Distilling Time Series Reasoning Into Small Language Models

Werner Laemlin [1] **Philippe Helluy** [1] **Svitlana Vyetrenko** [2 1]

Abstract

In this paper, we investigate the distillation of time series reasoning and anomaly detection capabilities into small, instruction-tuned language models. Leveraging a synthetic dataset of time series with various types of anomaly, we generate natural language annotations using a large multimodal model and use these to supervise the fine-tuning of compact *Qwen* models. We introduce evaluation metrics that assess the numerical precision as well as the quality of the distilled reasoning on anomaly detection tasks and show that fine-tuned models demonstrate strong anomaly detection and explanation capabilities, outperforming *GPT-4o*, while requiring significantly fewer computational resources and enabling broader practical deployment.

1. Introduction

1.1. Problem statement

Recent research has shown that small models can achieve performance comparable to much larger models when trained on carefully curated datasets. For instance, (Gunasekar et al., 2023) demonstrate that high-quality, instruction-tuned data can dramatically improve model capabilities without scaling up parameters. Similarly, (Eldan & Li, 2023) highlights that small language models trained on compact, well-structured data can generate coherent and creative narratives. This principle is further reinforced by advancements like `DeepSeek R1` (DeepSeek-AI et al., 2025), where focused data selection and training techniques enable relatively lightweight models to perform tasks traditionally reserved for larger architectures. Together, these studies suggest that data quality, not just model size, is a critical factor in achieving strong language model performance.

Time series data is ubiquitous across healthcare, finance, and industrial systems, where real-time, on-device analysis is essential for operational efficiency and timely decision-making. Models in these settings must address a range of tasks, including forecasting, anomaly detection, trend analysis, and event classification. Beyond performance and compactness, it is increasingly important for such models to be interpretable—able to explain temporal patterns and anomalies in natural language that aligns with human reasoning. Despite its significance, this direction remains largely unexplored. In this work, we take a step toward this goal by introducing a method for distilling anomaly detection into small, interpretable models capable of generating natural language explanations.

1.2. Related work

Recent work has investigated the application of general-purpose language models to time series tasks. Gruver and Wilson (Gruver et al., 2024) first demonstrated that pre-trained language models can achieve competitive zero-shot performance on standard forecasting benchmarks. Building on this, Jin et al. (Jin et al., 2024) proposed a broader framework in which a variety of time series tasks are reformulated as token prediction problems within a language modeling paradigm. In parallel, specialized foundational models pre-trained exclusively on time series data have been introduced. Chronos (Ansari et al.), Llaglama (Rasul et al., 2024), and TimesFM (Das et al., 2024) focus on large-scale pretraining for forecasting, achieving strong zero-shot and few-shot performance across diverse temporal datasets. Moirai (Woo et al., 2024) extends these methods to multivariate forecasting, capturing complex dependencies among multiple correlated series. Multitask models such as UniTS (Gao et al., 2024) and Moment (Goswami et al., 2024) further generalize this approach by jointly training across forecasting, imputation, and classification tasks, illustrating the advantages of shared temporal representations.

Systematic evaluations based on the series feature taxonomy proposed in (Fons et al., 2024) benchmark general-purpose language models on a range of time series understanding tasks, highlighting both their strengths and limitations relative to specialized models. TimeSeriesExam (Cai et al.,

[1]University of Strasbourg, France [2]Outsampler, Strasbourg, France. For correspondence email: Svitlana Vyetrenko <svitlana@outsampler.com>. Code needed to replicate results of this paper is available at: `https://github.com/wlemalin/DistillAnomalyTS`

International Conference on Machine Learning (ICML) 2025 Workshop on Foundation Models for Structured Data, Vancouver, Canada. PMLR 267, 2025.

2024) introduces a complementary benchmark focused on evaluating language models' ability to perform detailed reasoning over temporal patterns. In parallel, work in (Daswani et al., 2024) extends language model-based methods to the interpretation of visual temporal data, including line and bar charts. Together, these efforts reflect a broader shift toward treating time series and structured data as instances of language modeling.

