

# Model Selection with a Shapelet-based Distance Measure for Multi-source Transfer Learning in Time Series Classification<sup>\*</sup>

Jiseok Lee and Brian Kenji Iwana<sup>[0000–0002–5146–6818]</sup>

Graduate School of Information Science and Electrical Engineering,  
Kyushu University, Fukuoka, Japan  
`jiseok.lee@human.ait.kyushu-u.ac.jp`  
`iwana@ait.kyushu-u.ac.jp`

**Abstract.** Transfer learning is a common practice that alleviates the need for extensive data to train neural networks. It is performed by pre-training a model using a source dataset and fine-tuning it for a target task. However, not every source dataset is appropriate for each target dataset, especially for time series. In this paper, we propose a novel method of selecting and using multiple datasets for transfer learning for time series classification. Specifically, our method combines multiple datasets as one source dataset for pre-training neural networks. Furthermore, for selecting multiple sources, our method measures the transferability of datasets based on shapelet discovery for effective source selection. While traditional transferability measures require considerable time for pre-training all the possible sources for source selection of each possible architecture, our method can be repeatedly used for every possible architecture with a single simple computation. Using the proposed method, we demonstrate that it is possible to increase the performance of temporal convolutional neural networks (CNN) on time series datasets.

**Keywords:** Transfer Learning · Time Series Classification · Transferability Estimation.

## 1 Introduction

Neural networks have widespread usage in time series recognition. For example, temporal Convolutional Neural Networks (CNN) [19] have been shown to be effective across many time series domains [38, 2]. However, often, neural networks require large amounts of data [10, 15]. Also, acquiring large amounts of annotated data can take time and effort.

Several ideas exist to solve the problem of the requirement of large amounts of annotated data, such as transfer learning, self-supervised learning, data augmentation, etc. In particular, transfer learning has become a popular method of initializing neural networks. In transfer learning, to alleviate the need for data, neural networks can be trained on larger *source* datasets and fine-tuned for *target* datasets. In this way, the weights of the neural network can be trained to

---

<sup>\*</sup> This work was partially supported by MEXT-Japan (Grant No. 23K16949).

extract generalized features [46] and be used for the target task. In the image recognition domain, transfer learning is a standard practice. For example, using pre-trained models trained with ImageNet [8] in image recognition is standard practice. However, for time series, transfer learning is still a budding field [10].

In order to realize an effective method of transfer learning for time series, we propose a combination of multi-source transfer learning with a novel shapelet-based distance measure used for dataset selection. Specifically, to increase the effectiveness of transfer learning and the source dataset’s size, we propose a method of using multiple datasets for pre-training.

However, selecting the datasets for transfer learning is complex. Fawaz et al. [10] demonstrated that the choice of dataset for transfer learning for time series has a large effect on the effectiveness of transfer learning. Notably, only some datasets increased the accuracy of the model. Oftentimes, the accuracy was decreased when using an inappropriate dataset.

Thus, we propose a dataset distance-based measure to select the appropriate datasets for our multi-source pre-training. To do this, first, we extract discriminative shapelets using shapelet discovery [42]. Next, a dataset distance measure is created by comparing the discriminative shapelets between the source and target datasets. The idea is that datasets with similar discriminative shapelets would have similar features, thus leading to more effective transfer learning.

The contribution of this paper is as follows:

- We propose a new method of predicting and selecting source datasets for transfer learning in temporal neural networks. This method uses extracted shapelet similarity between the target and possible source datasets.
- We create a transfer learning method that combines multiple sources into one super dataset.
- We evaluate the proposed method on all 128 time series datasets from the 2018 UCR Time Series Archive (UCR Archive) [7].
- We provide code for easy transfer learning at <https://github.com/uchidalab/time-series-transferability>

## 2 Related Works

### 2.1 Transfer Learning for Time Series

Transfer learning has been used for various applications in image recognition [46]. Furthermore, transfer learning has become the standard practice for training networks, as pre-trained weights are available for the most popular image recognition network architectures.

Conversely, transfer learning is less common for time series recognition and temporal neural networks [10] outside of Natural Language Processing (NLP). However, there have been a few works that demonstrate the usefulness of transfer learning in the time series domain [10, 39, 36]. Other examples include using transfer learning with health data [4], human activity recognition [1], prediction

of internet load [9], and fall detection [23]. De Souza et al. [33] proposed decomposing time series into shapelets and noise and using the decompositions to pre-train models. In comparison to our method, we use shapelets as a distance measure between datasets and not directly to train models.

For the source selection, several works demonstrated that source selection with dataset similarity, computed by using Dynamic Time Warping (DTW) [27] distance, can be effective [10, 43, 22].

## 2.2 Multi-Source Transfer Learning

There have been a few works that use multiple source datasets for transfer learning. For example, Yao and Doretto [41] extend TrAdaBoost [6], a method of boosting transfer learning, to use with multiple sources. Huang et al. [13] improve on this and propose SharedBoost for multiple source transfer learning. Multi-transfer [34] combines multi-view and multi-source transfer learning. For multi-source transfer learning, Song et al. [32] use the conditional probability difference to weight source domains.

