TiMERT-C: A Transformer-Based Foundation Model for Time Series Classification

Josué Abraham López Sánchez
Universidad de Granada/DiCITIS
Superved by: Ph.D. José Manuel Benítez Sánchez &
PhD Candidate Luis Balderas Ruiz

November 28, 2024

Abstract

Time series classification is crucial in multiple domains, from finance to medicine. In parallel, Transformers have proven highly effective in capturing long-range dependencies in natural language processing. In this work, we present **TiMERT-C** (Time-series Modeling with Enhanced Representation Transformer for Classification), a methodology for time series classification using a Transformer-based foundation model trained from scratch. We explore the feasibility of applying a variant of Masked Language Modeling (MLM), specifically a Masked AutoEncoder (MAE), as a pretext task in a time series context. Our experimental results show that TiMERT-C outperforms or matches several state-of-the-art strategies in the UCR Archive. We further conduct a rigorous statistical comparison of our approach against other baselines using the Friedman test and corresponding post-hoc procedures, showing that TiMERT-C statistically outperforms certain methods at different levels of significance.

GitHub Repository: `https://github.com/Josu16/TiMERT`

1 Introduction

Time series classification (TSC) involves assigning discrete labels to time series data. It has a broad range of applications in activity recognition, medicine, finance, climatology, and many other domains [1, 2, 3, 4]. Traditional methods (e.g., Dynamic Time Warping – DTW) and more recent deep learning architectures (such as CNNs and RNNs) have significantly contributed to TSC. However, in recent years, Transformers have excelled in sequence modeling, particularly in natural language processing [5].

In parallel, the *foundation model* paradigm [6] has reshaped the field of machine learning by emphasizing large-scale self-supervised pre-training followed by fine-tuning. When success-

fully adapted to time series, foundation models can capture generic temporal representations that transfer well to different tasks [7].

In this paper, we propose **TiMERT-C**, a two-stage methodology:

- **Pre-training** (self-supervised) with a Masked AutoEncoder (MAE) objective, akin to MLM strategies in NLP.
- **Fine-tuning** (supervised) for classification on diverse subsets of time series.

Our experiments use the UCR Archive [8], a collection of 128 univariate time series datasets from various domains. We compare TiMERT-C with DTW and a contrastive self-supervised baseline, **TimeCLR** [7], under a rigorous statistical test suite to ascertain significance.

2 Related Work

2.1 Time Series Classification (TSC)

Classical approaches for TSC include distance-based methods (e.g., DTW [9]) and Fourier-based transformations [10]. More recent works such as ROCKET [11] and diverse neural architectures have often led to higher accuracy in TSC tasks [12].

2.2 Transformers for Sequential Data

Originally introduced by Vaswani et al. [5] in NLP, Transformers leverage multi-head self-attention to capture dependencies without recurrence. This model has inspired novel applications in time series forecasting [13], classification, and anomaly detection.

2.3 Foundation Models and Self-Supervision

Foundation models [6] rely on large-scale self-supervised learning (SSL), typically covering transformers for language (BERT, GPT) or computer vision (MAE for images [14]). In time series, Yeh et al. [7] explored contrastive SSL (*TimeCLR*) and found Transformers to be effective. Our work adopts an MAE-based masking approach (inspired by MLMs) to pre-train a foundation model purely on time series.

3 TiMERT-C Methodology

3.1 Overall Architecture

TiMERT is an *encoder-only* Transformer, adapted from the structure used in RoBERTa [15]. It includes:

- A **positional encoder** (fixed sinusoidal or learned).
- A **1D convolutional** layer for initial feature extraction.
- Multiple **Transformer encoder blocks** with pre-layer normalization.

Each block has multi-head self-attention, feed-forward layers, and normalization. Outputs are optionally projected.

