Section 5.3, Table 4 examines how transfer varies with the composition of the pretraining markets. We consider both single-source pretraining and target-excluded cross-market pretraining (defined in Section 3), where the latter provides a strict OOD setting in which the target market is never observed during pretraining.

Table 4: Average SR across six markets under different pretraining-market compositions. Gray: in-domain training (pretrain and evaluate on the same market). Blue: single-source transfer exceeding the target's in-domain result. Orange: target-excluded pretraining (target market unseen during pretraining). Parentheses report the gap to joint pretraining (ΔSR $= 1.396 -$ Avg. SR). **Bold** and underline indicate the best value per column overall and within each block, respectively.

	Evaluation market						
Pretraining setting ↓	**CSI300**	**CSI500**	**NI225**	**SP500**	**FTSE100**	**TWSE300**	**Avg. SR (ΔSR)**
One Market Only							
CSI300-only	1.146	1.608	1.053	1.468	0.391	1.272	1.158 (+.238)
CSI500-only	1.072	1.658	0.954	1.516	0.463	1.175	1.140 (+.256)
NI225-only	0.633	1.310	1.117	1.370	0.478	1.273	1.047 (+.349)
SP500-only	0.543	1.445	0.950	1.474	0.427	1.413	1.042 (+.354)
FTSE100-only	0.630	1.250	1.032	1.418	0.458	1.294	1.014 (+.382)
TWSE300-only	0.383	1.100	1.019	1.631	0.402	1.243	0.963 (+.433)
Remove One Market							
w/o CSI300	1.318	1.875	1.135	1.491	0.560	1.344	1.287 (+.109)
w/o CSI500	1.054	1.754	1.113	1.559	**0.617**	1.414	1.251 (+.145)
w/o NI225	1.264	1.969	1.200	1.549	0.587	1.405	1.329 (+.067)
w/o SP500	1.324	**2.129**	1.204	**1.710**	0.394	1.403	1.361 (+.035)
w/o FTSE100	1.321	1.949	**1.273**	1.664	0.594	1.412	1.369 (+.027)
w/o TWSE300	1.245	1.925	1.188	1.690	0.522	1.345	1.319 (+.077)
Overall							
Single Market	1.146	1.658	1.117	1.474	0.458	1.243	1.183 (+.213)
OOD performance	1.318	1.754	1.200	1.710	0.594	1.345	1.320 (+.076)
Joint All Markets	**1.520**	1.982	1.245	1.645	0.568	**1.418**	**1.396** (+.000)

Transfer to unseen markets. The *Overall* block shows strong transfer to unseen markets. Target-excluded pretraining outperforms single-market pretraining on all six markets, increasing average SR from 1.183 to 1.320, compared with 1.396 under joint pretraining. Thus, 64% of the single-market-to-joint improvement is recovered without target-market pretraining data, showing that indicator-based representations transfer effectively across markets.

Complementary roles of source markets. Transfer nevertheless varies across source–target pairs. The *One Market Only* block shows the strongest positive transfer between SP500 and TWSE300 in both directions, while the *Remove One Market* block shows the largest overall drops when CSI500 or CSI300 is removed. This distinction suggests that some markets transfer particularly well to specific targets, whereas others contribute structure that is broadly useful across markets. CR also show a similar trend to what we observed in SR, with full results are reported in Appendix E.4.

Toward scalable financial pretraining. Together with the indicator-space analysis in Section 6.1, these results reveal two complementary sources of transfer: ***diversity in what is reconstructed and diversity in where it is learned***. Diversity across indicator types broadens the supervision space, while market diversity exposes the encoder to heterogeneous market dynamics. Combined with the data-scaling results in Section 5.4, these findings support supervision and market diversity as key ingredients for scalable financial pretraining.

7 Conclusion

We presented DIVINE, a simple cross-market pretraining framework that learns financial representations by reconstructing diverse technical indicators from raw OHLCV history. Across six equity markets, DIVINE achieves the strongest average portfolio performance with a lightweight encoder, outperforming existing pretraining baselines and matching or exceeding substantially larger financial foundation models, while remaining robust and data-efficient. Our analyses further show that the gains arise from the learned representation rather than merely from exposure to technical indicators. More broadly, DIVINE benefit from diversity in both the supervision signal and the pretraining markets: broader indicator-type coverage strengthens representation learning, while heterogeneous source markets provide reusable structure even without observing the target market during pretraining. Together, these findings suggest that supervision-target design and cross-market diversity offer a principled and lightweight path toward transferable financial representations.

AI USE STATEMENT

We used generative AI tools for literature survey, feedback on experimental validation, and assistance with interpreting experimental results. We additionally used generative AI for manuscript organization, drafting assistance, and language editing. All scientific claims, experimental results, retrieved references, and AI-assisted content were critically reviewed and verified by the authors. The authors take full responsibility for the final content of this work.

REPRODUCIBILITY STATEMENT

To support reproducibility, we provide detailed specifications of the experimental pipeline throughout the main paper and appendices. Appendix A specifies the technical-indicator targets and their configurations; Appendix B describes data acquisition, stock-universe construction, temporal splits, input and target construction, and the portfolio backtesting protocol; Appendix C provides the target preprocessing procedure together with model architectures, training configurations, and optimization settings of DIVINE; and Appendix D details the implementation, hyperparameters, and transfer protocols of the compared baselines.

REFERENCES

Gerald Appel. *Technical Analysis: Power Tools for Active Investors*. FT Press, 2005.

Doron Avramov. Stock return predictability and model uncertainty. *Journal of Financial Economics*, 64(3):423–458, 2002.

Costas Azariadis and Bruce Smith. Financial intermediation and regime switching in business cycles. *American economic review*, pp. 516–536, 1998.

Mehmet Caner and Maurizio Daniele. Deep learning based residuals in non-linear factor models: Precision matrix estimation of returns with low signal-to-noise ratio. *Journal of Econometrics*, 251:106083, 2025. doi: 10.1016/j.jeconom.2025.106083.

Haokun Chen et al. MERA: Mixture of experts with retrieval-augmented representation for modeling diversified stock patterns. In *Companion Proceedings of the ACM on Web Conference*, 2025. doi: 10.1145/3701716.3715513.

Weijun Chen and Yanze Wang. DHMoE: Diffusion generated hierarchical multi-granular expertise for stock prediction. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 11490–11499, 2025.

Yu-Chen Den, Kuan-Yu Chen, Kendro Vincent, and Tien-Hao Chang. Integrating inductive biases in transformers via distillation for financial time series forecasting. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 803–814, 2026.

Xueying Ding, Aakriti Mittal, and Achintya Gopal. Delphyne: A pre-trained model for general and financial time series. *arXiv preprint arXiv:2506.06288*, 2025.

Jinyong Fan and Yanyan Shen. Stockmixer: A simple yet strong MLP-based architecture for stock price forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 8389–8397, 2024.

Jie Fang, Shutao Xia, Jianwu Lin, Zhikang Xia, Xiang Liu, and Yong Jiang. Alpha discovery neural network based on prior knowledge. *arXiv preprint arXiv:1912.11761*, 2019.

Jie Fang, Jianwu Lin, Shutao Xia, Yong Jiang, Zhikang Xia, and Xiang Liu. Neural network-based automatic factor construction. *Quantitative Finance*, 20(12):1715–1735, 2020.

Fuli Feng, Huimin Chen, Xiangnan He, Ji Ding, Maosong Sun, and Tat-Seng Chua. Enhancing stock movement prediction with adversarial training. *arXiv preprint arXiv:1810.09936*, 2018.

Fuli Feng, Xiangnan He, Xiang Wang, Cheng Luo, Yiqun Liu, and Tat-Seng Chua. Temporal relational ranking for stock prediction. *ACM Transactions on Information Systems*, 37(2):1–30, 2019.

Guanhai Feng, Stefano Giglio, and Dacheng Xiu. Taming the factor zoo: A test of new factors. *The Journal of Finance*, 75(3):1327–1370, 2020. doi: https://doi.org/10.1111/jofi.12883. URL https://onlinelibrary.wiley.com/doi/abs/10.1111/jofi.12883.

Shihao Gu, Bryan Kelly, and Dacheng Xiu. Empirical asset pricing via machine learning. *The Review of Financial Studies*, 33(5):2223–2273, 2020.

Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16000–16009, 2022.