1.3. Our contributions

1. We propose a framework for **interpretable time series reasoning model** construction via knowledge distillation, enabling small language models to detect anomalies in time series data and to explain them in natural language.

2. We introduce a **practical and reproducible distillation pipeline** that transfers time series reasoning skills, in particular anomaly detection and anomaly explanation — from large multimodal models to compact ones.

2. Distillation of time series reasoning

2.1. Background

Distillation is the process of transferring knowledge from a large, often complex model (the teacher) to a smaller, more efficient model (the student) that can understand and reason about a narrow topic (Xu et al., 2024; Hinton et al., 2015). This technique aims to retain the performance of the larger model while reducing computational requirements, making deployment more practical in real-time or resource-constrained settings. In addition, practical benefits can include the ability to fine-tune smaller models with sensitive data.

Recent advances in distilling reasoning—particularly mathematical inference—into small models have demonstrated that compact architectures can tackle complex tasks with surprising accuracy(Team, 2025; DeepSeek-AI et al., 2025). Distillation of time series reasoning has not been explored yet in the literature. In this paper, we explore this capability and demonstrate that small models can be effectively trained via distillation.

2.2. Synthetic dataset generation and annotation

Building on the anomaly taxonomy proposed by (Zhou & Yu, 2025), we construct a synthetic dataset encompassing seven distinct anomaly types. We generate time series captions by prompting a large model `GPT-4o` to generate a JSON object that contains the detected anomaly start and end points as well as a natural language description of said anomaly, then we filter those annotations to ensure quality of the training data, keeping only the detections that overlap ground truth anomaly.

As argued in (Zhou & Yu, 2025), large language models generally perform better in time series reasoning when processing time series data as images rather than text tokens; therefore, we use both numerical and visual time series presentations in order to generate high-quality time series annotations by a large model.

Then we use the filtered datasets to fine-tune Qwen2.5-1.5b & Qwen2.5-3b-VL. We evaluate both textual and visual input modalities and demonstrate that a small model can approach the performance of much larger teacher models when appropriately trained.

2.3. Evaluation metrics

In order to evaluate the quality of the dataset and, subsequently, the quality of the post-trained small models, we propose the following evaluation metrics:

1. **Affiliation metrics**, following Huet et al. (2022), we evaluate temporal proximity between a predicted set and the ground truth interval using the affiliation framework, which normalizes distances against a random baseline. In our setting, each series contains at most one ground truth anomaly interval $r = [a, b)$. Let the series domain be $I = [A, B)$ and let the prediction be the interval $\hat{r} = [\hat{a}, \hat{b})$.

   Directed distances. For a time point x and a set Y, define $d = \mathrm{dist}(x, Y) = \min_{y \in Y} |x - y|$. Distances are used in two directions: $\hat{r} \to r$ (precision) and $r \to \hat{r}$ (recall).

   Zone-normalized survival mapping. Distances are converted to probabilities in $[0, 1]$ by comparing them to a uniformly random timestamp drawn from the affiliation zone, which, with a single ground truth event, equals the series domain I. The closed-form survival functions are:

$$F_{\text{prec}}(d) \;=\; 1 - \frac{|r| + \min(d,\, a - A) + \min(d,\, B - b)}{|I|},$$

$$F_{\text{rec}}(d;\, y) \;=\; 1 - \frac{\min(d,\, y - A) + \min(d,\, B - y)}{|I|},$$

   with the convention $F(0) = 1$. Here F_{prec} scores a predicted timestamp at distance d from r, and F_{rec} scores a GT timestamp $y \in r$ at distance d from $\hat{\mathrm{r}}$.

