Measuring Pre-training Data Quality without Labels for Time Series Foundation Models

Songkang Wen[1]   **Vasilii Feofanov**[*2]   **Jianfeng Zhang**[1]

[1]Huawei Noah's Ark Lab, Shenzhen, China   [2]Huawei Noah's Ark Lab, Paris, France

Abstract

Recently, there has been a growing interest in time series foundation models that generalize across different downstream tasks. A key to strong foundation models is a diverse pre-training dataset, which is particularly challenging to collect for time series classification. In this work, we explore the performance of a contrastive-learning-based foundation model as a function of the data used for pre-training. We introduce contrastive accuracy, a new measure to evaluate the quality of the representation space learned by the foundation model. Our experiments reveal the positive correlation between the proposed measure and the accuracy of the model on a collection of downstream tasks. This suggests that the contrastive accuracy can serve as a criterion to search for time series datasets that can enhance the pre-training and improve thereby the foundation model's generalization.

1   Introduction

Nowadays, the success of large foundation models (Bommasani et al., 2021) such as GPT-4 (Achiam et al., 2023) or Llama (Touvron et al., 2023) cause dramatic changes in the research and applications. Instead of training a tailored model for a particular task, the foundation model is pre-trained on a collection of datasets with the goal to generalize simultaneously across various downstream tasks either by fine-tuning the model or directly using its output. This workflow effectively simplifies the choice of the model architecture while reducing the requirement for amount of labeled data. The wave of foundation models has now reached the time series domain, including forecasting (Rasul et al., 2023a; Woo et al., 2024) and classification models (Lin et al., 2024).

When it comes to real deployment of a time series foundation model (TSFM), a very important question is whether the pre-training dataset is sufficiently diverse, so the model generalizes well to new downstream tasks. Usually, this is verified by directly evaluating the performance on several supervised downstream tasks thereby requiring availability of annotated data samples and introducing a high cost for assesing the quality of different pre-trained datasets. Therefore, in this paper, we ask the following research question:

Can we evaluate the quality of pre-training data in an unsupervised manner?

By focusing on time series classification and contrastive pre-training, we show that it is possible to leverage important information from the representation space learned by the TSFM. More precisely, if new data points are not similar to pre-training data, their embeddings will tend to not satisfy the uniformity property of contrastive learning (Wang and Isola, 2020), directly impacting the foundation model's performance. Based on this observation, we introduce a new metric called contrastive accuracy that evaluates how well spread the data points are in the embedding space. We empirically show that the proposed metric positively correlates with the model's accuracy on the tasks unseen during pre-training, allowing us to use it for measuring the quality of pre-training examples without the need to regularly test the TSFM on supervised downstream tasks.

---

[*]Correspondence to: vasilii.feofanov@huawei.com.

2 Related Work

Time series representation learning has got a high attention in recent years, resulting in a wealth of excellent works. Many of them use the contrastive learning scheme for pre-training, including TS2Vec (Yue et al., 2022) and TS-TCC (Eldele et al., 2021) with a CNN backbone and TF-C (Zhang et al., 2022) with a ResNet backbone, demonstrating good performance on classification datasets, e.g, the UCR collection (Dau et al., 2019).

With the success of large language models (LLMs), popularity of the transformer architecture for time series analysis increases (Nie et al., 2023; Ilbert et al., 2024), while the development of foundation models is becoming a prevalent task. Numerous TSFMs have been proposed recently, including Lag-Llama (Rasul et al., 2023b), Time-LLM (Jin et al., 2023), One Fits All (Zhou et al., 2023), Brant (Zhang et al., 2023), MOIRAI (Woo et al., 2024). While some of them try to adapt LLM for time series data, others train foundation models from scratch on a large volume of time series data. In our work, we are based on the latter approach, studying how to evaluate the quality of pre-training data, which, to our knowledge, has not been addressed before.

Finally, we would like to notice that our framework reminds unlabeled performance estimation (Donmez et al., 2010) where the goal is to predict model's performance on unlabeled test data. However, most of these approaches focus on single-task classification for computer vision problems (Hendrycks and Gimpel, 2016; Yu et al., 2022; Xie et al., 2024), implying a fundamentally different methodology. To the best of our knowledge, time series foundation models have never been considered in this domain before.