Yi-Ling Hsu, Yu-Che Tsai, and Cheng-Te Li. Fingat: Financial graph attention networks for recommending top-K profitable stocks. *IEEE Transactions on Knowledge and Data Engineering*, 35 (1):469–481, 2021.

Yifan Hu, Yuante Li, Peiyuan Liu, Yuxia Zhu, Naiqi Li, Tao Dai, Shu-tao Xia, Dawei Cheng, and Changjun Jiang. Fintsb: A comprehensive and practical benchmark for financial time series forecasting, 2025. URL https://arxiv.org/abs/2502.18834.

Zura Kakushadze. 101 formulaic alphas. *Wilmott*, 2016(84):72–81, 2016.

Adriano Koshiyama, Stefano B. Blumberg, Nick Firoozye, Philip Treleaven, and Sebastian Flennerhag. Quantnet: transferring learning across trading strategies. *Quantitative Finance*, 21(12), 2021. doi: 10.1080/14697688.2021.1999487.

Donald R. Lambert. Commodity channel index: Tool for trading cyclic trends. *Commodities*, 9(10), 1980.

Hyeongmin Lee, Chanyeol Choi, Jihoon Kwon, Yoon Kim, Alejandro Lopez-Lira, Wonbin Ahn, and Yongjae Lee. Cross-sectional asset retrieval via future-aligned soft contrastive learning. *arXiv preprint arXiv:2602.10711*, 2026.

Tong Li, Zhaoyang Liu, Yanyan Shen, Xue Wang, Haokun Chen, and Sen Huang. Master: Market-guided stock transformer for stock price forecasting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 162–170, 2024.

Jiexi Liu and Songcan Chen. TimesURL: Self-supervised contrastive learning for universal time series representation learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 13918–13926, 2024.

John J Murphy. *Study guide to technical analysis of the financial Markets: a comprehensive guide to trading methods and applications*. Penguin, 1999.

Yuqi Nie, Nam H Nguyen, Phanwadee Sinthong, and Jayant Kalagnanam. A time series is worth 64 words: Long-term forecasting with transformers. *arXiv preprint arXiv:2211.14730*, 2022.

David E. Rapach, Jack K. Strauss, and Guofu Zhou. Out-of-sample equity premium prediction: Combination forecasts and links to the real economy. *The Review of Financial Studies*, 23(2): 821–862, 2010. doi: 10.1093/rfs/hhp063.

Ramit Sawhney, Shivam Agarwal, Arnav Wadhwa, Tyler Derr, and Rajiv Ratn Shah. Stock selection via spatiotemporal hypergraph attention network: A learning to rank approach. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 497–504, 2021.

Yu Shi, Zongliang Fu, Shuo Chen, Bohan Zhao, Wei Xu, Changshui Zhang, and Jian Li. Kronos: A foundation model for the language of financial markets. *arXiv preprint arXiv:2508.02739*, 2025.

Shuo Sun, Xinrun Wang, Wanqi Xue, Xiaoxuan Lou, and Bo An. Mastering stock markets with efficient mixture of diversified trading experts. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 2109–2119, 2023.

Mengyu Wang, Tiejun Ma, and Shay B. Cohen. Pre-training time series models with stock data customization. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2025. doi: 10.1145/3711896.3737005.

J. Welles Wilder. *New Concepts in Technical Trading Systems*. Trend Research, 1978.

Hongjie Xia, Huijie Ao, Long Li, Yu Liu, Sen Liu, Guangnan Ye, and Hongfeng Chai. CI-STHPAN: Pre-trained attention network for stock selection with channel-independent spatio-temporal hypergraph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 9187–9195, 2024.

Xiao Yang, Weiqing Liu, Dong Zhou, Jiang Bian, and Tie-Yan Liu. Qlib: An AI-oriented quantitative investment platform. *arXiv preprint arXiv:2009.11189*, 2020.

Jaemin Yoo, Yejun Soun, Yong-chan Park, and U Kang. Accurate multivariate stock movement prediction via data-axis transformer with multi-level contexts. In *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, pp. 2037–2045, 2021.

Zhihan Yue, Yujing Wang, Juanyong Duan, Tianmeng Yang, Changhai Zheng, Yunhai Tong, and Bixiong Xu. Ts2vec: Towards universal representation of time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 8980–8987, 2022.

Zhuohang Zhu, Haodong Chen, Qiang Qu, and Vera Chung. FinCast: A foundation model for financial time-series forecasting. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, 2025. doi: 10.1145/3746252.3761261.

A TECHNICAL INDICATOR TARGETS

Table 5 summarizes the 16 technical indicators used to construct the pretraining targets of DIVINE. All indicator values are computed using `stockstats` (version 0.6.8) [1], which serves as the authoritative implementation used in our experiments. The descriptions are intended to provide intuition about the financial quantity summarized by each target rather than to reproduce the exact numerical implementation.

Several indicators have well-established origins in the technical-analysis literature. RSI, ATR, and the directional-movement indicators DX and ADX follow Wilder (Wilder, 1978); MACD follows Appel (Appel, 2005); and CCI follows Lambert (Lambert, 1980). The remaining indicators are standard technical-analysis constructions summarized, for example, by Murphy (Murphy, 1999).

Unless otherwise specified, we use $\mathcal{K} = \{3, 5, 10, 14, 20\}$ as the set of period configurations k. MACD uses its fixed $(12, 26, 9)$ parameterization, while RSV additionally includes $k = 1$, yielding $D = 77$ indicator–period targets in total.

Table 5: Technical indicators used as pretraining targets. Descriptions summarize the financial quantity captured by each indicator; all numerical values follow the `stockstats` v0.6.8 implementation. The Scale-free column indicates whether a target is invariant to the overall price level (Y) and therefore unaffected by the sample-level normalization in Appendix C.1.

Indicator	**Description**	**Period k**	**Scale-free**
Simple Moving Average (SMA)	Mean closing price over the recent k periods.	$k \in \mathcal{K}$	N
Exponential Moving Average (EMA)	Exponentially weighted average of recent closing prices, assigning greater weight to more recent observations.	$k \in \mathcal{K}$	N
Moving Average Convergence Divergence (MACD)	Difference between short- and long-horizon exponential moving averages, summarizing changes in trend and momentum.	$(12, 26, 9)$	N
True Range (TR)	Recent trading-range magnitude accounting for both intraday high–low variation and gaps from the previous close.	$k \in \mathcal{K}$	N
Average True Range (ATR)	Smoothed measure of recent true range, commonly used to summarize price volatility.	$k \in \mathcal{K}$	N
Rate of Change (ROC)	Relative change in closing price over the preceding k periods.	$k \in \mathcal{K}$	Y
Relative Strength Index (RSI)	Relative magnitude of recent upward and downward price movements.	$k \in \mathcal{K}$	Y
Triple Exponential Average (TRIX)	Rate of change of a triple-smoothed exponential moving average, capturing smoothed price momentum.	$k \in \mathcal{K}$	Y
TRIX Moving Average (TRMA)	Moving average of TRIX, providing a smoother view of TRIX dynamics.	$k \in \mathcal{K}$	Y
Raw Stochastic Value (RSV)	Position of the current close within the recent high–low trading range.	$k \in \mathcal{K} \cup \{1\}$	Y
Stochastic %K	Smoothed stochastic value derived from RSV.	$k \in \mathcal{K}$	Y
Stochastic %D	Further smoothed version of Stochastic %K.	$k \in \mathcal{K}$	Y
Directional Index (DX)	Strength of directional price movement over the recent k periods.	$k \in \mathcal{K}$	Y
Average Directional Index (ADX)	Smoothed measure of directional-movement strength.	$k \in \mathcal{K}$	Y
Commodity Channel Index (CCI)	Deviation of typical price from its recent moving average relative to its typical dispersion.	$k \in \mathcal{K}$	Y
Money Flow Index (MFI)	Price–volume momentum measure combining typical price changes with trading volume.	$k \in \mathcal{K}$	Y

B DATASETS AND EVALUATION

B.1 DATA ACQUISITION AND STOCK UNIVERSE CONSTRUCTION

We collect daily stock-level OHLCV data for six equity-market datasets: CSI300, CSI500, S&P 500, Nikkei 225, FTSE 100, and Taiwan stocks. Historical data for CSI300 and CSI500 are obtained from `Qlib`, while data for S&P 500, Nikkei 225, and FTSE 100 are retrieved from Yahoo Finance through the `yfinance` package. Taiwan stock data are obtained from `FinLab`. For each dataset, we retain the available daily open, high, low, close, and trading-volume observations within the 2008–2024 study period.