   Affiliation precision and recall (single-event). Averaging the survival values yields proximity probabilities:

$$P_{\text{aff}} \;=\; \frac{1}{|\hat{\mathrm{r}}|} \int_{x \in \hat{\mathrm{r}}} F_{\text{prec}}\big(\mathrm{dist}(x, r)\big)\, dx,$$

$$R_{\text{aff}} = \frac{1}{|r|}\int_{y\in r} F_{\text{rec}}\big(\text{dist}(y,\hat{\text{r}});\, y\big)\, dy.$$

Affiliation F1. We summarize proximity with the harmonic mean

$$F1_{\text{aff}} = \frac{2\, P_{\text{aff}}\, R_{\text{aff}}}{P_{\text{aff}} + R_{\text{aff}}}$$

The scale is calibrated by the zone: 0.5 corresponds to random placement in I, while values approaching 1 indicate tight temporal alignment.

We compute $F1_{\text{aff}}$ per series and report its macro-average (i) within each anomaly type (per-type scores) and (ii) across all series (global macro-average).

2. **BLEURT** is a learned evaluation metric based on pretrained contextual embeddings fine-tuned on human judgment data (Sellam et al., 2020). It provides a scalar similarity score between a generated anomaly description and the corresponding ground truth description, capturing semantic adequacy and fluency beyond surface lexical overlap.

3. **BERTScore** evaluates text similarity by aligning tokens between a generated description and the ground truth description using contextual embeddings from a pretrained transformer model (Zhang et al., 2020). The metric assigns similarity scores at the token level and aggregates them into a scalar value reflecting semantic overlap. We compute BERTScore per description and report the macro-average across all samples.

4. **ROUGE-L** evaluates the longest common subsequence between the predicted and reference text. It is a simple lexical metric that rewards overlap in sequence structure, giving insight into how much of the reference wording is reused.

2.4. Small model training

We filter the annotations by removing LLM annotation that don't overlap the ground truth anomaly at all, and we use the curated datasets to fine-tune `Qwen2.5-1.5B-Instruct` and `Qwen2.5-VL-3B-Instruct` models (Qwen et al., 2025).

For the first model, we experimented with either giving (i) only the raw signal of the time series *(TS1)*, (ii) providing additional information with moving average and moving standard deviation *(TS3)*, or (iii) adding to the three previous values the local frequency, extracted from the short-time Fourier transform spectrogram *(TS4)*. For the VL model, we always provide as text, the raw signal as well as the moving

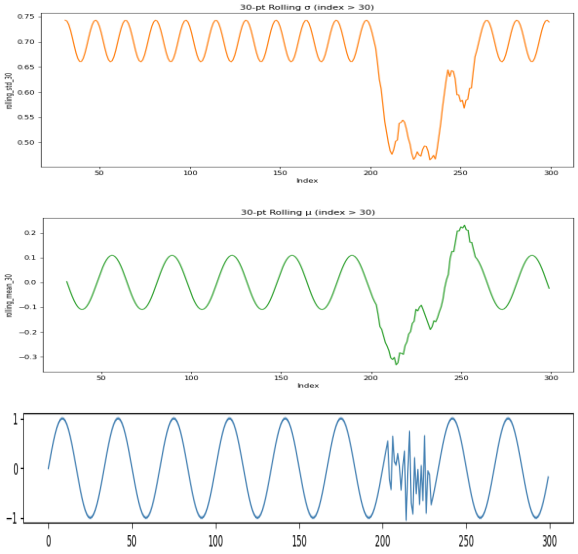

Figure 1. Time series raw signal (blue), moving average (green), moving standard deviation (orange).

PROMPT: Given the time series, determine whether there is an anomalous interval. If the series is entirely normal, return the empty JSON template. Otherwise, detect and describe the nature of the anomaly.

Return only a JSON object formatted exactly as follows, with no extra keys or text:

```
{
  "anomalies": [
    {
      "start": int,
      "end": int,
      "description": string
    }
  ]
}
```

Time series:

`timestamp : value, avg, std...`

Figure 2. Prompt used to annotate time series using GPT-4o.

