Online Few-Shot Time Series Classification for Aftershock Detection

Sheng Zhong
The University of New Mexico,
Albuquerque, New Mexico, USA
zhongs@unm.edu

Vinicius M. A. Souza
Pontifícia Universidade Católica do Paraná,
Curitiba, Paraná, Brazil
vinicius.email@gmail.com

Glenn Eli Baker
The Air Force Research Laboratory,
Albuquerque, New Mexico, USA
glenn.baker.3@us.af.mil

Abdullah Mueen
The University of New Mexico,
Albuquerque, New Mexico, USA
abdullah.al.mueen@gmail.com

ABSTRACT

Seismic monitoring systems sift through seismograms in real-time, searching for target events, such as underground explosions. In this monitoring system, a burst of aftershocks (minor earthquakes occur after a major earthquake over days or even years) can be a source of confounding signals. Such a burst of aftershock signals can overload the human analysts of the monitoring system. To alleviate this burden at the onset of a sequence of events (e.g., aftershocks), a human analyst can label the first few of these events and start an online classifier to filter out subsequent aftershock events. We propose an online few-shot classification model `FewSig` for time series data for the above use case. The framework of `FewSig` consists of a selective model to identify the high-confidence positive events which are used for updating the models and a general classifier to label the remaining events. Our specific technique uses a selective model based on sliding DTW distance and a general classifier model based on distance metric learning with Neighborhood Component Analysis (NCA). The algorithm demonstrates surprising robustness when tested on univariate datasets from the UEA/UCR archive. Furthermore, we show two real-world earthquake events where the `FewSig` reduces the human effort in monitoring applications by filtering out the aftershock events.

CCS CONCEPTS

• **Information systems** → **Data mining**; • **Applied computing** → **Earth and atmospheric sciences**; • **Computing methodologies** → *Online learning settings*; *Semi-supervised learning settings.*

KEYWORDS

Earthquake, Aftershock, Seismic Monitoring, Time Series Classification, Few-Shot Learning, Online Learning

ACM Reference Format:
Sheng Zhong, Vinicius M. A. Souza, Glenn Eli Baker, and Abdullah Mueen. 2023. Online Few-Shot Time Series Classification for Aftershock Detection.

KDD '23, August 6–10, 2023, Long Beach, CA, USA

ACM ISBN 979-8-4007-0103-0/23/08…$15.00
https://doi.org/10.1145/3580305.3599879

In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '23), August 6–10, 2023, Long Beach, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3580305.3599879

1 INTRODUCTION

In offline semi-supervised few-shot classification, a classification model is learned from a small number of positive instances, an arbitrary number of negative instances, and a sufficiently large number of unlabeled instances. In **online few-shot classification**, the unlabeled instances are time ordered and are only available to the model when they occur. The model is then incrementally updated with unlabeled data over time, unlike PU-learning [23] and other semi-supervised learning approaches [29, 34, 35] that exploit unlabeled data all at once. Moreover, in the online setting, a human expert labels only the first few instances from a stream instead of a few of the most representative instances from a large pool of unlabeled instances.

In addition to the small number of training instances, online few-shot learning poses two key challenges. First, the online classification process requires each test instance to be classified before the next instance arrives. This imposes a serious efficiency constraint, challenging computationally expensive algorithms for this task. Second, it must be determined whether the newly classified positive instances are classified with sufficiently high confidence that they should be added to the training set before potentially re-training the model.

In this paper, we propose a two-level framework. In the first level, a classifier exploits a pre-computed distance matrix under dynamic time warping (DTW) [26] distance to identify the high-confidence positive instances from unlabeled instances by bounding the false-positive rate at a maximum and adding them to the training set. In the second level, an ensemble classifier based on distance metric learning with Neighborhood Component Analysis (NCA) is trained with Focal Loss to tackle class imbalance. This classifier evaluates the unclassified instances from the first level. We demonstrate that this framework is significantly more accurate than existing semi-supervised algorithms in the online few-shot setting.

1.1 Motivation

We consider online few-shot classification for time series data with application to seismic monitoring. Seismic monitoring is an online task essential for national security and public safety. Current

seismic monitoring systems are not fully automated and require human analysts to review the information produced by algorithms to ensure accuracy. The amount of time an analyst takes to review a block of data (i.e., time series) is driven mainly by the number of events in that block and the amount of manual work needed to form each event completely. Large events can take longer to review as they are observed at more stations, and many of these arrivals may not be detected and associated automatically.