[1] `https://github.com/jealous/stockstats`

To ensure consistent temporal coverage, we apply a common coverage criterion when constructing the final stock universe for each dataset. For the Taiwan market, we first select the 300 stocks with the highest average trading volume over the 2008–2024 study period and denote this liquidity-based candidate universe as TWSE300. We then apply the coverage filter to all datasets, retaining stocks whose trading histories span the full study period and whose missing rate across OHLCV observations is below 3%. All methods are trained and evaluated on the same resulting stock universe for each market to support controlled method comparisons rather than a fully point-in-time investable-universe evaluation. Table 6 summarizes the final stock counts and temporal splits.

Table 6: Dataset statistics after stock-universe filtering.

Market	Region	# Stocks	Train	Validation	Test
CSI300	CN	295	2008–2019	2020	2021–2024
CSI500	CN	514			
NI225	JP	209			
SP500	US	525			
FTSE100	UK	81			
TWSE300	TW	205			

B.2 Input Features and Prediction Target

Following previous work (Sun et al., 2023; Den et al., 2026), we construct an eight-dimensional input vector consisting of standardized OHLCV variables over a 20-day rolling window and three moving-average deviation features at periods $k \in \{5, 10, 20\}$ days. The prediction target is the forward 5-day return. Formally:

$$\tilde{x}^t_{\text{close}} = \frac{x^t_{\text{close}} - \bar{x}^{[t-19:t]}_{\text{close}}}{\sigma^{[t-19:t]}_{x_{\text{close}}}}, \quad \tilde{x}^t_{d_k} = \frac{\frac{1}{k}\sum_{i=0}^{k-1} x^{t-i}_{\text{close}}}{x^t_{\text{close}}} - 1, \quad y_t = \frac{x^{t+h}_{\text{close}} - x^t_{\text{close}}}{x^t_{\text{close}}},$$

where $\tilde{x}^t_{\text{close}}$ denotes the normalized close price (O, H, L, and V are normalized identically), $\tilde{x}^t_{d_k}$ the moving-average deviation feature, and y_t the prediction target.

B.3 Portfolio Backtesting Protocol

We evaluate stock rankings through portfolio backtesting with top-k stock selection and softmax-based portfolio weighting. For each rebalancing day, the k stocks with the highest prediction scores are selected, and a softmax over their scores determines the portfolio weights. Unless otherwise specified, k is set to 5.

Because the default prediction horizon is $h = 5$ trading days, each portfolio is held for $W = h = 5$ trading days before rebalancing. A single choice of the initial rebalancing day would make the resulting backtest dependent on an arbitrary calendar offset. We therefore evaluate all W possible staggered starting offsets. Specifically, for each offset $w \in \{0, \ldots, W-1\}$, we construct a portfolio stream that starts at offset w and is subsequently rebalanced every W trading days. Algorithm 1 implements these staggered portfolio streams jointly using a circular buffer, where $\boldsymbol{Y}[w, :]$ corresponds to the daily portfolio-return series $\{r^{(w)}_t\}$ for starting offset w.

For each random seed $s \in \{0, \ldots, 4\}$ and starting offset $w \in \{0, \ldots, 4\}$, let $\{r^{(s,w)}_t\}_{t=1}^{T_w}$ denote the resulting daily portfolio-return series. We report the annualized Sharpe Ratio (SR) and Calmar Ratio (CR). Let

$$\bar{r}^{(s,w)} = \frac{1}{T_w}\sum_{t=1}^{T_w} r^{(s,w)}_t, \tag{4}$$

and

$$\sigma^{(s,w)} = \sqrt{\frac{1}{T_w}\sum_{t=1}^{T_w}\left(r^{(s,w)}_t - \bar{r}^{(s,w)}\right)^2}. \tag{5}$$

The Sharpe Ratio is

$$\mathrm{SR}^{(s,w)} = \sqrt{252}\,\frac{\bar{r}^{(s,w)}}{\sigma^{(s,w)}}. \tag{6}$$

For the Calmar Ratio, let

$$C_t^{(s,w)} = \sum_{\tau=1}^{t} r_\tau^{(s,w)} \tag{7}$$

denote the cumulative return and define the maximum drawdown as

$$D^{(s,w)} = \max_t \left(\max_{\tau \le t} C_\tau^{(s,w)} - C_t^{(s,w)} \right). \tag{8}$$

The Calmar Ratio is then

$$\mathrm{CR}^{(s,w)} = \frac{252\,\bar{r}^{(s,w)}}{D^{(s,w)}}. \tag{9}$$

For each model, we evaluate five random seeds and five staggered starting offsets, yielding $5 \times 5 = 25$ seed–offset evaluations per setting:

$$\overline{\mathrm{SR}} = \frac{1}{25}\sum_{s=0}^{4}\sum_{w=0}^{4} \mathrm{SR}^{(s,w)}, \qquad \overline{\mathrm{CR}} = \frac{1}{25}\sum_{s=0}^{4}\sum_{w=0}^{4} \mathrm{CR}^{(s,w)}. \tag{10}$$

Unless otherwise stated, transaction costs are not included to remain consistent with the benchmark evaluation protocol.

Algorithm 1 Portfolio Return Calculation with Staggered Rebalancing

Require: Predictions $\boldsymbol{P} \in \mathbb{R}^{B\times N}$, next-day returns $\boldsymbol{R} \in \mathbb{R}^{B\times N}$, window length W
Ensure: Portfolio returns $\boldsymbol{Y} \in \mathbb{R}^{W\times B}$

Initialize portfolio return matrix $\boldsymbol{Y} \leftarrow \boldsymbol{0}^{W\times B}$
for $d = 0$ to $B-1$ **do** ▷ For each day in batch
  $\texttt{row_index} \leftarrow d \bmod W$
  $\texttt{time_range} \leftarrow \min(W, B-d)$
  Extract predictions $\boldsymbol{p}_d \leftarrow \boldsymbol{P}[d,:]$ ▷ Shape: N
  Sort stocks: $\boldsymbol{I} \leftarrow \mathrm{argsort}(\boldsymbol{p}_d, \text{descending})$
  $\mathcal{S}_{\text{top}} \leftarrow \boldsymbol{I}[:k]$ ▷ Top-k stock indices
  $\boldsymbol{p}_{\text{top}} \leftarrow \boldsymbol{p}_d[\mathcal{S}_{\text{top}}]$ ▷ Top-k predictions
  $\boldsymbol{w} \leftarrow \mathrm{softmax}(\boldsymbol{p}_{\text{top}})$ ▷ Portfolio weights
  $\boldsymbol{R}_{\text{range}} \leftarrow \boldsymbol{R}[d : d+\texttt{time_range}, \mathcal{S}_{\text{top}}]$ ▷ Future returns
  $\boldsymbol{Y}[\texttt{row_index}, d : d+\texttt{time_range}] \mathrel{+}= \boldsymbol{R}_{\text{range}}^{\top}\boldsymbol{w}$ ▷ Accumulate weighted returns
end for
return $\boldsymbol{Y}$

Extension to Long–Short Portfolios. The same backtesting procedure can be extended to the long–short setting. The long leg is constructed exactly as in Algorithm 1. For the short leg, we instead select the bottom-k stocks according to the same prediction scores, negate their scores before applying the softmax so that lower-ranked stocks receive larger short weights, and multiply the resulting portfolio return by -1. The daily long–short return is then obtained by equally averaging the returns of the long and short legs. We compute the Sharpe and Calmar ratios from this combined daily return series using the same staggered offsets and seed averaging described above.

C Implementation Details of DIVINE

C.1 Detailed Preprocessing Algorithm (Section 4.1)

Technical indicators are computed once along each stock's historical series using only observations available up to each timestamp, rather than being reinitialized within each encoder input window. This preserves the standard recursive computation of indicators such as EMA and avoids window-dependent reinitialization effects.

We provide the exact preprocessing procedure used to construct the normalized indicator targets in Algorithm 2. All statistics used for clipping and standardization are estimated from the training split only and are kept fixed for validation and testing. The procedure consists of three stages: sample-level scaling of OHLCV and scale-dependent indicator targets, per-market clipping and standardization, and pooled re-standardization for joint cross-market pretraining. It is worth noting that scale-free indicators are not affected by the sample-level price normalization, while price-dependent targets are divided by the corresponding within-window maximum price. Table 5 lists the classification of each target.