ANNOTATION:

```
{
  "anomalies": [
    {
      "start": 204,
      "end": 226,
      "description": "The series
deviates from a sine wave pattern,
showing irregular fluctuations."
    }
  ]
}
```

Figure 3. Example of time series annotation generated by GPT-4o.

average and standard deviation as it is the configuration that lead to better results when working with text, additionally we provide the pictures of the raw signal *(TS1)*, then we add the moving average and moving standard deviation *(TS3)*, and in our last experiment we also give the short time Fourier transform spectrogram *(TS4)*.

It should be noted that tokenization of numerical values poses challenges for large language models (LLMs), as most tokenizers are optimized for natural language and not for numeric precision (Spathis & Kawsar, 2023). Decimal numbers, scientific notation, and long integers are often split into multiple tokens inconsistently, which can hinder model performance on tasks involving quantitative reasoning. To ensure consistent tokenization, we rescale and round all floating numbers obtained by the time series generation process to integers with left padding from 00 to 99, and ensure that each digit is represented as a single token (Yuan et al., 2023).

2.5. Experimental results and evaluation

To evaluate the quality of time series reasoning that we transfer to small models `Qwen2.5-1.5B-Instruct` and `Qwen2.5-VL-3B-Instruct` through the distillation process, we evaluate the annotations they generate after post-training both for numerical precision and for quality of reasoning using the evaluation metrics described in Section 2.3.

2.5.1. In sample:

As evident from Table 1, the results show clear differences in anomaly detection performance across the evaluated models. As expected, 4o, which served as the annotation source, achieves a strong overall performance, reflecting its alignment with the ground truth. Among the fine-tuned Qwen models, Qwen-TS3 slightly outperforms 4o in overall F1 score (0.9461), indicating that augmenting the raw time series with rolling mean and standard deviation features provides valuable context for anomaly localization. By contrast, the simpler Qwen-TS (trained on just the raw values) underperforms, confirming that relying solely on the raw signal limits generalization. Interestingly, Qwen-TS4, which incorporates centroids extracted from STFT spectrograms in addition to rolling statistics, shows degraded performance, suggesting that the frequency-domain centroid representation may have introduced additional complexity without improving alignment with the annotations.

Table 2 show that incorporating visual inputs can substantially enhance anomaly detection when combined with rich textual features (*values + moving average + moving standard deviation*). While, 4o achieves a strong baseline, all Qwen-VL fine-tuning models surpass it for in sample data. With only one image (TS), Qwen-VL already outperforms 4o, indicating that even minimal visual context provides complementary information to the text features. Increasing to three images (TS3) maintains comparable overall performance which suggests that the additional features may not provide further benefit when they are already represented in the textual modality. The best results are obtained with four images (TS4), which yields the highest overall score and strong performance across nearly all anomaly types. These findings suggest that visual embeddings, when aligned with textual representations, consistently boost the performance of small multimodal LLMs

Type	**GPT-4o**	**Qwen** TS	**Qwen** TS3	**Qwen** TS4
freq	**0.9748**	0.8197	0.9714	0.4867
point	**0.9816**	**0.9732**	0.9713	0.6145
range	**0.9950**	0.9015	0.9778	0.9006
trend	0.8625	0.3727	**0.8864**	0.6697
noisy-freq	**0.9669**	0.5346	0.9571	0.5594
noisy-point	0.9490	0.8451	**0.9654**	0.7760
noisy-trend	0.8550	0.5680	0.8090	**0.8307**
Overall	**0.9450**	0.7401	**0.9461**	0.7184

Table 1. Affiliation F1: Evaluation results after post-training of Qwen2.5–1.5B-Instruct (7 epochs for each training). Note that prior to training, Qwen is unable to produce valid answers.