Multi-source transfer learning has also been used for time series data. One of the typical methods of multi-source transfer learning is to use a preliminary classifier to select the sources. For example, for electroencephalogram (EEG) data, Jinpeng Li et al. [20] trained each source individually, and then they tested the target domain on each and selected the top-performing models. Ren et al. [26] classify EEG data using a multi-source model. Huiming Lu et al. [22] use an ensemble model to implement multi-source transfer learning for building energy prediction.

Also, Yao et al. [40] use multi-source transfer learning with Variational Mode Decomposition to improve PM2.5 concentration forecasting while selecting sources using Euclidean Distance and Maximum Mean Discrepancy. Lotte and Guan [21] use a search algorithm to combine different datasets. Senanayaka et al. [29] used a similarity-based approach for multi-source transfer learning to generate a mixed domain of multiple sources and targets in the pre-training stage.

## 3 Transferability Measure

### 3.1 Problem Setting

Given source dataset  $\mathcal{S} = \{(\mathbf{s}_1, z_1), \dots, (\mathbf{s}_m, z_m), \dots, (\mathbf{s}_M, z_M)\}$ , where  $(\mathbf{s}_m, z_m)$  is the  $m$ -th pair of pattern  $\mathbf{s}_m$  and respective label  $z_m$ , transfer learning aims to train a network  $f$  with  $\mathcal{S}$ , so that it will be a practical starting point for target dataset  $\mathcal{T} = \{(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_n, y_n), \dots, (\mathbf{x}_N, y_N)\}$ , where  $(\mathbf{x}_n, y_n)$  is the  $n$ -th pair of pattern  $\mathbf{x}_n$  and respective label  $y_n$ . Unlike domain adaptation, there is no assumption that the task of the source and target datasets are related.

As Fawaz et al. [10] found, not all source datasets are useful for transfer learning with time series. Thus, a suitable source dataset  $\mathcal{S}$  for each target dataset  $\mathcal{T}$  should be determined. Under the problem setting, this determination should

be performed before training  $\mathcal{T}$ , i.e., without exhaustively fine-tuning  $\mathcal{T}$  on all possible datasets.

Two types of measures have been proposed to solve the source selection problem. One class of measures is to estimate the transferability of the pre-trained network  $f$ , and the other is to measure the similarity of the datasets.

### 3.2 Transferability Estimation

Transferability estimation measures attempt to predict how effective transfer learning will be for model  $f$ , pre-trained on source dataset  $\mathcal{S}$ , for target dataset  $\mathcal{T}$ . These methods first use models  $f$  pre-trained on datasets  $\mathcal{S}$  to predict target dataset  $\mathcal{T}$ . Specifically, prediction  $f(x_n)$  is done using the data  $x_n$  of  $\mathcal{T}$  with the source labels  $c \in C$  and the features  $\mathcal{F}(x_n) \in \mathcal{F}$  are extracted from model  $f$  trained by  $\mathcal{S}$ . While the source labels  $C$  might be unrelated to the target task, the outputs are used for the transferability estimation. In other words, models trained for  $\mathcal{S}$  are used as-is with dataset  $\mathcal{T}$ , and the amount of information inferred by the model is measured and used as a transferability estimation measure. Some transferability estimation measures include, Log Expected Empirical Prediction (LEEP) [24], Negative Conditional Entropy (NCE) [37], Log Maximum Evidence (LogME) [45], Transrate [12], and H-score [3].

For example, LEEP [24] first predicts the target dataset  $\mathcal{T}$  using trained  $f$ . LEEP is then calculated by:

$$\text{LEEP}(\mathcal{S}, \mathcal{T}) = \frac{1}{N} \sum_{n=1}^N \log \left( \sum_{c \in C} \hat{P}(y_n|c) f(x_n) \right), \quad (1)$$

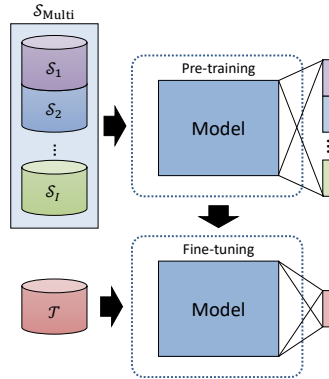
where  $\hat{P}(y_n|c)$  is the empirical conditional distribution calculated by:

$$\hat{P}(y_n|c) = \frac{\hat{P}(y_n, c)}{\hat{P}(c)} = \frac{\frac{1}{N} \sum_{n: y_n=c} f(x_n)}{\frac{1}{N} \sum_{n=1}^N f(x_n)}. \quad (2)$$

LEEP is the average log-likelihood of the prediction of  $\mathcal{T}$  in trained network  $f$  multiplied by  $\hat{P}(y_n|c)$  for each source class  $c$ .

**Dataset Similarity Measure for Source Selection** As an alternative to transferability estimation, dataset similarity can also be used to predict transferability. The previous methods are helpful because only the pre-trained network  $f$  and not the original dataset  $\mathcal{S}$  is needed to calculate transferability. However, unlike image recognition, standard models with downloadable weights for time series recognition are lacking. Therefore, requiring pre-trained networks is a detriment because it requires training many networks before pre-training the actual network for the task.