3.2 Masked AutoEncoder (MAE) for Pre-training

We adopt an MAE variant for time series:

1. **Input Masking**: Randomly mask a fraction (e.g., 15%) of time steps.
2. **Encoding**: Pass the masked sequence into TiMERT, producing a hidden representation.
3. **Decoding/Projection**: Attempt to reconstruct the original time series only on the masked regions, optimizing MSE loss:

$$\mathcal{L} = \sum_{t \in \text{masked}} (x_t - \hat{x}_t)^2 .$$

3.3 Fine-Tuning for Classification

After pre-training, we add a classification head $F(\cdot)$ on top of the encoder:

$$z = \text{TransformerEncoder}(x), \quad \hat{y} = F(z).$$

All weights (including the encoder) are updated via supervised training on labeled datasets.

4 Experimental Setup

4.1 Datasets: UCR Archive

We use UCR Archive [8]. To ensure the model learns generic features, we split the archive:

- **65% of the datasets** for pre-training (no labels used).
- **35% of the datasets** for fine-tuning and testing.

Fourier-based interpolation/decimation [10] standardizes time series lengths to a fixed size (e.g., 512).

4.2 Compared Methods

- **DTW + k-NN**: A widely used baseline with DTW distance [9].
- **TimeCLR** [7]: A contrastive approach for time series foundation models.

4.3 Training Details

- **Pre-training:** 400 epochs with Adam. Learning rate set to 10^{-4}. Mask ratio 15%.
- **Fine-tuning:** 400 epochs with AdamW. Full model unfreezing, cross-entropy loss for classification.
- **Hardware:** Up to 4 NVIDIA GPUs (GTX 1080 Ti), Ubuntu 20.04, Docker-based container.

5 Results and Statistical Comparison

5.1 Performance Overview

TiMERT-C achieves an average classification accuracy of about 79.48% over the chosen UCR subsets, compared to approximately 77.47% for DTW and 77.11% for TimeCLR. Table 1 shows a sample of per-dataset accuracy, whereas Table 2 summarizes the mean outcomes.

5.2 Statistical Tests: Friedman, Iman-Davenport, & Post-Hoc Analysis

To rigorously evaluate the differences among the three methods (*DTW*, *TimeCLR*, *TiMERT-C*), we follow the recommendations of Demšar [16] and García & Herrera [17] for multiple classifiers over multiple datasets.

5.2.1 Friedman Test and Iman-Davenport

First, we compute the Friedman statistic F_F based on average ranks across all datasets:

$$\chi_F^2 = \frac{12N}{k(k+1)} \left(\sum_{j=1}^{k} R_j^2 - \frac{k(k+1)^2}{4} \right),$$

where:

- N is the number of datasets,
- k is the number of classifiers,

- R_j is the average rank of classifier j.

We also use the Iman-Davenport statistic F_F^{ID} which tends to be more conservative for $k > 2$:

$$F_F^{\text{ID}} = \frac{(N-1)\,\chi_F^2}{N(k-1) - \chi_F^2}.$$

In our experiments, the Friedman test yields a statistic around 4.30 (with $p \approx 0.1164$), and the Iman-Davenport test yields $F_F^{\text{ID}} \approx 2.21$ (with $p \approx 0.1160$). *At the 5% significance level, neither test rejects the null hypothesis that all three methods have the same average rank.*

5.2.2 Pairwise Comparisons (Holm/Hochberg) with TiMERT-C as Control

Despite the global test not rejecting H_0 at 5%, we often proceed with post-hoc pairwise comparisons, especially if we suspect certain pairs may differ more than others [16]. Taking *TiMERT-C* as the control method (R_0), we compare:

$$z = \frac{R_0 - R_i}{\sqrt{\frac{k(k+1)}{6N}}},$$

where R_0 is TiMERT-C's average rank and R_i is the rank of the i-th classifier.

TiMERT-C vs. TimeCLR. We obtain $z \approx 2.06$ ($p \approx 0.0398$ unadjusted). Applying Holm/Hochberg correction for $k - 1 = 2$ comparisons, p_{adj} becomes ≈ 0.0797. At $\alpha = 0.05$, this borderline suggests **marginal significance** that TiMERT-C outperforms TimeCLR.

TiMERT-C vs. DTW. We get $z \approx 1.26$ ($p \approx 0.2059$). After correction, p_{adj} remains around 0.2059. Thus, *we cannot conclude that TiMERT-C is significantly better than DTW* at the conventional 5% level. At $\alpha = 0.10$, the difference is still not sufficient to reject the null hypothesis for this particular pair.