Type	**GPT-4o**	**Qwen-VL** TS (12 ep.)	**Qwen-VL** TS3 (14 ep.)	**Qwen-VL** TS4 (15 ep.)
freq	0.9710	**0.9821**	0.9722	0.9799
point	0.9787	0.9785	0.9781	**0.9800**
range	0.9845	0.9844	0.9870	**0.9916**
trend	0.8820	0.9168	0.9644	**0.9670**
noisy-freq	0.9453	**0.9870**	0.9703	0.9715
noisy-point	0.9765	0.9826	**0.9835**	0.9734
noisy-trend	0.8102	0.9287	**0.9333**	0.9233
Overall	0.9590	0.9758	0.9758	**0.9786**

Table 2. Affiliation F1: Evaluation results after post-training of Qwen2.5-VL-3B-Instruct

2.5.2. Out sample:

The out-of-sample evaluation (Table 3) shows that GPT-4o maintains strong precision but suffers from low recall, indicating conservative predictions that miss many true anomalies. By contrast, our fine-tuned Qwen models achieve more balanced trade-offs: QwenVL-TS3 and QwenVL-TS4 reach the highest overall F1 scores, demonstrating that small multimodal models can generalize competitively beyond their training data.

Semantic similarity metrics (Table 4) further reveal that

the text-only `Qwen-TS3` produces the most faithful and coherent explanations, outperforming both GPT-4o and the multimodal variants. This suggests that while multimodal fine-tuning enhances anomaly detection robustness, unimodal fine-tuning yields superior textual alignment with ground-truth rationales.Together, these results indicate that fine-tuned small models can rival GPT-4o in out-of-sample settings.

Model	Precision	Recall	F1
Mistral	0.6096	0.4922	0.5893
GPT-4o	**0.7341**	0.4359	0.6973
Qwen-TS3	0.6779	0.6225	0.6413
Qwen-TS4	0.7199	0.6493	0.6758
QwenVL-TS3	0.6895	**0.7693**	0.7178
QwenVL-TS4	0.7140	0.7400	**0.7189**

Table 3. Comparison of model outputs based on Precision, Recall, and F1 metrics.

Model	BLEURT	BERTScore	ROUGE-L
Mistral	0.3602	0.8856	0.1438
GPT-4o	0.3922	0.8730	0.1286
Qwen-TS3	**0.5029**	**0.8975**	**0.2970**
QwenVL-TS3	0.4684	0.8913	0.1487

Table 4. Comparison of model outputs based on BLEURT, BERTScore, and ROUGE-L metrics. Here Qwen and QwenVL represent the best fine-tuned models we trained.

3. Conclusion

We presented a practical framework for distilling *time series reasoning* with a focus on anomaly localization and natural-language explanation, from a large multimodal teacher into a compact, instruction-tuned student model. Our pipeline couples synthetic data generation with high-quality teacher annotations and evaluates the resulting students using a set of metrics that jointly probe numerical fidelity and explanatory quality.

We showed that small `Qwen` models (1.5B text and 3B multimodal) can acquire meaningful anomaly detection and explanation capabilities after distillation. Text-only students benefit from lightweight statistical context (rolling mean/std), while multimodal students consistently improve when visual inputs are aligned with textual representations, achieving strong $F1_{\text{aff}}$ across diverse anomaly types. These findings indicate that interpretable time series capabilities can be compressed into small, deployable models without losing alignment to ground truth, and that carefully chosen features/modality pairing matters more than raw parameter count.

4. Acknowledgements

This work has been supported by the Institut Thematique Interdisciplinaire IRMIA++ at the University of Strasbourg (https://irmiapp.unistra.fr/) and the Gutenberg Circle.

5. Impact Statement

This work advances methods for distilling time series reasoning capabilities into compact language models. The resulting models may broaden access to anomaly detection and explanation in settings where computational resources are limited, including healthcare monitoring, financial risk assessment, and industrial control systems. By coupling anomaly localization with natural-language explanations, these models have the potential to support more transparent and interpretable decision-making in real-world deployments.