3 Proposed Method

First, in Section 3.1, we introduce the setup we are working in. Then, we present briefly the architecture of our foundation model and how it was pre-trained in Section A, respectively. Finally, in Section 3.3, we propose a new evaluation metric called the contrastive accuracy.

3.1 Problem Setup

We consider the task of unsupervised pre-training where an *unlabeled* pre-training set X_0 is given with m pre-training sequences of length t. The goal is to design a foundation model $F : \mathbb{R}^t \to \mathbb{R}^{d_{\text{hid}}}$ that projects any time series $\mathbf{x} \in \mathbb{R}^t$ to a discriminative hidden space $\mathbb{R}^{d_{\text{hid}}}$. In this work, the quality of the foundation model is evaluated on a collection of downstream tasks $\mathcal{D} = \{D_i\}_{i=1}^{p}$, where D_i consists of observations X_i and labels Y_i. For each downstream task, we perform a train-test split, append F with a linear head, fine-tune it on the training set and compute the accuracy score on the test set. The performance of F with a pre-training dataset X_0 that is averaged over all downstream tasks is denoted by $\mathcal{P}_{\text{test}}(X_0)$. Similarly, $\mathcal{P}_{\text{train}}(X_0)$ denotes the average performance when it is evaluated on the training set.

3.2 Architecture and Pre-training

Similarly to Nie et al. (2023) and Lin et al. (2024), we use the ViT (Dosovitskiy et al., 2021) as the backbone, but our implementation slightly differs from theirs. Instead of dividing a sequence into disjoint patches, we employ a single CNN layer allowing them to be overlapped. Similarly to other TSFMs, we reshape any time series to a fixed sequence length equal to 512. More details on the architecture are given in Appendix A.

For pre-training of the foundation model, we use the contrastive learning that aims to train a such encoder that outputs similar representations for two random augmentations of the same sample (positive pair) and dissimilar representations for augmentations of two different samples (negative pair). More formally, let $\mathcal{T}$ be a considered space of transformations (augmentations) such that $\forall \phi \in \mathcal{T}, \mathbf{x} \in \mathcal{X}$ we have $\phi(\mathbf{x}) \in \mathcal{X}$. In our experiments, we have considered the `RandomCropResize` with a crop length varying from 70% to 80%. To measure similarity of two embeddings, we first project the output of the foundation model $F(\mathbf{x})$ to a lower dimension using a MLP projector

$g : \mathbb{R}^{d_{\text{hid}}} \to \mathbb{R}^{d'_{\text{hid}}}$ and then compute the cosine similarity between the two vectors defined as follows:

$$s_{\cos}(\mathbf{q}, \mathbf{k}) := \frac{\mathbf{q}^{\top} \mathbf{k}}{\|\mathbf{q}\| \cdot \|\mathbf{k}\|}, \qquad \forall (\mathbf{q}, \mathbf{k}) \in \mathbb{R}^{2d'_{\text{hid}}}.$$

Given a batch $B = \{\mathbf{x}_i\}_{i=1}^{b}$, for each example $\mathbf{x}_i$, we sample two augmentation functions ϕ and ψ uniformly from $\mathcal{T}$, i.e., $\phi, \psi \sim \mathcal{U}(\mathcal{T})$, compute the pairwise similarities between all the examples in the following way:

$$\mathbf{s}_i(\phi, \psi) = [s_{\cos}(g \circ F \circ \phi(\mathbf{x}_i),\ g \circ F \circ \psi(\mathbf{x}_j))]_{j=1}^{b} \in \mathbb{R}^b.$$

Following Oord et al. (2018) as well as He et al. (2020) and denoting the cross-entropy error function by $l_{\text{ce}} : \mathbb{R}^b \times \{1, \ldots, b\} \to \mathbb{R}$, we update the weights of F and g by minimizing the contrastive loss defined by $\sum_{i=1}^{b} l_{\text{ce}} \left(\frac{\mathbf{s}_i(\phi, \psi)}{T},\ i \right)$, where $T \in (0, +\infty)$ is a temperature.