Algorithm 2 Target Normalization, PyTorch-style

```
# X: (N, L, 5) OHLCV input windows for market m
# Z: (N, D) raw indicator targets for market m
# price_idx: indices of price-based target dimensions
# scale_free_idx: indices of scale-free target dimensions
# Training split only; statistics are frozen at inference.

def per_sample_max_norm(X, Z):
    p_max = X[:, :, :4].max(dim=1).values.max(dim=1).values  # (N,)
    v_max = X[:, :, 4].max(dim=1).values                      # (N,)
    X[:, :, :4] = X[:, :, :4] / p_max[:, None, None]
    X[:, :, 4]  = X[:, :, 4]  / v_max[:, None]
    Z[:, price_idx] = Z[:, price_idx] / p_max[:, None]
    return X, Z  # scale-free dims untouched

def per_market_clip_and_znorm(Z):
    # statistics computed over all training samples (N,) per dim
    med = Z.median(dim=0).values                # (D,)
    iqr = Z.quantile(0.75, dim=0) - Z.quantile(0.25, dim=0)  # (D,)
    Z = Z.clip(med - 10 * iqr, med + 10 * iqr)  # IQR clipping
    mu, sigma = Z.mean(dim=0), Z.std(dim=0)      # (D,)
    Z = (Z - mu) / (sigma + eps)                  # z-norm
    return Z, mu, sigma

# --- Stage 1: per-sample max-normalization ---
X, Z = per_sample_max_norm(X, Z)

# --- Stage 2: per-market IQR clipping + z-normalization ---
Z, mu_m, sigma_m = per_market_clip_and_znorm(Z)

# --- Stage 3 (cross-market only): pooled re-standardization ---
# After concatenating all markets:
# Z_all: (N_total, D), padded to common cross-section width
mu_pool, sigma_pool = Z_all.mean(dim=0), Z_all.std(dim=0)
Z_all = (Z_all - mu_pool) / (sigma_pool + eps)
```

C.2 Architecture and Training Configuration

Table 7 summarizes the optimization and transfer settings used for DIVINE, while Table 8 reports the architecture-specific configurations of the GRU, Transformer, and Mamba encoders. Unless otherwise stated, all encoder variants use the same indicator reconstruction objective, preprocessing pipeline, and downstream ranking protocol.

During pretraining, the encoder and a randomly initialized linear reconstruction head are optimized jointly to reconstruct the 77 normalized indicator targets from a 120-day OHLCV context. Each training sample is one trading day's full stock cross-section, and a batch of 256 days may draw from different markets under cross-market pretraining. Because stock universes differ in size across markets, cross-sections are zero-padded to a common width and the padded entries are excluded from the reconstruction loss. Batches are accumulated over 16 steps of 16 days each. For all backbones, the final sequence representation is mapped to the same 77-dimensional target space by this

linear head. After pretraining, the reconstruction head is discarded and replaced by a two-layer market-specific MLP with hidden dimension 64 for downstream stock ranking.

DIVINE is transferred with a frozen encoder by default. Freezing isolates the contribution of the pretrained representation, so that differences across pretraining settings in Section 5.3 reflect the pretraining objective rather than downstream adaptation. The Transformer encoder is the sole exception, arising from two interacting factors: **(1)** the gap between the 120-day pretraining context length and 20-day downstream windows, and **(2)** the use of sinusoidal positional encoding. Because the representation is taken from the last token, the positional encoding assigns the downstream representation a position index that differs from the one it occupies during pretraining. We offset the downstream positions so that the 20-day window occupies the final positions of the pretraining context (shifting the position index from 0–19 to 100–119), without which transfer degrades severely. Even with this correction, frozen transfer remains weaker than full fine-tuning for the Transformer. We therefore fully fine-tune the Transformer encoder, allowing its positional representations to adapt to the shorter downstream window. Results under both transfer protocols are reported in Appendix F.2. In constrast, the recurrent and state-space encoders carry no absolute positional encoding and are therefore unaffected. Adapting the framework to architectures with absolute positional encodings, for instance through relative or bias-based position representations, remains an open direction.

For both pretraining and downstream transfer, we use AdamW with a fixed learning rate of 10^{-3} and no weight decay or learning-rate scheduling. Both stages are trained for 100 epochs with an effective batch size of 256, without early stopping. Gradients are accumulated over 16 steps and clipped to a maximum norm of 1.0. Final performance is averaged over five random seeds. All experiments are implemented in PyTorch and run in full precision on NVIDIA RTX 3090 GPUs.

Table 7: Training and transfer configuration of DIVINE. Settings are shared across encoder backbones unless otherwise specified.

Stage	Configuration	Value
Pretraining	Input features	OHLCV
	Context length	120 days
	Reconstruction targets	77
	Reconstruction head	Linear
	Loss	Mean squared error
Downstream transfer	Input features	OHLCV
	Input window	20 days
	Prediction horizon	5 days
	Prediction head	2-layer MLP
	Encoder update	Frozen by default
	Ranking objective	Differentiable Spearman loss
Shared Settings	Optimizer	AdamW
	Learning rate	10^{-3}
	Weight decay	0.0
	Effective batch size	256 days
	Training epochs	100
	Gradient accumulation	16 steps (16 days each)
	Gradient clipping	1.0
	LR scheduler	None
	Early stopping	None
	Random seeds	$\{0, 1, 2, 3, 4\}$

D Implementation Details of Baselines

D.1 Generic and Financial Pretraining Methods

Transfer Protocol. We follow each method's original transfer protocol whenever it can be applied directly to cross-sectional ranking, and otherwise use full fine-tuning to give the encoder the maximum capacity to adapt. In practice this yields full fine-tuning for all pretraining baselines, though for two different reasons. PatchTST, MERA, and CI-STHPAN already fine-tune the full encoder

Table 8: Architecture-specific configurations of the GRU, Transformer, and Mamba encoders used in DIVINE.

Configuration	GRU	Transformer	Mamba
Input dimension	5	5	5
Hidden dimension	64	64	64
Number of layers	2	2	2
Attention heads	–	8	–
Attention mask	–	Causal	–
Positional encoding	–	Sinusoidal	–
FFN dimension	–	256	–
State dimension	–	–	16
Convolution width	–	–	4
Expansion factor	–	–	2
Dropout	0.0	0.1	0.1
Representation pooling	Last hidden state	Last token	Last hidden state
Downstream MLP hidden dim.	64	64	64
Downstream encoder update	Frozen	Full fine-tuning	Frozen
Encoder parameters (M)	0.05	0.10	0.07

in their original protocols, so we follow them directly. The remaining methods depart from the ranking objective and prediction head used here, and therefore fall under the second case: TS2Vec and TimesURL evaluate frozen representations with linear or kernel-based heads, FASCL retrieves stocks by embedding similarity, and SSPT selects its freezing strategy by hyperparameter search. All baselines are transferred with a shared MLP head identical to that used in DIVINE. For completeness, we additionally evaluate all pretraining methods under both frozen-encoder transfer and full fine-tuning in Appendix F.2.

Implementation Details and Hyperparameter Settings. Table 9 summarizes the pretraining configuration of all baselines. We follow each method's released configuration where one is specified, and otherwise set the remaining hyperparameters to values consistent with our evaluation setting. Four cases require further explanation. FASCL has no released implementation, so we reproduce it from the paper using the reported hyperparameters. For TimesURL we adopt the effective command-line default learning rate of 10^{-4}, as the repository specifies inconsistent values elsewhere. For PatchTST we disable RevIN, because our input windows are already temporally z-scored and we require identical preprocessing across encoders. Finally, PatchTST and CI-STHPAN both specify long self-supervised context length ($L = 512$) and correspondingly large patch sizes. Since all methods share a 20-day downstream input window in our evaluation, retaining these settings would introduce a substantial mismatch between the pretraining and transfer stages, and would leave only one or two patches per downstream window. We therefore shorten the pretraining context length and use non-overlapping patches of $P = S = 4$.

Table 9: Pretraining configuration of the compared pretraining baselines. Batch size follows each method's own sampling unit, where one day denotes the full stock cross-section of a single trading day.