Appendix: Additional Results

	GPT-4o			Qwen-TS3			QwenVL-TS3		
Anomaly Type	**ROUGE**	**BLEURT**	**BERTScore**	**ROUGE**	**BLEURT**	**BERTScore**	**ROUGE**	**BLEURT**	**BERTScore**
freq	0.2411	0.3984	0.8924	0.3353	0.4449	0.9083	0.2965	0.5008	0.9140
noisy-freq	0.2577	0.3502	0.8916	0.2887	0.4369	0.9049	0.2553	0.4278	0.9046
noisy-point	0.0963	0.2924	0.8621	0.1395	0.2877	0.8627	0.1813	0.4243	0.8907
noisy-trend	0.1128	0.3705	0.8799	0.0987	0.4183	0.8835	0.2284	0.5068	0.8991
point	0.1864	0.4621	0.8892	0.1096	0.3740	0.8695	0.2543	0.5585	0.9041
range	0.4451	0.5977	0.9225	0.5841	0.6717	0.9400	0.6239	0.6742	0.9407
trend	0.1571	0.3913	0.8864	0.0952	0.3974	0.8807	0.1905	0.4726	0.8913

Table 5. Comparison of model outputs based on BLEURT, BERTScore, and ROUGE-L metrics.

Type	**GPT-4o**	**Qwen-VL** TS pic-only (12 ep)	**QwenVL** TS3 pic-only (14 ep, pic)	**QwenVL** TS4 pic-only (15 ep, pic)	**QwenVL** TS (12 ep)	**QwenVL** TS3 (14 ep)	**QwenVL** TS4 (15 ep)
freq	0.9710	0.8204	0.9379	0.8143	**0.9821**	0.9722	0.9799
point	0.9787	0.7077	0.8360	0.8162	0.9785	0.9781	**0.9800**
range	0.9845	0.9113	0.8501	0.8202	0.9844	0.9870	**0.9916**
trend	0.8820	0.9033	0.9039	0.8570	0.9168	0.9644	**0.9670**
noisy-freq	0.9453	0.5570	0.8168	0.8868	**0.9870**	0.9703	0.9715
noisy-point	0.9765	0.8278	0.8442	0.7954	0.9826	**0.9835**	0.9734
noisy-trend	0.8102	0.8142	0.8291	0.5499	0.9287	**0.9333**	0.9233
Overall	0.9590	0.7942	0.8546	0.8130	0.9758	0.9758	**0.9786**

Table 5. Ablation test for Qwen2.5-VL-3B-Instruct, models on the left are only trained on pictures, those on the right also have: raw time series, moving average and moving standard deviation as text data

References

Ansari, A. F., Stella, L., Turkmen, C., Zhang, X., Mercado, P., Shen, H., Shchur, O., Rangapuram, S. S., Arango, S. P., Kapoor, S., Zschiegner, J., Maddix, D. C., Wang, H., Mahoney, M. W., Torkkola, K., Wilson, A. G., Bohlke-Schneider, M., and Wang, Y. Chronos: Learning the language of time series.

Cai, Y., Choudhry, A., Goswami, M., and Dubrawski, A. Timeseriesexam: A time series understanding exam, 2024. URL https://arxiv.org/abs/2410.14752.

Das, A., Kong, W., Sen, R., and Zhou, Y. A decoder-only foundation model for time-series forecasting, 2024. URL https://arxiv.org/abs/2310.10688.

Daswani, M., Bellaiche, M. M. J., Wilson, M., Ivanov, D., Papkov, M., Schnider, E., Tang, J., Lamerigts, K., Botea, G., Sanchez, M. A., Patel, Y., Prabhakara, S., Shetty, S., and Telang, U. Plots unlock time-series understanding in multimodal models, 2024. URL https://arxiv.org/abs/2410.02637.