Conversely, measuring the distance between datasets only requires access to the datasets. Following this, Fawaz et al. [10] proposed to use DTW [27] between representative time series patterns from each class in the target and



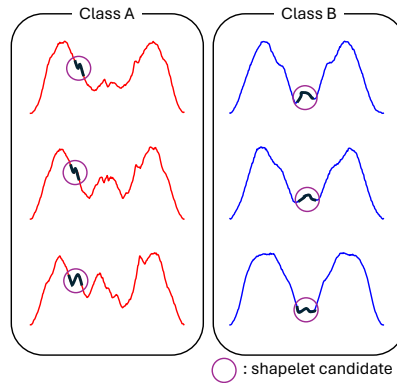
**Fig. 1.** An illustration of our multi-source transfer learning. Source datasets  $\mathcal{S}_i$  are selected using a transferability measure, and the neural network is pre-trained. The trained weights are then fine-tuned using target dataset  $\mathcal{T}$ .

source datasets. The representative time series is the average time series of each class found by DTW Barycenter Averaging (DBA) [25]. They defined the distance between datasets as the distance between the most similar average time series from each dataset. In this paper, we define this method as DBA-DTW. By using the dataset-based distance measure, the appropriate source dataset can be selected for the target dataset. The benefit of this and the proposed method is that no initial trained model is required to measure transferability.

## 4 Multi-Source Transfer Learning

We propose a simple yet effective method of combining multiple source datasets for transfer learning. As shown in Fig. 1, to perform multi-source pre-training, we compile multiple source datasets  $\mathcal{S}_1, \dots, \mathcal{S}_i, \dots, \mathcal{S}_I$  into one super dataset  $\mathcal{S}_{\text{Multi}} = \{\mathcal{S}_1, \dots, \mathcal{S}_i, \dots, \mathcal{S}_I\}$ . In order to transfer knowledge from a model trained with multiple sources, the datasets are pre-processed so that they have the same number of time steps as the target dataset. Note that by resampling the source datasets, the features and characteristics of the datasets might not be preserved. However, this fact is optional because the purpose of the pre-training is to acquire a robust set of initial weights for transfer learning and not the classification accuracy of the source datasets. In addition to resampling,  $\mathcal{S}_{\text{Multi}}$  is balanced so that each sub-dataset has the same number of time series. To balance  $\mathcal{S}_{\text{Multi}}$ , oversampling is performed while preserving the class ratios. This is done due to the discrepancy in the size of possible source datasets; it ensures that every dataset has an equal contribution to the transfer learning.

As shown in Fig. 1, in order to train a network with multiple datasets, the one-hot vector of the ground truth of each source dataset  $\mathcal{S}_i$  are concatenated, and the output nodes are extended accordingly. In this way, the network is then trained using  $\mathcal{S}_{\text{Multi}}$  for a classification task using all of the classes from all of



**Fig. 2.** Examples of shapelets from the Arrowhead dataset. The left and right figures are three time series patterns from the same classes.

the source datasets. The result is the ability to pre-train a network with a larger dataset with a larger number of classes.

After training the network, fine-tuning can be performed as typical transfer learning does. The weights of the trained network can be used as an initialization for a target dataset and fine-tuned for a specific task. While the experimental results use temporal CNNs, there is no theoretical limitation on the type of neural network used.

## 5 Shapelet Similarity-based Source Selection

In order to use the proposed transfer learning method effectively, some source datasets need to be selected. However, as mentioned, selecting the source datasets needs to be performed. Thus, we propose a new method of measuring the transferability of networks through a novel dataset similarity measure using discriminative shapelets.

### 5.1 Shapelet

A shapelet refers to a subsequence extracted from time series data that are maximally representative of a class [42]. These subsequences are intended to encapsulate fundamental patterns or discriminative features within a class. For example, the circled subsequences in Fig. 2 are well discovered within a class and represent differences between the two classes. In the figure, the circled shapelets are segments of the time series unique to each class.

### 5.2 Matrix Profile for Shapelet Discovery

Since a shapelet can be any subsequence from a time series, finding the maximum representative shapelet would be too costly with brute force. In order to overcome

this issue, Matrix Profile [44] has been shown to find discriminative shapelet candidates efficiently.

Matrix Profile is an algorithm that represents a time series based on the distances between subsequences of that time series and their nearest neighbor. Specifically, given an ordered list of all subsequences  $\mathcal{A}$  of a single time series  $\mathbf{t}$ , Matrix Profile  $\mathbf{p}$  is a sequence that holds the distances between each subsequence  $\mathcal{A}_r$  to its nearest neighbor, or:

$$\mathbf{p} = \|\mathcal{A}_1 - \mathcal{E}_1\|, \dots, \|\mathcal{A}_r - \mathcal{E}_r\|, \dots, \|\mathcal{A}_R - \mathcal{E}_R\|, \quad (3)$$

where  $\mathcal{E}_r$  is the nearest subsequence of  $\mathcal{A}$  to the respective  $\mathcal{A}_r$  and  $\|\cdot\|$  is the sum of the pair-wise Euclidean distances between each element in the subsequences. By finding Matrix Profile  $\mathbf{p}$  using time series  $\mathbf{t}$ , Matrix Profile can be used as a fast motif and discord discovery method.