3.3 Contrastive Accuracy

It has been shown that a good representation learned by contrastive learning should satisfy uniformity property, i.e., to have a feature distribution that tends to be uniform on the unit hypersphere in order to preserve maximal information (Wang and Isola, 2020). Based on this observation, we introduce the *contrastive accuracy* metric (CA, denoted by A_{con}), that measures how scattered the embeddings of the evaluation examples $\mathrm{X}' = \{\mathbf{x}_i\}_{i=1}^{n}$ are in the representation space. Specifically, we count how many examples have two embeddings (obtained from the two augmentations) to be the nearest neighbors with respect to the other examples in the dataset:

$$A_{\text{con}}^{(\mathrm{X}')}(\mathrm{X}_0) := \mathbb{E}_{\phi \sim \mathcal{U}(\mathcal{T})} \mathbb{E}_{\psi \sim \mathcal{U}(\mathcal{T})} \left[\frac{1}{n} \sum_{i=1}^{n} \mathbb{I} \left(\operatorname*{argmax}_{j = \{1, \ldots, n\}} [\mathbf{s}_i(\phi, \psi)]_j = i \right) \right],$$

where X_0 denotes the pre-training dataset on which the foundation model F was trained. When the dataset size n is large, we split the data into disjoint batches and evaluate $\mathbf{s}_i(\phi, \psi)$ only on those examples that belong to the same batch as i. Further, we will experimentally show that the contrastive accuracy is able to hint what generalization performance on the downstream tasks we can expect from the foundation model pre-trained on X_0.

4 Experiments

In our experiments, we use the UCR (Dau et al., 2019) collection to demonstrate the benefit of the methodology proposed in Section 3. To minimize the effect of randomness, we run each experiment 5 times and report the averaged values. All experiments were performed on a single NVIDIA Tesla V100-32GB GPU card.

4.1 Correlation with Performance

In our first experiment, we show that the contrastive accuracy is able to correlate with the generalization performance of the pre-training model. Given a pre-training data X_0, we randomly draw a subsample $\mathrm{X}_0^{(r\%)}$, which contains $r\%$ examples of X_0, then evaluate $A_{\text{con}}^{(\mathrm{X}_0)}(\mathrm{X}_0^{(r\%)})$ and compare it with the train and the test performance $\mathcal{P}_{\text{train}}(\mathrm{X}_0^{(r\%)})$ and $\mathcal{P}_{\text{test}}(\mathrm{X}_0^{(r\%)})$. As X_0, we have picked one of the four largest datasets in the UCR (ElectricDevices, Crop, FordB, and FordA), and evaluate the performance on the rest 127 datasets in average. Figure 1 illustrates the results for ElectricDevices when varying $r \in [10\%, 100\%]$, and the other results can be found in Appendix B. We can observe that the contrastive accuracy correlates well with the performance, approximating its growth with increasing subsampling ratio. This allows us to perform model selection without directly testing the pre-training model on the downstream tasks. For example, this experiment may help to identify sufficient number of pre-training examples per dataset when the model is trained on a collection of different pre-training datasets.
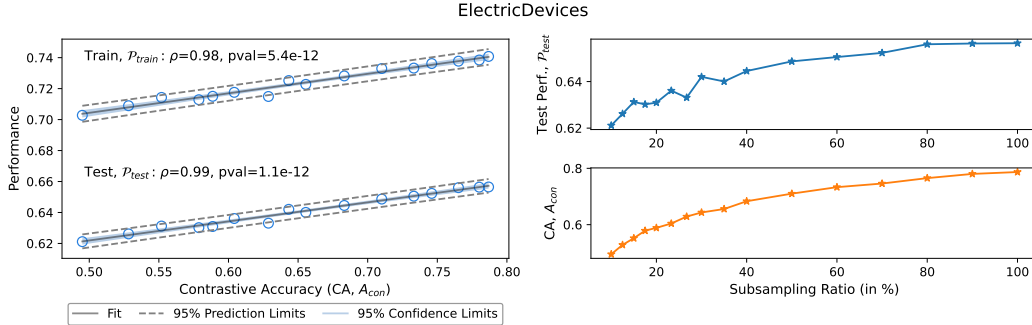
Figure 1: The correlation (ρ) between the contrastive accuracy and the foundation model's performance when varying the ratio of examples used for pre-training.