DeepSeek-AI, Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., Zhang, X., Yu, X., Wu, Y., Wu, Z. F., Gou, Z., Shao, Z., Li, Z., Gao, Z., Liu, A., Xue, B., Wang, B., Wu, B., Feng, B., Lu, C., Zhao, C., Deng, C., Zhang, C., Ruan, C., Dai, D., Chen, D., Ji, D., Li, E., Lin, F., Dai, F., Luo, F., Hao, G., Chen, G., Li, G., Zhang, H., Bao, H., Xu, H., Wang, H., Ding, H., Xin, H., Gao, H., Qu, H., Li, H., Guo, J., Li, J., Wang, J., Chen, J., Yuan, J., Qiu, J., Li, J., Cai, J. L., Ni, J., Liang, J., Chen, J., Dong, K., Hu, K., Gao, K., Guan, K., Huang, K., Yu, K., Wang, L., Zhang, L., Zhao, L., Wang, L., Zhang, L., Xu, L., Xia, L., Zhang, M., Zhang, M., Tang, M., Li, M., Wang, M., Li, M., Tian, N., Huang, P., Zhang, P., Wang, Q., Chen, Q., Du, Q., Ge, R., Zhang, R., Pan, R., Wang, R., Chen, R. J., Jin, R. L., Chen, R., Lu, S., Zhou, S., Chen, S., Ye, S., Wang, S., Yu, S., Zhou, S., Pan, S., Li, S. S., Zhou, S., Wu, S., Ye, S., Yun, T., Pei, T., Sun, T., Wang, T., Zeng, W., Zhao, W., Liu, W., Liang, W., Gao, W., Yu, W., Zhang, W., Xiao, W. L., An, W., Liu, X., Wang, X., Chen, X., Nie, X., Cheng, X., Liu, X., Xie, X., Liu, X., Yang, X., Li, X., Su, X., Lin, X., Li, X. Q., Jin, X., Shen, X., Chen, X., Sun, X., Wang, X., Song, X., Zhou, X., Wang, X., Shan, X., Li, Y. K., Wang, Y. Q., Wei, Y. X., Zhang, Y., Xu, Y., Li, Y., Zhao, Y., Sun, Y., Wang, Y., Yu, Y., Zhang, Y., Shi, Y., Xiong, Y., He, Y., Piao, Y., Wang, Y., Tan, Y., Ma, Y., Liu, Y., Guo, Y., Ou, Y., Wang, Y., Gong, Y., Zou, Y., He, Y., Xiong, Y., Luo, Y., You, Y., Liu, Y., Zhou, Y., Zhu, Y. X., Xu, Y., Huang, Y., Li, Y., Zheng, Y., Zhu, Y., Ma, Y., Tang, Y., Zha, Y., Yan, Y., Ren, Z. Z., Ren, Z., Sha, Z., Fu, Z., Xu, Z., Xie, Z., Zhang, Z., Hao, Z., Ma, Z., Yan, Z., Wu, Z., Gu, Z., Zhu, Z., Liu, Z., Li, Z., Xie, Z., Song, Z., Pan, Z., Huang, Z., Xu, Z., Zhang, Z., and Zhang, Z. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning, 2025. URL https://arxiv.org/abs/2501.12948.

Eldan, R. and Li, Y. Tinystories: How small can language models be and still speak coherent english?, 2023. URL https://arxiv.org/abs/2305.07759.

Fons, E., Kaur, R., Palande, S., Zeng, Z., Balch, T., Veloso, M., and Vyetrenko, S. Evaluating large language models on time series feature understanding: A comprehensive taxonomy and benchmark, 2024. URL https://arxiv.org/abs/2404.16563.

Gao, S., Koker, T., Queen, O., Hartvigsen, T., Tsiligkaridis, T., and Zitnik, M. Units: A unified multi-task time series model, 2024. URL https://arxiv.org/abs/2403.00131.

Goswami, M., Szafer, K., Choudhry, A., Cai, Y., Li, S., and Dubrawski, A. Moment: A family of open time-series foundation models, 2024. URL https://arxiv.org/abs/2402.03885.

Gruver, N., Finzi, M., Qiu, S., and Wilson, A. G. Large language models are zero-shot time series forecasters, 2024. URL https://arxiv.org/abs/2310.07820.