5.2.3 Discussion on the Statistical Findings

Overall:

- Global tests (Friedman and Iman-Davenport) do *not* reject the null hypothesis, implying no strong evidence that the three methods differ significantly when all datasets are considered together at $\alpha = 5\%$.
- The post-hoc comparisons indicate a near-significant difference between **TiMERT-C** and **TimeCLR** ($p_{\text{adj}} \approx 0.0797$).
- Comparisons with DTW do not yield significance: $p_{\text{adj}} \approx 0.2059$.

Practically, *TiMERT-C* still exhibits higher mean accuracy than both DTW and TimeCLR across the tested datasets; yet, large inter-dataset variance dilutes the overall significance. These findings show the importance of using robust statistical tools rather than relying solely on average accuracy.

Table 1: Average Ranks of DTW, TimeCLR, and TiMERT-C. Lower rank is better.

Method	Avg. Rank
DTW	2.03
TimeCLR	2.20
TiMERT-C	**1.77**

Table 2: Mean Accuracy (%) Over the Evaluated Subsets.

Model	Mean Acc	Std	N. of Datasets
DTW	77.47	10.2	45
TimeCLR	77.11	12.3	45
TiMERT-C	**79.48**	10.0	45

6 Conclusions and Future Work

We introduced **TiMERT-C**, a Transformer-based foundation model for time series classification that leverages a Masked AutoEncoder (MAE) pretext task. Empirical evidence shows that TiMERT-C yields competitive or superior performance compared to certain baselines, although post-hoc tests reveal that the observed improvement is only marginally significant over TimeCLR under standard significance levels, and not significant against DTW.

Our findings encourage further research, such as:

- Extension to *multivariate* and *irregularly sampled* time series.
- Testing additional self-supervised objectives (e.g., contrastive or adversarial).
- Enhancing model interpretability via attention visualization or attribution methods.

Exploring these directions can help consolidate Transformers as robust foundation models in time series applications.

References

[1] X. Zhang, et al. "Machine learning for biomedical signal analysis," *Applied Sciences*, 2020.

[2] T. Cook, et al. "Time series classification in financial asset management," *Finance Applications*, 2019.

[3] F. Giannetti, et al. "Time-series to image encoding for precipitation forecasting," *Clim. Conf.*, 2024.

[4] F. Smalley, et al. "Time series classification in industry 4.0 scenarios," *Ind. J. Data Sci.*, 2022.

[5] A. Vaswani, et al. "Attention is all you need," *NIPS*, 2017.

[6] R. Bommasani, et al. "On the opportunities and risks of foundation models," *arXiv preprint*, 2022.

[7] C.-C. Yeh, et al. "Toward a foundation model for time series data," *arXiv:2310.03916*, 2023.

[8] E. Keogh. "UCR Time Series Classification Archive," `https://www.cs.ucr.edu/~eamonn/time_series_data_2018/`, 2018.

[9] D. Berndt and J. Clifford. "Using dynamic time warping to find patterns in time series," *KDD Workshop*, 1994.

[10] E. Keogh and M. Pazzani. "An indexing scheme for fast similarity search in large time series databases," *Proc. 11th Int. Conf. on Scientific and Statistical DB*, 1998.

[11] T. Dempster, et al. "ROCKET: Exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining and Knowledge Discovery*, 2020.

[12] G. Middlehurst, et al. "A review of time series classification methods," *ACM Comput. Surv.*, 2024.

[13] Z. Ahmed, et al. "Transformer-based frameworks for time series forecasting," *IEEE Access*, 2023.

[14] K. He, et al. "Masked autoencoders are scalable vision learners," *CVPR*, 2022.

[15] Y. Liu, et al. "RoBERTa: Robustly optimized bert pretraining approach," *arXiv preprint*, 2019.

[16] J. Demšar. "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, 2006.

[17] S. García and F. Herrera. "An extension on statistical comparisons of classifiers over multiple data sets for all pairwise comparisons," *JMLR*, vol. 9, 2008, pp. 2677–2694.