4.2 Improvement Prediction

In this experiment, in addition to the pre-training dataset X_0, we consider having another one X_0^{new} and ask whether it is possible to predict the performance improvement from including X_0' to the pre-training data, i.e., $\Delta \mathcal{P}(X_0, X_0^{\text{new}}) := \mathcal{P}(X_0 \cup X_0^{\text{new}}) - \mathcal{P}(X_0)$. For this, we propose to consider $\Delta A_{\text{con}}(X_0, X_0^{\text{new}}) := A_{\text{con}}^{(X_0^{\text{new}})}(X_0^{\text{new}}) - A_{\text{con}}^{(X_0^{\text{new}})}(X_0)$ and measure its correlation with $\Delta \mathcal{P}(X_0, X_0^{\text{new}})$. For this experiment, we split the UCR collection into two disjoint sets denoted by $\mathcal{C} = \{X_0^{(i)}\}_{i=1}^{12}$ and $\mathcal{D} = \{X_i\}_{i=13}^{128}$, where the latter is used to evaluate the performance as described in Section 3.1 (more details are given in Appendix B). In the first experiment, we fix the initial pre-training dataset X_0 by taking one from $\mathcal{C}$ and vary X_0^{new} within $\mathcal{C} \setminus \{X_0\}$. In the second experiment, we fix X_0^{new} and vary X_0 in the similar way. For each pair $(X_0, X_0^{\text{new}}$, we compute $\Delta \mathcal{P}(X_0, X_0^{\text{new}})$ and $\Delta A_{\text{con}}(X_0, X_0^{\text{new}})$ and plot the results for the two experiments in Figure 2 for AllGestureWiimoteX dataset and in Appendix B for the rest 11 datasets. In Figure 2, we can observe a positive correlation between the difference in contrastive accuracy and the performance improvement suggesting that the proposed unsupervised criterion can help to search data that can be included to the pre-training dataset in order to improve the representaion and thereby the generalization performance of the foundation model.
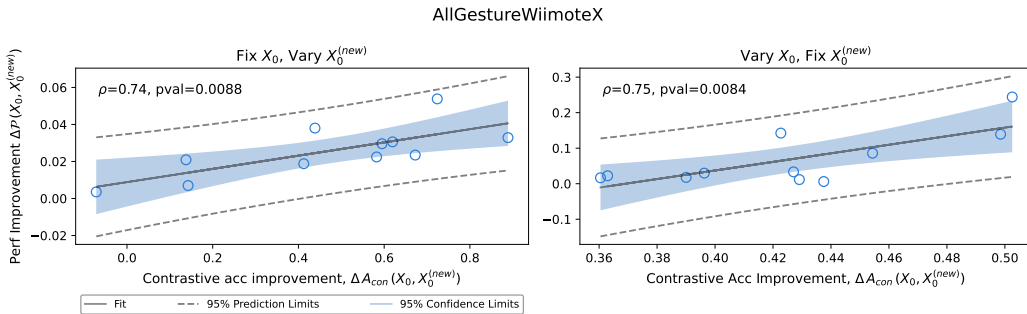

Figure 2: The correlation (ρ) between the improvement in contrastive accuracy and the performance improvement on 116 UCR datasets when expanding the pre-training dataset X_0 by including $X_0^{(\text{new})}$.

5 Conclusion and Future Work

In this paper, we studied the task of evaluating the effect of pre-training data on the foundation model's performance. We proposed the contrastive accuracy and experimentally showed its promise as a criterion to select pre-training data. As a future work, we would like to test our approach with larger pre-training datasets and explore the limits of contrastive pre-training for time series data. Particularly, unlike in computer vision, there is still an open question regarding which augmentation techniques are relevant for contrastive learning in time series data and what their impact is.

4

References

Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Aleman, F. L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., et al. (2023). Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Bommasani, R., Hudson, D. A., Adeli, E., Altman, R., Arora, S., von Arx, S., Bernstein, M. S., Bohg, J., Bosselut, A., Brunskill, E., et al. (2021). On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*.

Dau, H. A., Bagnall, A., Kamgar, K., Yeh, C.-C. M., Zhu, Y., Gharghabi, S., Ratanamahatana, C. A., and Keogh, E. (2019). The ucr time series archive. *IEEE/CAA Journal of Automatica Sinica*, 6(6):1293–1305.