In order to use Matrix Profile for shapelet discovery, a few modifications are performed. First, given a dataset  $\mathcal{S}$ , all of the time series of each class  $c$  are concatenated into a single time series  $\mathbf{t}^{(c)}$ . For example, given class 1 and class 2, a time series  $\mathbf{t}^{(1)}$  and  $\mathbf{t}^{(2)}$  are created. Then, instead of just calculating Matrix Profile using the nearest neighbors of  $\mathcal{A}^{(1)}$  with itself in (3), a Matrix Profile calculation is made for each combination of the two classes, or  $\mathbf{p}^{(1,1)}$ ,  $\mathbf{p}^{(1,2)}$ ,  $\mathbf{p}^{(2,2)}$ , and  $\mathbf{p}^{(2,1)}$ . Finding the highest values in the differences  $\mathbf{p}^{(1,2)} - \mathbf{p}^{(1,1)}$  and  $\mathbf{p}^{(2,1)} - \mathbf{p}^{(2,2)}$  will identify the maximally representative shapelet candidates for class 1 and 2, respectively. Because this is only compatible with two-class classification, we extend shapelet discovery via Matrix Profile using a one-versus-all approach for each class.

### 5.3 Shapelet Similarity-based Source Selection

Now that the representative shapelets  $\mathcal{P}^{(\mathcal{S})}$  and  $\mathcal{P}^{(\mathcal{T})}$  can be found for each source dataset  $\mathcal{S}$  and target dataset  $\mathcal{T}$ , respectively, we use them as a basis for a dataset distance measure. We propose two shapelet distance measure schemes, Average Shapelet, and Minimum Shapelet distances. Fig. 3 represents an overview of Average Shapelet and Minimum Shapelet.

**Average Shapelet** takes the average distance for each combination of  $\mathcal{P}_i^{(\mathcal{S})}$  and  $\mathcal{P}_j^{(\mathcal{T})}$ , or:

$$D_{\text{as}} = \frac{1}{IJ} \sum_{i,j} \|\mathcal{P}_i^{(\mathcal{S})} - \mathcal{P}_j^{(\mathcal{T})}\|, \quad (4)$$

where  $\mathcal{P}_i^{(\mathcal{S})}$  and  $\mathcal{P}_j^{(\mathcal{T})}$  are the  $i$ -th and  $j$ -th shapelet in  $\mathcal{P}^{(\mathcal{S})}$  and  $\mathcal{P}^{(\mathcal{T})}$ , respectively. By using the average distance between shapelets, this measure compares all of the discriminative features of the datasets simultaneously. The general idea of this measure is that if all of the shapelets of the datasets are similar, then the datasets might be similar.

**Minimum Shapelet** is defined as the distance between the most similar pair of shapelets of  $\mathcal{S}$  and  $\mathcal{T}$ , or:

$$D_{\text{ms}} = \min_{i,j} \|\mathcal{P}_i^{(\mathcal{S})} - \mathcal{P}_j^{(\mathcal{T})}\|. \quad (5)$$

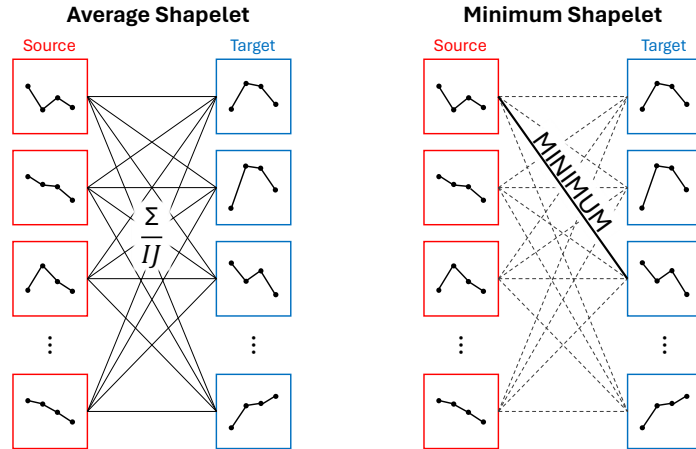


Fig. 3. Overview of Average Shapelet and Minimum Shapelet.

Instead of measuring all of the features, this measure allows the distance calculation to ignore features that might be specific to a dataset.

## 6 Experimental Result

### 6.1 Dataset

The experiments were conducted using all of the UCR Archive [7], which consists of 128 datasets. We use the predetermined training and test set split provided by the archive. Also, no pre-processing was performed except for resizing datasets through Gaussian smoothing with different lengths.

### 6.2 Settings and Architecture

For the experiment, we adopted a 1-dimensional CNN model based on the VGG architecture [31]. The convolutional network used three blocks of convolutional layers and a pooling layer. The first block has two convolutional layers of size 3, and the subsequent blocks have three convolutional layers. Max pooling is used with the first two blocks, and global average pooling (GAP) is used with the final block. While GAP is not required for the proposed method, it is required for LEAP, NCE, H-score, Transrate, and LogMe due to having different-sized datasets; thus, we used it for all evaluations.