Method	Context	Emb. dim	Batch size	Learning rate	Training budget	Code
FASCL	64	384	1 day	$10^{-3} \rightarrow 10^{-6}$	50 epochs	—
TS2Vec	64	320	16 windows	10^{-3}	600 iters	GitHub
TimesURL	64	320	8 windows	10^{-4}	600 iters	GitHub
SSPT	32	128	1024 windows	10^{-4}	100 epochs	GitHub
PatchTST	64	1024	64 windows	LR range test	100 epochs	GitHub
MERA	20	128	2 days	10^{-4}	500 epochs	GitHub
CI-STHPAN	20	128	1 day	LR range test	100 epochs	GitHub

D.2 Financial Foundation Model

Pretraining Exposure. The released financial foundation models are pretrained on substantially larger external corpora than the six-market corpus used by DIVINE. Kronos (Shi et al., 2025) is

pretrained on more than 12 billion K-line records from 45 global exchanges through June 2024, resulting in temporal overlap with our 2021–2024 evaluation period. FinCast (Zhu et al., 2025) is pretrained on 2.4 million time series spanning financial and non-financial domains; however, its released documentation does not provide sufficient temporal coverage information to determine whether a similar overlap exists.

Adaptation and Transfer Protocol. For both models, we use the preprocessing pipelines specified in their original publications [2] rather than applying our shared window normalization.

Kronos. We evaluate Kronos-mini, Kronos-small, and Kronos-base using their released pretrained checkpoints. For each downstream market, we follow the released target-market adaptation procedure, selecting the checkpoint with the lowest validation loss on the corresponding training split. We then transfer the adapted sequence representation to the downstream 5-day cross-sectional ranking objective. We select the *timestamp-free representation transfer* configuration as our primary evaluation setting, as it is most aligned with the other baselines in our benchmark: representation transfer mirrors how other pretrained encoders are evaluated (rather than relying on Kronos's native autoregressive interface), and omitting timestamp conditioning removes an informational advantage not available to other methods. For completeness, we additionally evaluate timestamp-conditioned representation transfer and direct autoregressive forecasting; results under all three configurations are reported in Table 10.

FinCast. No target-domain adaptation pipeline is publicly provided; as the repository explicitly supports zero-shot transfer, we initialize directly from the released pretrained checkpoint. Because FinCast is a univariate model, encoding only a subset of input channels would place it at an informational disadvantage relative to other baselines that consume the full OHLCV bar. To fairly leverage all available information without retraining the model, we encode each input channel independently using the pretrained backbone and concatenate the resulting representations before the downstream MLP, using the representation at the final timestep as the model output. We evaluate three channel configurations (C, CV, and OHLCV) to ablate the contribution of each information source; results are reported in Table 11, with the full OHLCV configuration used in Table 1 as the most informationally complete and fairly comparable setting.

Full Results. Tables 10 and 11 report full results under all evaluated configurations for Kronos and FinCast, respectively. For Kronos, representation transfer substantially outperforms direct autoregressive inference, and removing timestamp conditioning—which aligns Kronos with the information available to other baselines—consistently improves performance across all three model scales. For FinCast, incorporating additional channels improves transfer performance, with the full OHLCV variant achieving the best average SR and CR; this is also the most informationally complete configuration and therefore the most comparable to other methods. Notably, the configuration selected for Table 1 in each case is both the most methodologically aligned with other baselines *and* the strongest-performing setting, suggesting our protocol choices do not inadvertently favor or penalize these models.

E Detailed Results of Main Paper

E.1 Per-Indicator Reconstruction Quality (from Section 4.2)

Figure 4 reports per-indicator reconstruction R^2 on both the training and test splits, decomposed by period parameter and market. Overall, the encoder reliably reconstructs the pretraining targets: across the 77 indicator–period combinations and six markets, average test R^2 reaches 0.847, with median train–test gap of 0.008 and all gaps below 0.05, confirming that pretraining converges without severe overfitting.

MFI failure on CSI markets. One systematic exception stands out: MFI reconstruction fails on CSI300 and CSI500 across all period parameters (test $R^2 < 0$), while remaining comparable to other indicators on the four non-Chinese markets. MFI is computed as the ratio of positive money

[2] https://github.com/shiyu-coder/Kronos; https://github.com/vincent05r/FinCast-fts

Table 10: Kronos results across representation-transfer and native autoregressive configurations. "Transfer" denotes representation transfer to the downstream ranking task, "AR" denotes native autoregressive inference, and "w/ stamp" / "w/o stamp" indicate whether timestamp conditioning is used.

Scale	Protocol	CSI300		CSI500		NI225		SP500		FTSE100		TWSE300		Avg.	
		SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
Mini	Transfer, w/o stamp	1.260	1.344	1.637	1.822	0.762	0.592	1.266	1.644	0.522	0.365	1.214	1.085	**1.110**	**1.142**
	Transfer, w/ stamp	0.969	0.843	1.462	1.398	0.788	0.668	0.925	0.942	0.445	0.294	1.094	0.883	0.947	0.838
	AR, w/o stamp	-0.043	0.003	-0.572	-0.162	0.728	0.627	1.340	2.136	0.568	0.413	0.626	0.418	0.441	0.573
	AR, w/ stamp	0.140	0.085	-0.404	-0.123	0.763	0.660	1.391	2.453	0.419	0.327	0.651	0.412	0.493	0.636
Small	Transfer, w/o stamp	1.459	1.516	1.916	2.229	0.903	0.783	1.523	2.329	0.524	0.378	1.209	1.186	**1.256**	**1.404**
	Transfer, w/ stamp	1.171	1.160	1.598	1.565	0.794	0.651	1.224	1.622	0.597	0.423	1.118	0.972	1.084	1.066
	AR, w/o stamp	0.152	0.125	-0.275	-0.095	0.575	0.485	1.268	1.963	0.494	0.371	0.592	0.384	0.468	0.539
	AR, w/ stamp	0.196	0.143	-0.115	-0.030	0.573	0.482	1.255	1.805	0.451	0.339	0.595	0.370	0.492	0.518
Base	Transfer, w/o stamp	1.326	1.305	1.803	1.979	0.895	0.714	1.387	2.040	0.359	0.252	1.154	1.076	**1.154**	**1.228**
	Transfer, w/ stamp	1.034	0.856	1.506	1.328	0.797	0.713	1.045	1.085	0.542	0.370	1.244	1.117	1.028	0.912
	AR, w/o stamp	0.011	0.052	-0.684	-0.161	0.666	0.554	1.345	2.063	0.486	0.342	0.605	0.368	0.405	0.537
	AR, w/ stamp	0.245	0.181	-0.074	-0.007	0.596	0.504	1.329	1.846	0.578	0.445	0.641	0.429	0.553	0.566

Table 11: FinCast representation-transfer results across input-channel configurations. C, CV, and OHLCV denote close-only, close-volume, and full OHLCV inputs, respectively.

Channels	CSI300		CSI500		NI225		SP500		FTSE100		TWSE300		Avg.	
	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
C	0.748	0.584	0.627	0.441	0.689	0.550	0.633	0.506	0.494	0.307	0.797	0.590	0.665	0.496
CV	0.907	0.759	0.729	0.566	0.687	0.533	0.755	0.613	0.493	0.339	0.851	0.594	0.737	0.567
OHLCV	0.780	0.668	1.036	0.819	0.649	0.507	0.699	0.560	0.434	0.274	0.924	0.673	**0.754**	**0.583**

flow to total money flow over the window, and in the CSI markets this ratio concentrates tightly around 0.5, leaving almost no variation for the encoder to reconstruct. The remaining 76 targets are unaffected, with reconstruction quality comparable to the other markets.

Notably, despite this failure, downstream ranking performance on CSI300 and CSI500 remains the strongest among all six markets (Table 1). This dissociation between per-indicator reconstruction quality and downstream utility reinforces the observation from Section 5.3: the hidden state encodes ranking-relevant structure that is not tightly coupled to the reconstruction accuracy of any individual indicator.

E.2 Additional results of Data Efficiency (Section 5.4)

Table 12 and 13 report the average Calmar Ratio (CR) across six markets under varying pretraining-data and downstream-label fractions, complementing the Sharpe Ratio results reported in Section 5.4. The CR results exhibit trends consistent with SR: DIVINE improves as more pretraining data become available and maintains strong performance as the amount of downstream supervision is reduced.