Donmez, P., Lebanon, G., and Balasubramanian, K. (2010). Unsupervised supervised learning i: Estimating classification and regression errors without labels. *Journal of Machine Learning Research*, 11(4).

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., and Houlsby, N. (2021). An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

Eldele, E., Ragab, M., Chen, Z., Wu, M., Kwoh, C. K., Li, X., and Guan, C. (2021). Time-series representation learning via temporal and contextual contrasting. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 2352–2359.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. (2020). Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738.

Hendrycks, D. and Gimpel, K. (2016). A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*.

Ilbert, R., Odonnat, A., Feofanov, V., Virmaux, A., Paolo, G., Palpanas, T., and Redko, I. (2024). SAMformer: Unlocking the potential of transformers in time series forecasting with sharpness-aware minimization and channel-wise attention. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F., editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 20924–20954. PMLR.

Jin, M., Wang, S., Ma, L., Chu, Z., Zhang, J. Y., Shi, X., Chen, P.-Y., Liang, Y., Li, Y.-F., Pan, S., et al. (2023). Time-llm: Time series forecasting by reprogramming large language models. *arXiv preprint arXiv:2310.01728*.

Lin, C., Wen, X., Cao, W., Huang, C., Bian, J., Lin, S., and Wu, Z. (2024). Nutime: Numerically multi-scaled embedding for large- scale time-series pretraining. *Transactions on Machine Learning Research*.

Loshchilov, I. and Hutter, F. (2016). Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*.

Nie, Y., H. Nguyen, N., Sinthong, P., and Kalagnanam, J. (2023). A time series is worth 64 words: Long-term forecasting with transformers. In *International Conference on Learning Representations*.

Oord, A. v. d., Li, Y., and Vinyals, O. (2018). Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Rasul, K., Ashok, A., Williams, A. R., Khorasani, A., Adamopoulos, G., Bhagwatkar, R., Biloš, M., Ghonia, H., Hassen, N. V., Schneider, A., et al. (2023a). Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*.

Rasul, K., Ashok, A., Williams, A. R., Khorasani, A., Adamopoulos, G., Bhagwatkar, R., Biloš, M., Ghonia, H., Hassen, N. V., Schneider, A., et al. (2023b). Lag-llama: Towards foundation models for time series forecasting. *arXiv preprint arXiv:2310.08278*.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., et al. (2023). Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Wang, T. and Isola, P. (2020). Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., and Sahoo, D. (2024). Unified training of universal time series forecasting transformers. In Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., and Berkenkamp, F., editors, *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 53140–53164. PMLR.

Xie, R., Odonnat, A., Feofanov, V., Deng, W., Zhang, J., and An, B. (2024). Mano: Exploiting matrix norm for unsupervised accuracy estimation under distribution shifts. *arXiv preprint arXiv:2405.18979*.

Yu, Y., Yang, Z., Wei, A., Ma, Y., and Steinhardt, J. (2022). Predicting out-of-distribution error with the projection norm. In *International Conference on Machine Learning*, pages 25721–25746. PMLR.

Yue, Z., Wang, Y., Duan, J., Yang, T., Huang, C., Tong, Y., and Xu, B. (2022). Ts2vec: Towards universal representation of time series. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(8):8980–8987.

Zhang, D., Yuan, Z., Yang, Y., Chen, J., Wang, J., and Li, Y. (2023). Brant: Foundation model for intracranial neural signal. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Zhang, X., Zhao, Z., Tsiligkaridis, T., and Zitnik, M. (2022). Self-supervised contrastive pre-training for time series via time-frequency consistency. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 3988–4003. Curran Associates, Inc.

Zhou, T., Niu, P., Wang, X., Sun, L., and Jin, R. (2023). One fits all: Power general time series analysis by pretrained lm. *arXiv preprint arXiv:2302.11939*.

A   Architecture and Implementation Details

In this paper, similarly to Nie et al. (2023) and Lin et al. (2024), we employ the ViT (Vision Transformer) as the backbone. Our goal is to retain information effectively, so we simultaneously use both overlapping and non-overlapping patches. Here is how we achieve it: we apply a one-layer 1D-CNN to handle the overlapping patches and use mean pooling to transform their embeddings to match the number of non-overlapping patches. Next, we embed the μ and σ values from the non-overlapping patches and concatenate the output with the overlapping patches. This result together with the class (CLS) token is then fed into the transformer. The entire network framework is depicted in Figure 3.