For training, we pre-train the network for 10,000 iterations with Adaptive Moment Estimation (Adam) optimizer [17] with an initial learning rate of 0.0001. For transfer learning, we then fine-tune the network for 5,000 iterations. The batch size is set to 32 for both pre-training and fine-tuning. For statistical validity, we fine-tuned the model three times in order to have the mean of

the three models’ performances. Since the traditional transferability measures, LEEP, NCE, LogME, Transrate, and H-Score, require a trained network, an initial network to calculate the transferability is trained for 5,000 iterations.

There are two hyperparameters associated with the shapelet discovery by Matrix Profile. First, we use a fixed shapelet size of 15 because this is the largest possible size on the UCR Archive’s smallest dataset. Next, we use the top 10 shapelet candidates per class.

### 6.3 Comparative Evaluation

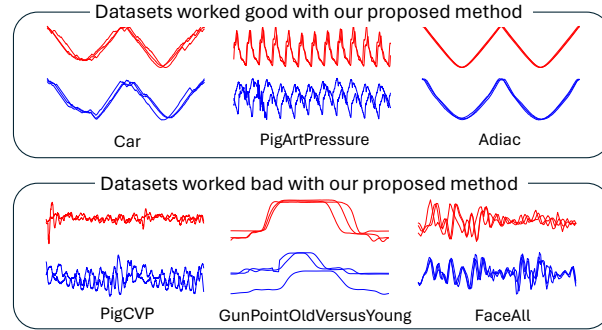
To evaluate the proposed method, we compared it to not using transfer learning, to using transfer learning using a dataset selected by a shapelet-based distance measure, and to using a dataset selected by the other transferability metrics. The comparative measures used for source selection include using DTW between DBA class representatives (DBA-DTW) [10], LEEP [24], NCE [37], Transrate [12], LogME [45], and H-score [3].

**Table 1.** Average test accuracy.

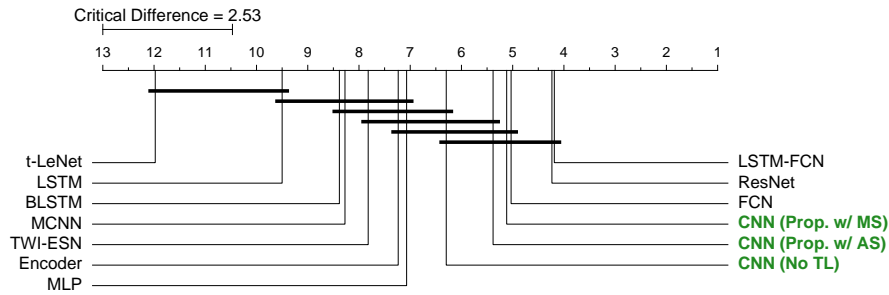
Method	Accuracy (%)
No Transfer Learning (TL)	74.18
TL w/ Random Source	76.89
TL w/ DBA-DTW	78.85
TL w/ H-Score	77.24
TL w/ LEEP	76.43
TL w/ LogME	79.59
TL w/ NCE	78.92
TL w/ Transrate	77.22
Proposed w/ Average Shapelet (10 shapelets)	79.91
Proposed w/ Minimum Shapelet (10 shapelets)	<b>80.25</b>

The results of the experiments are shown in Table 1. In the table, TL represents typical transfer learning that selects a single source dataset, and MTL represents our proposed method, Multi-source Transfer Learning. Compared to the model with random initialized weights, Multi-source Transfer Learning with a Minimum Shapelet of 10 candidates showed the highest improvement. On the other hand, the Average Shapelet showed lower improvement than the Minimum Shapelet; however, it is still better than other comparative methods.

Additionally, we looked into the datasets that showed better and worse performance by adopting our proposed method. Fig. 4 shows sample plots of the datasets. From the sample plot, we can affirm that our proposed method works better with datasets with more clear features. Also, the datasets with more noise showed worse performance than our proposed method. This trend is due to our proposed method focusing on shapelet similarity.

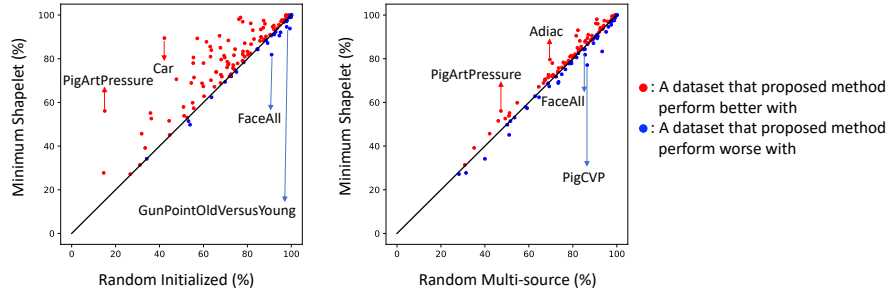


**Fig. 4.** Sample plot of datasets of UCR Archive. Random three samples are plotted from two classes. The upper three datasets performed better with our proposed method, and the lower three datasets showed adverse effects by adopting our proposed method.