Table 12: Average CR across market by pretrain efficiency

Pretrain Data Portion	10%	25%	50%	75%	Full
From Scratch	–	–	–	–	0.938
TS2Vec	0.916	0.953	0.951	0.917	0.923
PatchTST	1.173	1.206	1.210	1.229	1.203
FASCL	0.863	0.908	0.947	0.927	0.933
MERA	0.988	0.986	0.842	0.955	0.833
DIVINE	1.257	1.535	1.619	1.605	**1.790**

E.3 Additional Results on Period Diversity of Indicators (Section 6.1)

Table 14 reports the per-market SR and CR for each single-period configuration, together with the full multi-period setting. To construct the per-market oracle, we retrospectively select, for each

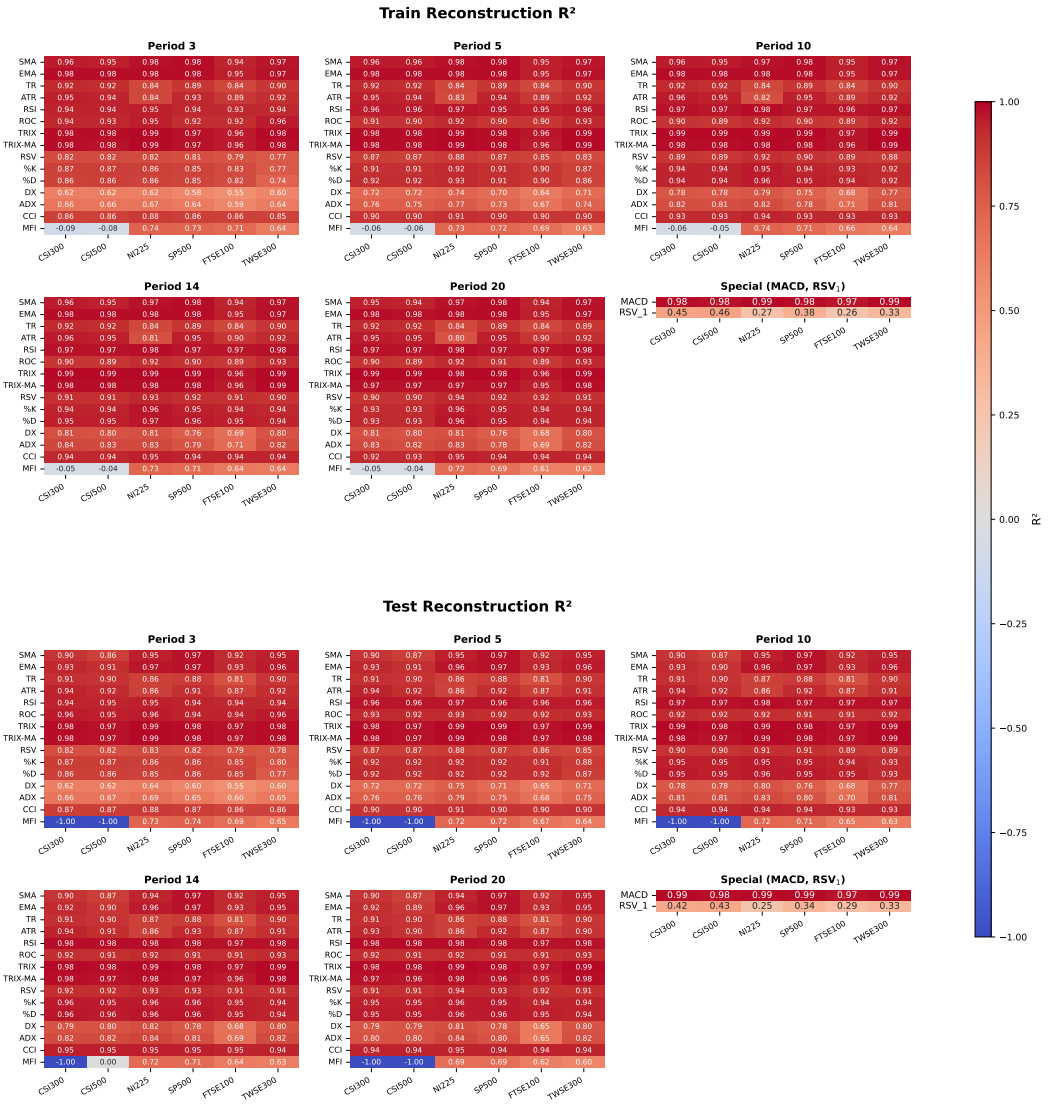

Figure 4: Per-indicator reconstruction R^2 on train (upper block) and test (lower block) splits. Rows correspond to the five period parameters $k \in \{3, 5, 10, 14, 20\}$; the bottom row contains the two special cases (MACD with fixed parameterization, RSV additionally computed at $k = 1$). Each cell shows R^2 for one indicator–market combination. R^2 is clipped to $[-1, 1]$ for visualization.

market and each metric, the best-performing single period among $k \in \{3, 5, 10, 14, 20\}$, and then average the selected values across the six markets. The oracle therefore represents an optimistic upper bound achievable by choosing the period separately for each market with hindsight, rather than a deployable model-selection procedure.

The full multi-period configuration performs comparably to this oracle, supporting the observation in Section 6.1 that combining multiple period scales primarily improves robustness to period choice rather than providing a large additional performance gain.

Table 13: Average CR across market by finetune efficiency

Finetune Data Portion	**1%**	**2%**	**5%**	**10%**	**25%**	**50%**	**75%**	**Full**
From Scratch	0.686	0.670	0.679	0.735	0.818	0.826	0.900	0.938
TS2Vec	0.573	0.535	0.654	0.670	0.785	0.853	0.832	0.923
PatchTST	0.560	0.512	0.619	0.642	1.026	1.155	1.180	1.203
FASCL	0.511	0.507	0.629	0.613	0.684	0.849	0.923	0.933
MERA	0.632	0.537	0.712	0.782	0.822	0.870	0.779	0.833
Kronos-small	0.627	0.662	0.773	0.963	1.107	1.158	1.229	1.404
DIVINE	0.800	0.910	0.987	1.233	1.466	1.652	1.687	**1.790**

Table 14: SR of Technical Indicator at Period k. **Bold** and Underline means the best performance of the column across and within different blocks, respectively.

	CSI300		**CSI500**		**NI225**		**SP500**		**FTSE100**		**TWSE300**		**Avg.**	
Period k	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
3-only	1.094	1.023	1.854	1.926	1.138	0.962	**1.700**	3.694	0.589	0.385	1.189	1.196	1.261	1.531
5-only	1.229	1.183	1.868	1.708	1.014	0.902	1.555	3.407	0.532	0.357	1.373	1.450	1.262	1.501
10-only	1.333	1.473	1.869	1.976	1.153	1.064	1.626	3.633	**0.622**	0.403	1.382	1.519	1.331	1.678
14-only	1.243	1.194	1.893	1.848	**1.335**	**1.236**	1.625	3.773	0.551	**0.419**	1.395	1.475	1.340	1.657
20-only	1.021	0.929	1.846	2.006	1.157	0.998	1.581	3.748	0.507	0.348	1.394	1.525	1.251	1.592
Oracle	1.333	1.473	1.893	2.006	1.335	1.236	1.700	3.773	0.622	0.419	1.395	1.525	**1.402**	1.739
All windows	**1.520**	**1.640**	**1.982**	**2.207**	1.245	1.146	1.645	**3.774**	0.568	0.413	**1.418**	**1.549**	1.396	**1.790**

E.4 Additional Results on Market Composition (Section 6.2)

Table 15 reports the corresponding Calmar Ratio (CR) results under different pretraining-market compositions. The results exhibit a pattern consistent with the Sharpe Ratio (SR) analysis in Section 6.2.

Table 15: CR by market and leave one pretraining market out. **Bold** and Underline means the best performance of the column across and within different blocks, respectively.