Throughout all our experiments, we maintain the same model parameters. The chosen parameter values for different layers are outlined in Table 1. To better train the model, we apply a linear learning rate warm-up in the first 10 epochs (Loshchilov and Hutter, 2016) and a cosine learning rate decay in the subsequent epochs.

B   Experiments

In Figure 4, we display the complete results for the experiment presented in Section 4.1.

Next, in Figure 5 and 6, we give the complete results of the experiment described in Section 4.2. For the pre-training collection $\mathcal{C}$, we chose the following 12 UCR datasets: AllGestureWiimoteX, CricketY, EOGVerticalSignal, Haptics, MelbournePedestrian, PLAID, Phoneme, ScreenType, UWaveGestureLibraryX, WordSynonyms, WormsTwoClass, Yoga.
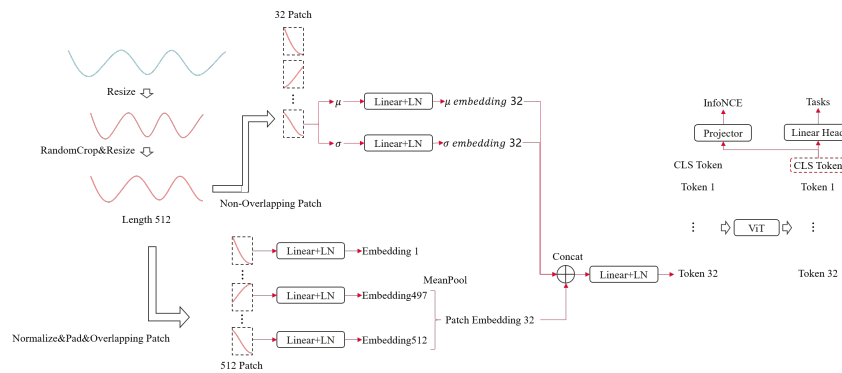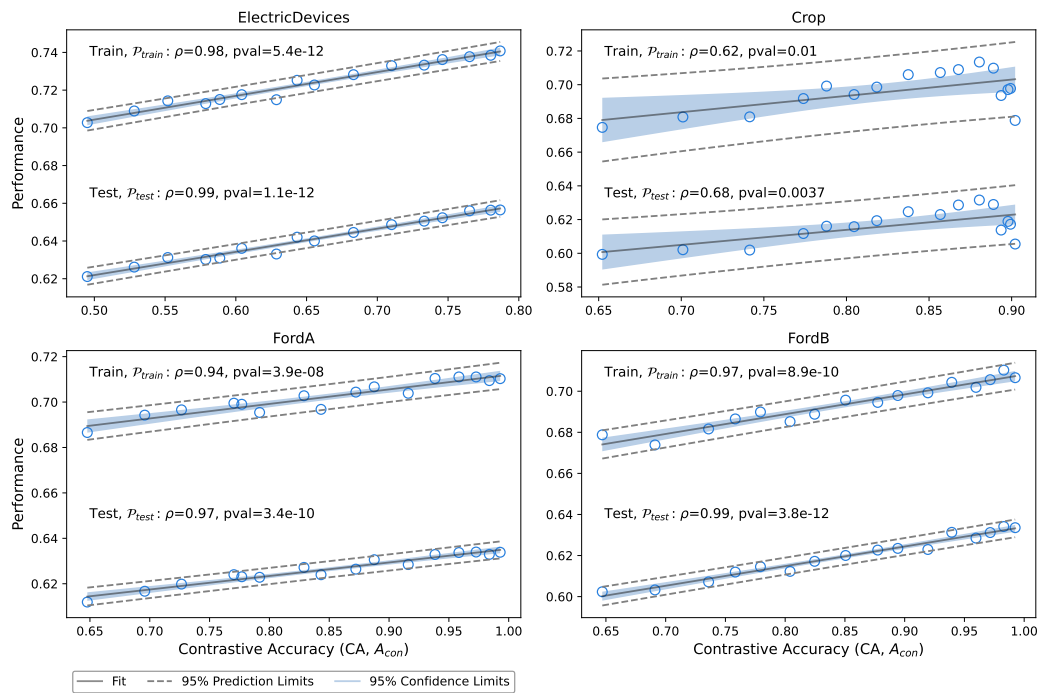
Figure 3: The framework of the model.