**Fig. 5.** A Nemenyi post-hoc test diagram. The proposed methods are in green. The numbers indicate the average rank when tested on all 128 datasets.

To compare the proposed method to other methods found in the literature, we compare it against reported results on the UCR Archive from methods that use neural networks. Fig. 5 is a Nemenyi post-hoc test diagram comparing the proposed methods, the base models, and various reported evaluations. The comparison models include a temporal Residual Network (ResNet) [38], Fully Convolutional Network (FCN) [38], MLP [38], Multi-scale CNN (MCNN) [5], Time Warping Invariant Echo State Network (TWI-ESN) [35], Time Le-Net (t-LeNet) [18], universal Encoder [30], LSTM [11], BLSTM [28], LSTM-FCN [16]. The models were evaluated by Wang et al. [38], Ismail Fawaz et al. [14], and Iwana and Uchida [15]. The figure shows that the networks used by the proposed method are comparable to the other state-of-the-art neural networks on the same datasets.



**Fig. 6.** Ablation study of our proposed method. The Y-axis represents the classification accuracy of our proposed method with 14 sources selected by Minimum Shapalet. The X-axis of the left and right figures represent the classification accuracy of random initialized models and proposed multi-source transfer learning with 14 random sources.

## 7 Discussion

### 7.1 Ablation Study

We compared the classification performances with multi-source pre-training with shapelet-based source selection, multi-source pre-training with random source selection, and without transfer learning. Fig. 6 compares the performances of each dataset of the UCR Archive. Our proposed method showed significantly better results than the random initialized model and multi-source transfer learning with random source selection. Specifically, the result on the right figure demonstrates that our proposed shapelet similarity-based source selection is effective for source selection on multi-source transfer learning. Also, according to the t-test, our proposed method is effective with most datasets,  $p < 0.001$ .

### 7.2 Number of Datasets

In order to examine how the hyperparameters for the proposed method affect performance, we examined the number of shapelet candidates, the number of source datasets, and the two shapelet similarity-based distance measures. For the number of shapelet candidates, we discovered three, five, and ten shapelet candidates for every dataset. For the number of source datasets, we selected 1, 2, 4, 6, 8, 10, 12, 14, 16, 18, and 20 datasets according to shapelet-based distance measures. Finally, we examined the Average Shapelet and the Minimum Shapelet for the shapelet-based distance measure.

The experimental results in Fig. 7 showed better performance with more sources for all selection methods, including Random multi-source selection. Thus, increasing the number of datasets using our multi-source transfer learning method effectively increases the effect of pre-training while alleviating the risk of negative transfer. However, there was no significant difference when the

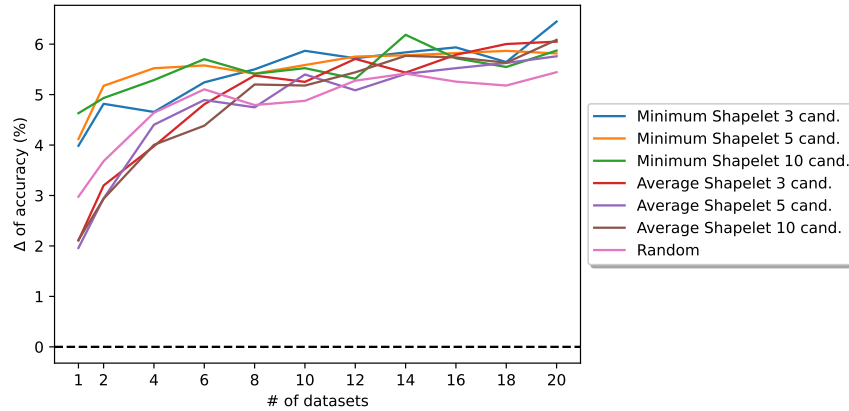


Fig. 7. Experimental result of examining hyperparameters.

number of datasets was more than 10. This implies a diminishing returns effect with the number of datasets. Thus, there is a limit to how many datasets should be combined. However, one possible reason for the diminishing returns is due to our training scheme. We set the number of pre-training iterations to a fixed number, no matter how many datasets and data samples were added for a fair comparison. Therefore, it might be possible to increase the accuracy further using multiple datasets with more iterations.

### 7.3 Similarity Based Source Selection

Generally, with a small-scale target dataset, pre-training with a closely related source dataset allows for more efficient training while reducing the risk of overfitting. However, many time series tasks such as EGG, Speeches, and Gestures have different features and usually contain a small number of patterns. Thus, selecting a source dataset for time series transfer learning is often a big challenge, and transfer learning with non-related tasks is usually not helpful.

According to the experimental result, as depicted in Table 1, selecting a source dataset based on dataset ranking metrics of DBA-DTW and Minimum Shapelet resulted in superior performance compared to a random source selection and even to the classic transferability measures. Therefore, for time series classification, where lack of data is a frequent challenge, measuring time series similarity can serve as a valuable indicator of transferability.