Markets	**CSI300**	**CSI500**	**NI225**	**SP500**	**FTSE100**	**TWSE300**	**Avg.**
One Market Only							
CSI300-only	1.041	1.459	0.805	3.488	0.221	1.385	1.397
CSI500-only	1.119	1.625	0.755	3.112	0.272	1.232	1.352
NI225-only	0.482	1.013	0.828	3.760	0.310	1.287	1.280
SP500-only	0.714	1.330	0.714	3.374	0.247	1.464	1.262
FTSE100-only	0.493	0.916	0.762	3.469	0.288	1.363	1.215
TWSE300-only	0.291	0.865	0.771	4.050	0.253	1.357	1.264
Remove One Market							
w/o CSI300	1.466	2.051	1.028	**4.796**	0.361	1.458	**1.860**
w/o CSI500	1.084	1.789	0.965	3.527	0.394	**1.565**	1.554
w/o NI225	1.146	2.041	1.037	4.003	0.407	1.446	1.680
w/o SP500	1.197	**2.400**	1.072	4.104	0.251	1.525	1.758
w/o FTSE100	1.281	2.027	**1.182**	3.659	0.388	1.498	1.672
w/o TWSE300	1.208	2.056	1.070	3.882	0.339	1.390	1.658
Overall							
Single Market	1.041	1.626	0.828	3.374	0.288	1.357	1.419
OOD performance	1.466	1.789	1.037	4.104	0.388	1.390	1.696
Joint All Market	**1.640**	2.207	1.146	3.774	**0.413**	1.549	1.790

F Extra Analysis

F.1 Preprocessing and Normalization Ablations

Table 16 evaluates the normalization stages introduced in Section 4.1 while keeping the remaining training configuration fixed. At the sample level, the default max-scaling achieves the strongest

performance (1.396/1.790 SR/CR). Replacing both feature and target scaling with the z-based alternative remains competitive (1.344/1.645), whereas mismatched transformations can degrade performance, particularly when z-normalized features are paired with max-normalized targets (0.890/0.818). These results motivate applying compatible sample-level transformations to the input and scale-dependent indicator targets.

Per-market target standardization has a larger effect on optimization: removing it lowers average SR to 0.988, while removing IQR clipping alone has a smaller effect (1.296). Removing both operations substantially degrades performance to 0.601. The cross-market balancing mechanisms provide more moderate improvements: performance remains at 1.316 without balanced market sampling and 1.345 without pooled re-standardization. Notably, both settings remain above single-market pretraining (1.183; Table 2), indicating that the cross-market gain is not solely explained by these balancing operations.

Table 16: Ablation of preprocessing and normalization choices. All results report average performance across six markets.

Component	Variant	Avg. SR	Avg. CR
Default	DIVINE	1.396	1.790
Sample-level normalization	feat z-norm, target z-norm	1.344	1.645
	feat z-norm, target max-norm	0.890	0.818
	feat max-norm, target z-norm	1.312	1.523
Per-market target normalization	w/o standardization	0.988	1.342
	w/o IQR clipping	1.296	1.751
	w/o both	0.601	0.603
Cross-market training	w/o balanced market sampling	1.316	1.692
	w/o pooled re-standardization	1.345	1.676

We also vary the historical context available during pretraining. Increasing the context from 20 to 60 and 120 bars improves average SR/CR from 1.259/1.482 to 1.300/1.676 and 1.396/1.790, respectively. Note that this context length is distinct from the indicator period parameters analyzed in Section 6.1 and from the fixed downstream input window. Pretraining context length is not directly comparable across methods: the pretraining baselines use context length ranging from 20 to 64 bars, while ‡Kronos and †FinCast are pretrained with 90 and 128 bars, respectively. Even at DIVINE's shortest setting of 20 bars, which matches the downstream input window, it already exceeds every pretraining baseline in Table 1 and matches ‡Kronos-small, indicating that its advantage does not stem from a longer pretraining context.

F.2 Robustness to Downstream Transfer Protocol

Different pretraining methods may exhibit different downstream adaptation behavior. To complement the transfer settings used in the main comparison, we re-evaluate all generic and financial pretraining methods under two shared downstream protocols: (i) frozen-encoder transfer, where only the common two-layer MLP prediction head is optimized, and (ii) full fine-tuning, where both the pretrained encoder and prediction head are updated. The downstream ranking objective, data splits, prediction head, and evaluation protocol are otherwise kept unchanged. We do not select between the two protocols on a per-method basis. Table 17 reports the corresponding market-wise results. Under frozen-encoder transfer, all three DIVINE backbones achieve higher average SR and CR than every compared pretraining baseline. The same pattern holds under full fine-tuning. The effect of adaptation itself is method-dependent—some pretrained encoders improve with full fine-tuning, whereas others perform better when frozen—but DIVINE remains strongest on average under either common protocol. These results show that the main performance advantage is robust to the choice of downstream adaptation strategy and is not explained by encoder freezing alone.

F.3 Standard Deviations of Average Results

Table 18 complements Table 1 by reporting the mean and standard deviation of the average SR and CR across the 25 seed–offset evaluations. For each seed–offset combination, SR and CR are first averaged across the six markets, and the standard deviation is then computed over the resulting 25 average values.

Table 17: Robustness to downstream transfer protocol. We report portfolio performance under frozen-encoder transfer and full fine-tuning across six markets.

Markets →		CSI300		CSI500		NI225		SP500		FTSE100		TWSE300		Avg.	
Method	**Transfer**	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
General Time-Series Pretraining															
TS2Vec	Frozen	0.927	0.744	1.210	1.160	0.733	0.582	0.914	0.832	0.545	0.371	1.204	1.175	0.922	0.811
	Full FT	1.014	0.893	1.522	1.640	0.780	0.664	0.952	1.008	0.493	0.344	1.140	0.989	0.983	0.923
PatchTST	Frozen	0.584	0.407	0.859	0.648	0.744	0.616	1.580	2.901	0.533	0.369	0.945	0.829	0.874	0.962
	Full FT	0.662	0.533	1.147	1.113	0.807	0.653	1.618	3.117	0.437	0.334	1.371	1.465	1.007	1.202
TimesURL	Frozen	1.100	1.140	1.479	1.616	0.584	0.472	1.068	1.306	0.496	0.340	1.024	0.944	0.959	0.970
	Full FT	1.022	0.972	1.302	1.277	0.576	0.448	1.057	1.191	0.412	0.268	1.117	0.958	0.914	0.852
Financial Pretraining															
MERA	Frozen	0.729	0.554	1.283	1.277	1.328	1.192	1.459	2.870	0.346	0.281	1.222	1.102	1.061	1.213
	Full FT	1.075	1.055	0.677	0.472	0.769	0.554	1.117	1.131	0.704	0.469	1.116	0.955	0.962	0.833
CI-STHPAN	Frozen	0.426	0.352	1.082	0.821	0.591	0.450	1.719	3.697	0.313	0.221	0.409	0.339	0.757	0.980
	Full FT	0.720	0.518	0.949	0.646	0.286	0.261	1.527	2.075	0.705	0.553	0.520	0.423	0.785	0.746
FASCL	Frozen	0.690	0.597	0.964	0.831	1.044	0.878	1.072	1.337	0.514	0.355	0.929	0.800	0.869	0.800
	Full FT	0.547	0.485	0.951	0.881	0.954	0.742	1.300	1.713	0.718	0.504	1.246	1.275	0.953	0.933
SSPT	Frozen	0.754	0.604	0.870	0.669	0.938	0.771	0.867	1.285	0.585	0.378	0.908	0.701	0.820	0.735
	Full FT	0.447	0.404	0.931	0.724	1.079	0.896	1.162	2.035	0.540	0.362	1.095	0.938	0.876	0.893
DIVINE (Ours)															
GRU encoder	Frozen	1.520	1.640	1.982	2.207	1.245	1.146	1.645	3.774	0.568	0.413	1.418	1.549	1.396	1.790
	Full FT	1.130	1.072	1.597	1.564	1.164	0.962	1.511	4.037	0.645	0.472	1.397	1.413	1.241	1.587
Mamba encoder	Frozen	1.083	1.183	1.675	1.841	1.075	0.961	1.807	3.519	0.645	0.425	1.273	1.336	1.259	1.544
	Full FT	1.344	1.385	1.704	1.513	1.034	0.913	1.071	3.066	0.666	0.568	1.406	1.434	1.204	1.479
TFM encoder	Frozen	1.045	0.939	1.512	1.498	1.170	0.989	1.516	3.144	0.323	0.161	0.994	1.039	1.093	1.295
	Full FT	1.383	1.371	1.811	1.813	1.193	0.970	1.456	3.586	0.646	0.446	1.510	1.835	1.333	1.670

Table 18: Average portfolio performance with standard deviations across 25 seed–offset evaluations. Values report mean ± standard deviation for SR and CR averaged across the six equity markets.