Figure 4: The correlation (ρ) between the contrastive accuracy and the foundation model's performance when varying the ratio of examples used for pre-training.

Table 1: The parameters and setting for the model and experiments in this paper.

| Part	Parameters	Value
RandomCropResize	Crop Scale	0.7 0.8
	Resize Length	512
Non-Overlapping Patch	Patch Length	16
	CNN Kernel Size	17
Overlapping Patch	CNN Padding	8
	Output Channel	256
	Tokens Number	32
	CLS Tokens	1
	Token dimension	256
ViT	Layer	6
	Number Of Head	8
	Dimension of Head	128
	MLP dimension	512
Linear for mu	layer	1
	dim	1x32
Linear for std	layer	1
	dim	1x32
InfoNCE Loss	Temperature	0.1
	LayerNorm	
	Layer	2
Non-Linear Projector	Input Dim	256
	Hidden Dim	512
	Output Dim	256
	Activation Function	ReLU
	LayerNorm	
Classifier Linear Head	layer	1
	Dim	256xClassNum
	Learning Rate	2e-04
Training Setting	Epochs	500
	Batch Size	64
	Learning Rate	2e-04
Testing Setting	Epochs	500
	Batch Size	256
	Type	AdamW
Optimizer	Betas	0.9 0.999
	Weight Decay	0.05
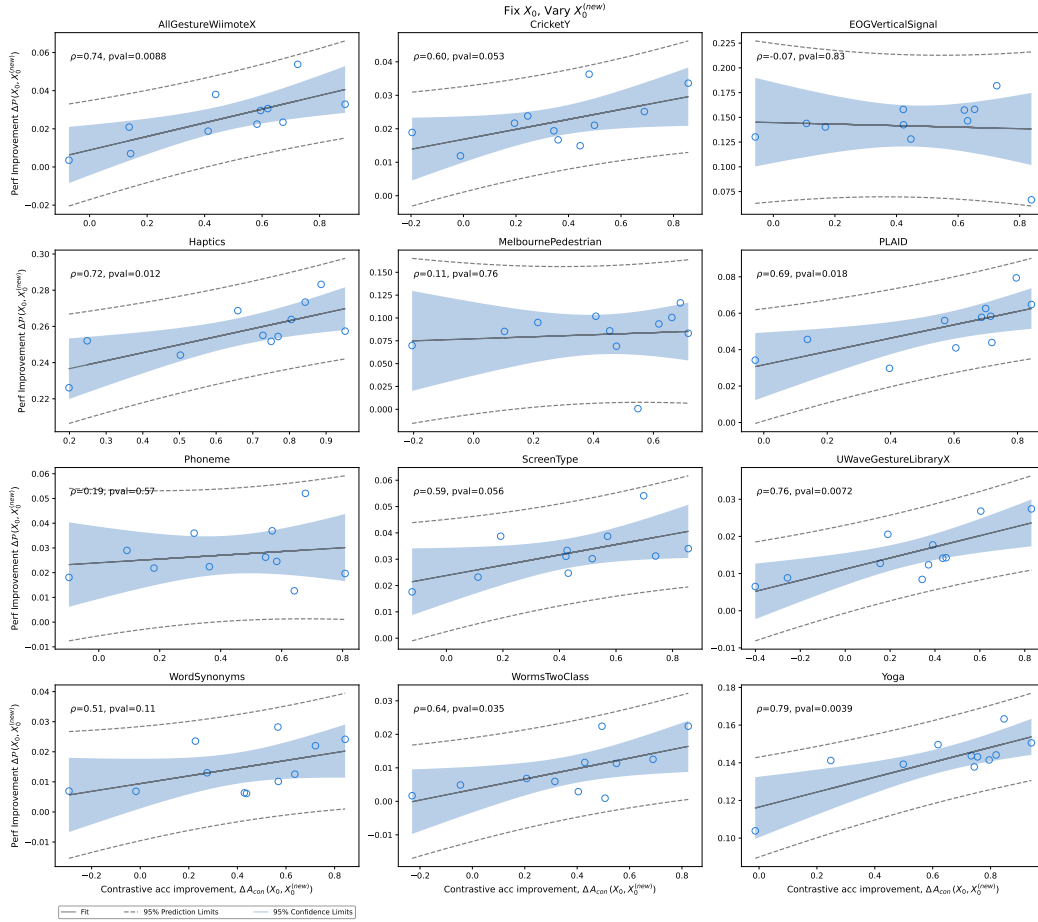
Figure 5: The correlation (ρ) between the improvement in contrastive accuracy and the performance improvement on 116 UCR datasets when expanding the pre-training dataset X_0 by including $X_0^{(\text{new})}$. To measure correlation, we fix X_0 and vary $X_0^{(\text{new})}$ across 11 datasets.
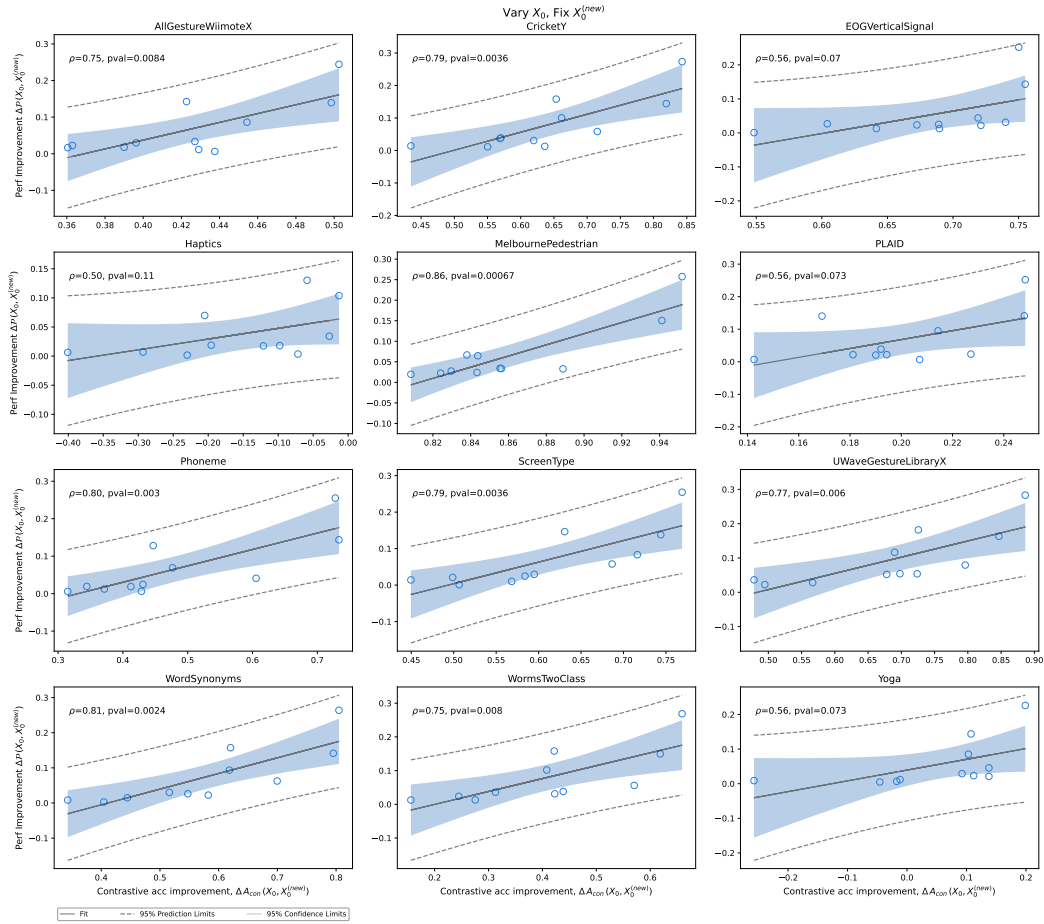
Figure 6: The correlation (ρ) between the improvement in contrastive accuracy and the performance improvement on 116 UCR datasets when expanding the pre-training dataset X_0 by including $X_0^{(\text{new})}$. To measure correlation, we fix $X_0^{(\text{new})}$ and vary X_0 across 11 datasets.