### 7.4 Computational Time

Regarding computational time, shapelet similarity-based source selection has a huge benefit compared to other transferability estimation metrics. Shapelet similarity-based dataset ranking has two steps of calculation: one step is to generate shapelet, and the other is to calculate the similarity among generated shapelets. Thus, the shapelet similarity-based source selection requires

$O(n^2 + w)$ ;  $O(n^2)$  to generate shapelet and  $O(w_s)$  to calculate the similarity among shapelets, where  $n$  is the length of the dataset and  $w_s$  is the length of shapelet. The DBA-DTW also has the benefits of computational time compared to the other transferability estimation measures; however, it is not like the shapelet similarity-based method. DBA-DTW requires  $O(i \cdot n^2 + w_d^2)$ ;  $O(i \cdot n^2)$  to generate prototypes through DBA and  $O(w_d^2)$  to calculate distance among generated shapelets, where  $w_d$  is the length of a prototype of DBA.

The other transferability estimation measures, such as LEEP, require a pre-trained model to measure transferability. Training a model requires significant time, and they have a disadvantage in that they need to re-calculate the transferability when the model architecture is changed. In this research, it is required to train  $128 \times 128$  models for evaluation. However, our proposed method requires no re-calculation even though the model architecture has been changed. Thus, unlike transferability estimation measures, additional datasets can be calculated quickly and used for other tasks.

## 8 Conclusion

In this paper, we suggest using transfer learning for temporal neural networks using a proposed multi-source pre-training. Specifically, we demonstrate that by combining multiple datasets into a super dataset using pre-processing and adjusting the classification task with the concatenation of the classes, it is possible to pre-train a network using a large amount of data and classes.

Furthermore, in order to select appropriate datasets out of the large number of possible datasets that exist, we propose a new transferability measure based on shapelets. Our novel method calculates the distance between datasets using a shapelet similarity. The shapelet-based distance compares the class-discriminative shapelets between classes of the target dataset and the source dataset. We demonstrate that by using multi-source transfer learning with our shapelet similarity-based source selection, it is possible to increase the time series classification accuracy with little downside.

In future work, we will investigate the combinations of source datasets to optimize our proposed method further. We hope to contribute to the time series classification community by continuing to expand upon these techniques.

## References

1. An, S., Bhat, G., Gumussoy, S., Ogras, U.: Transfer learning for human activity recognition using representational analysis of neural networks. *ACM Transactions on Computing for Healthcare* **4**(1), 1–21 (2023)
2. Bai, S., Kolter, J.Z., Koltun, V.: An empirical evaluation of generic convolutional and recurrent networks for sequence modeling. *arXiv preprint arXiv:1803.01271* (2018)
3. Bao, Y., Li, Y., Huang, S.L., Zhang, L., Zheng, L., Zamir, A., Guibas, L.: An information-theoretic approach to transferability in task transfer learning. In: *ICIP* (2019)

4. Clark, R., Doyle, T.E.: A priori quantification of transfer learning performance on time series classification for cyber-physical health systems. In: IEEE CCECE (2022)
5. Cui, Z., Chen, W., Chen, Y.: Multi-scale convolutional neural networks for time series classification. arXiv preprint arXiv:1603.06995 (2016)
6. Dai, W., Yang, Q., Xue, G.R., Yu, Y.: Boosting for transfer learning. In: ICML (2007)
7. Dau, H.A., Keogh, E., Kamgar, K., Yeh, C.C.M., Zhu, Y., Gharghabi, S., Ratanamahatana, C.A., Yanping, Hu, B., Begum, N., Bagnall, A., Mueen, A., Batista, G., Hexagon-ML: The ucr time series classification archive (October 2018), [https://www.cs.ucr.edu/~eamonn/time\\_series\\_data\\_2018/](https://www.cs.ucr.edu/~eamonn/time_series_data_2018/)
8. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
9. Dridi, A., Afifi, H., Mounghla, H., Boucetta, C.: Transfer learning for classification and prediction of time series for next generation networks. In: IEEE ICC (2021)
10. Fawaz, H.I., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Transfer learning for time series classification. In: IEEE ICBD (2018)
11. Hochreiter, S., Schmidhuber, J.: Long short-term memory. *Neural computation* **9**(8), 1735–1780 (1997)
12. Huang, L.K., Huang, J., Rong, Y., Yang, Q., Wei, Y.: Frustratingly easy transferability estimation. In: ICML. pp. 9201–9225 (2022)
13. Huang, P., Wang, G., Qin, S.: Boosting for transfer learning from multiple data sources. *Pattern Recognition Letters* **33**(5), 568–579 (2012)
14. Ismail Fawaz, H., Forestier, G., Weber, J., Idoumghar, L., Muller, P.A.: Deep learning for time series classification: a review. *Data mining and knowledge discovery* **33**(4), 917–963 (2019)
15. Iwana, B.K., Uchida, S.: An empirical survey of data augmentation for time series classification with neural networks. *PLOS ONE* (2021)
16. Karim, F., Majumdar, S., Darabi, H., Chen, S.: Lstm fully convolutional networks for time series classification. *IEEE Access* **6**, 1662–1669 (2017)
17. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
18. Le Guennec, A., Malinowski, S., Tavenard, R.: Data augmentation for time series classification using convolutional neural networks. In: ECML/PKDD WAALTD (2016)
19. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. *Proc. IEEE* **86**(11), 2278–2324 (1998)
20. Li, J., Qiu, S., Shen, Y.Y., Liu, C.L., He, H.: Multisource transfer learning for cross-subject EEG emotion recognition. *IEEE Transactions on Cybernetics* pp. 1–13 (2019)
21. Lotte, F., Guan, C.: Learning from other subjects helps reducing brain-computer interface calibration time. In: IEEE ICASSP (2010)
22. Lu, H., Wu, J., Ruan, Y., Qian, F., Meng, H., Gao, Y., Xu, T.: A multi-source transfer learning model based on lstm and domain adaptation for building energy prediction. *Int. J. of Electrical Power & Energy Systems* **149**, 109024 (2023)
23. Maray, N., Ngu, A.H., Ni, J., Debnath, M., Wang, L.: Transfer learning on small datasets for improved fall detection. *Sensors* **23**(3), 1105 (2023)
24. Nguyen, C., Hassner, T., Seeger, M., Archambeau, C.: LEEP: A new measure to evaluate transferability of learned representations. In: ICML. pp. 7294–7305 (2020)