Method	**Param. (M)**	Avg. SR	Avg. CR
Financial models trained from scratch			
RankLSTM	0.05	0.768 $\pm$0.216	0.772 $\pm$0.135
FinGAT	0.04	0.764 $\pm$0.129	0.595 $\pm$0.148
MASTER	0.7	0.898 $\pm$0.158	0.950 $\pm$0.227
AlphaMix	0.04	0.873 $\pm$0.112	0.931 $\pm$0.191
StockMixer	0.01	0.647 $\pm$0.099	0.518 $\pm$0.102
DHMoE	10.2	0.902 $\pm$0.166	0.901 $\pm$0.149
General Time-Series Pretraining			
TS2Vec	0.6	0.983 $\pm$0.123	0.923 $\pm$0.180
PatchTST	0.6	1.007 $\pm$0.085	1.202 $\pm$0.209
TimesURL	0.6	0.914 $\pm$0.141	0.852 $\pm$0.199
Financial Pretraining			
MERA	1.5	0.962 $\pm$0.127	0.833 $\pm$0.209
CI-STHPAN	0.5	0.785 $\pm$0.103	0.746 $\pm$0.223
FASCL	14.2	0.953 $\pm$0.146	0.933 $\pm$0.190
SSPT	0.4	0.876 $\pm$0.167	0.893 $\pm$0.235
Financial Foundation Models			
FinCast	991.4	0.754 $\pm$0.104	0.583 $\pm$0.112
Kronos-mini	5.3	1.110 $\pm$0.109	1.142 $\pm$0.198
Kronos-small	24.5	1.256 $\pm$0.134	1.404 $\pm$0.228
Kronos-base	99.7	1.154 $\pm$0.162	1.228 $\pm$0.252
DIVINE (Ours)			
GRU encoder	0.05	1.396 $\pm$0.092	1.790 $\pm$0.266
Mamba encoder	0.07	1.259 $\pm$0.100	1.544 $\pm$0.160
TFM encoder	0.1	1.333 $\pm$0.096	1.670 $\pm$0.302

F.4 Main Results with Transaction Cost

We evaluate robustness to transaction costs by incorporating market-specific brokerage commissions, exchange fees, and regulatory charges. Per-side (buy/sell) rates are approximately: CSI300 & CSI500 0.006%/0.056%, NI225 0.002%/0.002%, S&P500 negligible ($\sim$0%), FTSE100 0.55%/0.05%, and TWSE300 0.1425%/0.4425%. At each rebalance, costs are charged on the actual change in portfolio weights rather than full turnover, making Table 19 a realistic rather than worst-case estimate.

In Table 19, we compare DIVINE with the five strongest methods from Table 1. Results show that DIVINE maintains the highest average SR/CR of 1.067/1.483 under these conditions, demonstrating robustness to realistic trading frictions. The exception is FTSE100, where the relatively high per-side costs cause all methods to yield negative SR/CR, consistent with the already modest pre-cost performance observed in Table 1.

Table 19: Portfolio backtest results of best performing methods in Table 1 with transaction cost. Best and second-best performance per column are highlighted in **bold** and underline, respectively.

Markets →	**CSI300**		**CSI500**		**NI225**		**SP500**		**FTSE100**		**TWSE300**		**Avg.**	
Method	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR	SR	CR
TS2Vec	0.897	0.735	1.427	1.483	0.772	0.656	0.952	1.008	-0.814	-0.204	0.175	0.125	0.568	0.634
PatchTST	0.560	0.432	1.054	0.981	0.799	0.646	1.618	3.117	-0.783	-0.204	0.598	0.470	0.641	0.907
MERA	0.988	0.996	0.591	0.420	0.763	0.550	1.117	1.131	-0.539	-0.166	0.408	0.353	0.555	0.547
FASCL	0.457	0.395	0.873	0.785	0.947	0.736	1.300	1.713	**-0.471**	**-0.147**	0.529	0.372	0.606	0.642
Kronos-small	1.347	1.342	1.825	2.061	0.895	0.775	1.523	2.329	-0.935	-0.220	0.265	0.176	0.820	1.077
DIVINE	**1.424**	**1.487**	**1.904**	**2.070**	**1.239**	**1.139**	**1.645**	**3.774**	-0.541	-0.168	**0.733**	**0.595**	**1.067**	**1.483**

F.5 Analysis of IC and RankIC Metrics

We report IC-based metrics in addition to our primary portfolio metrics to provide a more comprehensive evaluation. Our primary evaluation follows standard practice in quantitative finance, where portfolio construction selects only the top-k ranked stocks, making Sharpe Ratio (SR) and Calmar Ratio (CR) the more direct measures of practical model utility. IC, by contrast, evaluates the quality of the *entire* ranking distribution; a model with strong overall ranking correlation does not necessarily place the most profitable stocks at the top—a fundamental mismatch with the top-k selection regime.

This discrepancy is evident when comparing cross-market averages across Tables 1 and 20: ‡Kronos-small and PatchTST achieve average SR/CR of 1.256/1.404 and 1.007/1.202, respectively, yet PatchTST outperforms ‡Kronos-small on IC-based metrics despite underperforming in portfolio returns. The same divergence appears within individual markets. On TWSE300, PatchTST attains the highest IC and RankIC while DIVINE achieves the higher portfolio SR/CR (1.418/1.549 vs. 1.371/1.465), illustrating that stronger overall rank correlation need not translate into better top-k selection. On FTSE100, the market where DIVINE is weakest in portfolio terms (Section 5.2), IC-based metrics likewise rank DIVINE below several baselines, consistent with the limitation discussed there.

Averaged across markets, DIVINE nonetheless ranks first on both IC and RankIC, indicating that its portfolio advantage does not come at the cost of overall ranking quality. We report these metrics for completeness while noting that per-market divergences between the two metric families reinforce our use of SR and CR as the primary evaluation criteria.

Table 20: Ranking-based performance across six markets. Best and second-best performance per column are highlighted in **bold** and underline, respectively.

Market	Method	IC	ICIR	RankIC	RankICIR
CSI300	DIVINE	**0.0503**	**0.3712**	**0.0437**	**0.3446**
	MERA	0.0323	0.2358	0.0298	0.2286
	PatchTST	0.0427	0.3053	0.0379	0.2930
	FASCL	0.0218	0.1938	0.0188	0.1994
	TS2Vec	0.0309	0.2871	0.0305	0.2993
	Kronos-small	0.0328	0.2737	0.0306	0.2738
CSI500	DIVINE	**0.0753**	**0.6191**	**0.0661**	0.5631
	MERA	0.0582	0.4815	0.0511	0.4551
	PatchTST	0.0663	0.5364	0.0589	0.5128
	FASCL	0.0591	0.5373	0.0459	0.5174
	TS2Vec	0.0677	0.5968	0.0595	**0.6128**
	Kronos-small	0.0608	0.5495	0.0558	0.5712
NI225	DIVINE	**0.0138**	**0.1073**	**0.0130**	**0.1110**
	MERA	0.0092	0.0815	0.0096	0.0884
	PatchTST	0.0043	0.0394	0.0072	0.0657
	FASCL	0.0083	0.0854	0.0084	0.0975
	TS2Vec	0.0060	0.0692	0.0076	0.0859
	Kronos-small	0.0060	0.0667	0.0076	0.0832
S&P500	DIVINE	**0.0352**	0.2392	**0.0192**	**0.1476**
	MERA	0.0231	0.1592	0.0116	0.0792
	PatchTST	0.0275	0.2105	0.0149	0.1346
	FASCL	0.0155	0.1317	0.0093	0.1185
	TS2Vec	0.0032	0.0383	0.0050	0.0661
	Kronos-small	0.0214	**0.2417**	0.0120	0.1277
FTSE100	DIVINE	0.0084	0.0529	0.0094	0.0632
	MERA	0.0072	0.0477	0.0079	0.0542
	PatchTST	0.0098	0.0692	0.0099	0.0723
	FASCL	**0.0143**	**0.1014**	**0.0129**	**0.0966**
	TS2Vec	0.0067	0.0527	0.0085	0.0672
	Kronos-small	0.0075	0.0597	0.0083	0.0644
TWSE300	DIVINE	0.0319	0.2391	0.0402	0.3426
	MERA	0.0279	0.2128	0.0406	0.3325
	PatchTST	**0.0324**	**0.2639**	**0.0424**	**0.3696**
	FASCL	0.0267	0.2168	0.0363	0.3403
	TS2Vec	0.0203	0.2141	0.0255	0.2711
	Kronos-small	0.0242	0.2208	0.0311	0.3055
Average	DIVINE	**0.0358**	**0.2715**	**0.0319**	**0.2620**
	MERA	0.0263	0.2031	0.0251	0.2063
	PatchTST	0.0305	0.2375	0.0285	0.2413
	FASCL	0.0243	0.2111	0.0219	0.2283
	TS2Vec	0.0225	0.2097	0.0228	0.2337
	Kronos-small	0.0255	0.2354	0.0242	0.2376