Gunasekar, S., Zhang, Y., Aneja, J., Mendes, C. C. T., Giorno, A. D., Gopi, S., Javaheripi, M., Kauffmann, P., de Rosa, G., Saarikivi, O., Salim, A., Shah, S., Behl, H. S., Wang, X., Bubeck, S., Eldan, R., Kalai, A. T., Lee, Y. T., and Li, Y. Textbooks are all you need, 2023. URL https://arxiv.org/abs/2306.11644.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network, 2015. URL https://arxiv.org/abs/1503.02531.

Huet, A., Navarro, J. M., and Rossi, D. Local evaluation of time series anomaly detection algorithms. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '22, pp. 635–645. ACM, August 2022. doi: 10.1145/3534678.3539339. URL http://dx.doi.org/10.1145/3534678.3539339.

Jin, M., Zhang, Y., Chen, W., Zhang, K., Liang, Y., Yang, B., Wang, J., Pan, S., and Wen, Q. Position: What can large language models tell us about time series analysis, 2024. URL https://arxiv.org/abs/2402.02713.

Qwen, :, Yang, A., Yang, B., Zhang, B., Hui, B., Zheng, B., Yu, B., Li, C., Liu, D., Huang, F., Wei, H., Lin, H., Yang, J., Tu, J., Zhang, J., Yang, J., Yang, J., Zhou, J., Lin, J., Dang, K., Lu, K., Bao, K., Yang, K., Yu, L., Li, M., Xue, M., Zhang, P., Zhu, Q., Men, R., Lin, R.,

Li, T., Tang, T., Xia, T., Ren, X., Ren, X., Fan, Y., Su, Y., Zhang, Y., Wan, Y., Liu, Y., Cui, Z., Zhang, Z., and Qiu, Z. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Rasul, K., Ashok, A., Williams, A. R., Ghonia, H., Bhagwatkar, R., Khorasani, A., Bayazi, M. J. D., Adamopoulos, G., Riachi, R., Hassen, N., Biloš, M., Garg, S., Schneider, A., Chapados, N., Drouin, A., Zantedeschi, V., Nevmyvaka, Y., and Rish, I. Lag-llama: Towards foundation models for probabilistic time series forecasting, 2024. URL https://arxiv.org/abs/2310.08278.

Sellam, T., Pu, A., Chung, H. W., Gehrmann, S., Tan, Q., Freitag, M., Das, D., and Parikh, A. P. Learning to evaluate translation beyond english: Bleurt submissions to the wmt metrics 2020 shared task, 2020. URL https://arxiv.org/abs/2010.04297.

Spathis, D. and Kawsar, F. The first step is the hardest: Pitfalls of representing and tokenizing temporal data for large language models, 2023. URL https://arxiv.org/abs/2309.06236.

Team, N. Sky-t1: Train your own o1 preview model within $450. https://novasky-ai.github.io/posts/sky-t1, 2025. Accessed: 2025-01-09.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., and Sahoo, D. Unified training of universal time series forecasting transformers, 2024. URL https://arxiv.org/abs/2402.02592.

Xu, X., Li, M., Tao, C., Shen, T., Cheng, R., Li, J., Xu, C., Tao, D., and Zhou, T. A survey on knowledge distillation of large language models, 2024. URL https://arxiv.org/abs/2402.13116.

Yuan, Z., Yuan, H., Tan, C., Wang, W., and Huang, S. How well do large language models perform in arithmetic tasks?, 2023. URL https://arxiv.org/abs/2304.02015.

Zhang, T., Kishore, V., Wu, F., Weinberger, K. Q., and Artzi, Y. Bertscore: Evaluating text generation with bert, 2020. URL https://arxiv.org/abs/1904.09675.

Zhou, Z. and Yu, R. Can llms understand time series anomalies?, 2025. URL https://arxiv.org/abs/2410.05440.