25. Petitjean, F., Ketterlin, A., Gançarski, P.: A global averaging method for dynamic time warping, with applications to clustering. *Pattern Recognition* **44**(3), 678–693 (2011)
26. Ren, R., Yang, Y., Ren, H.: EEG emotion recognition using multisource instance transfer learning framework. In: *ICICML* (2022)
27. SAKOE, H., CHIBA, S.: Dynamic programming algorithm optimization for spoken word recognition. *Readings in Speech Recognition* pp. 159–165 (1990)
28. Schuster, M., Paliwal, K.: Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing* **45**(11), 2673–2681 (1997)
29. Senanayaka, A., Al Mamun, A., Bond, G., Tian, W., Wang, H., Fuller, S., Falls, T., Rahimi, S., Bian, L.: Similarity-based multi-source transfer learning approach for time series classification. *International Journal of Prognostics and Health Management* **13**(2) (2022)
30. Serra, J., Pascual, S., Karatzoglou, A.: Towards a universal neural network encoder for time series. In: *AIRD*. pp. 120–129 (2018)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition (2015)
32. Song, H.J., Park, S.B.: Identifying intention posts in discussion forums using multi-instance learning and multiple sources transfer learning. *Soft Computing* **22**(24), 8107–8118 (2017)
33. de Souza, A.F.M., Cassenote, M.R.S., Silva, F.: Transfer learning of shapelets for time series classification using convolutional neural network. In: Britto, A., Valdivia Delgado, K. (eds.) *Intelligent Systems*. pp. 325–339 (2021)
34. Tan, B., Zhong, E., Xiang, E.W., Yang, Q.: Multi-transfer: Transfer learning with multiple views and multiple sources. In: *SIAM ICDM* (2013)
35. Tanisaro, P., Heidemann, G.: Time series classification using time warping invariant echo state networks. In: *IEEE ICMLA*. pp. 831–836 (2016)
36. Thompson, A.: Transfer learning with time series prediction: Review. *SSRN Electronic Journal* (2022)
37. Tran, A., Nguyen, C., Hassner, T.: Transferability and hardness of supervised classification tasks. In: *ICCV* (2019)
38. Wang, Z., Yan, W., Oates, T.: Time series classification from scratch with deep neural networks: A strong baseline. In: *IJCNN* (2017)
39. Weber, M., Auch, M., Doblender, C., Mandl, P., Jacobsen, H.A.: Transfer learning with time series data: A systematic mapping study. *IEEE Access* **9**, 165409–165432 (2021)
40. Yao, B., Ling, G., Liu, F., Ge, M.F.: Multi-source variational mode transfer learning for enhanced pm2. 5 concentration forecasting at data-limited monitoring stations. *Expert Systems with Applications* **238**, 121714 (2024)
41. Yao, Y., Doretto, G.: Boosting for transfer learning with multiple sources. In: *IEEE CVPR* (2010)
42. Ye, L., Keogh, E.: Time series shapelets. In: *ACM KDD* (2009)
43. Ye, R., Dai, Q.: Implementing transfer learning across different datasets for time series forecasting. *Pattern Recognition* **109**, 107617 (2021)
44. Yeh, C.C.M., Zhu, Y., Ulanova, L., Begum, N., Ding, Y., Dau, H.A., Silva, D.F., Mueen, A., Keogh, E.: Matrix profile i: All pairs similarity joins for time series: A unifying view that includes motifs, discords and shapelets. In: *IEEE ICDM* (2016)
45. You, K., Liu, Y., Wang, J., Long, M.: Logme: Practical assessment of pre-trained models for transfer learning. In: *ICML*. pp. 12133–12143 (2021)
46. Zhuang, F., Qi, Z., Duan, K., Xi, D., Zhu, Y., Zhu, H., Xiong, H., He, Q.: A comprehensive survey on transfer learning. *Proceed. IEEE* **109**(1), 43–76 (2021)