For example, prior to the 2011 Tohoku Earthquake and Tsunami event (the strongest earthquake recorded in Japanese history), the Late Event Bulletin (LEB [1]) of the International Data Center (IDC www.ctbto.org) averaged 120 events per day with approximately 2,000 time-defining associated arrivals (i.e., seismic signals). In the immediate aftermath of Tohoku, the LEB contained 830 events per day with approximately 20,000 time-defining associated arrivals. This alone is a 7× to 10× increase in the analyst workload [25]. In addition, the standard STA/LTA [2, 15] detectors become less effective in detecting rapid aftershocks, requiring the analysts to add more signals manually and associate them to these aftershocks.

With an increasing streaming workload, possible mediations are increasing the number of analysts (i.e., resource) and/or delay reporting (i.e., admit vulnerability). A real-time aftershock detector can reduce the manual workload significantly if it does not admit more errors than humans. However, such a detector poses several computational challenges. First, there is no training data until the main event happens. The old/historical events and their aftershocks are worthless because their epicenters are unlikely to be the same as the epicenter of the current event. Hence, historical signals do not bear any more similarity to the aftershocks than the similarity they bear to any other earthquakes. Second, the waveforms of the same event vary when observed at different stations because the waves take different paths through the earth. However, the current number of stations on Earth is too little to capture enough training data for a station-agnostic model. Therefore, the classical computer vision approach[31] that trains an offline model and adapts it to new events is not an option.

To address these challenges, we aim to exploit the similarities of the main earthquake and the first several aftershocks to the later aftershock signals and do so in real-time. The similarity of the aftershock signals is due to the source characteristics and the propagation paths from the sources to the stations, which are the two factors that control the waveforms, being similar. We must use only a few training instances of the aftershocks (i.e., the positive class) that we can collect shortly after the main shock. The detector must work in real-time to reduce human workload as well as to improve itself by learning from recent events. In other words, each event must be classified before the next one arrives.

The main contributions of this work are summarized below:

- We develop a novel online few-shot time series model (`FewSig`), that can be trained on a few positive signals, and adapt to the new unlabeled signals iteratively.
- We evaluate `FewSig` on 68 datasets and compare them with online versions of existing semi-supervised time series classification algorithms. The comprehensive evaluation details the parameter sensitivity, efficiency, and effectiveness of the proposed model.
- We evaluate the online performance of `FewSig` for the aftershock detection task on two earthquakes, the 2015 Gorkha earthquake in Nepal (Mw 7.8) and the 2017 Chiapas earthquake in Mexico (Mw 8.2).

2 RELATED WORK

2.1 Semi-supervised learning on time series

Semi-supervised learning (SSL) methods have been proposed to avoid poorly generalizable models due to insufficient labeled data to train a supervised model. Wei's Algorithm [34] and DTWD [4] are examples of self-training methods, a well-known semi-supervised learning approach. Souza et al. [30] perform clustering to select the most representative instances from an unlabeled dataset to be labeled by an expert and then perform label propagation to classify the remaining instances. Nguyen et al. [23] proposed a solution based on clustering and self-training, considering only positive and unlabeled instances. Marussy et al. proposed SUCCESS [16], a model based on constrained hierarchical clustering. Xu et al. proposed [36] based on the graph-theoretic SSL algorithm. SSSL [32] performs self-training with shapelet classification on unlabeled instances. Several state-of-the-art deep-learning models have shown dominance in recent works. For example, Jawed et al. proposed a multi-task learning network (MTL) [12] to jointly train the ConvNet with classification and forecasting by sharing latent representations. SemiTime [6] shows better results than MTL on some datasets, the model learns past-future temporal relations from the unlabeled data, and the backbone feature extractor is shared with the TSC module which is trained on the labeled dataset. SSTSC [35] increases the richness of the temporal context by splitting a time series into past-anchor-future, making the model learn a higher-quality semantic context from the unlabeled dataset.

Even with good results in the semi-supervised scenario, it is worth mentioning that none of these works are adequate for the online few-shot learning setting proposed in this work, in which an unlabeled dataset does not exist to help train the initial model.

2.2 Few-shot learning

A wide range of applications encounters inherent constraints such as privacy and safety issues, ethical issues, or prohibitive costs of manual analysis, which require training a supervised model using a limited number of labeled instances per class. The paradigm of few-shot learning [33] has been designed to tackle this challenge. Existing models, such as MAML [7] and ProtoNet [28], exploit task-agnostic knowledge acquired during the meta-training stage for faster learning in new tasks, assuming the data for the training and testing tasks are from the same distribution [10, 24]. However, aftershock sequences inherently present out-of-distribution datasets. The origins of different aftershock sequences are seldom in the same region, and a single event may not manifest any similarity across seismometers located in different places due to path variation. The complexity of the Earth and the sparse distribution of seismometers make this task exceedingly challenging for these models.

The complexity further amplifies when considering cross-domain learning [10]. As noted by Gao et al. in work [10], the performance of the same meta-learning model can drastically differ across various target domains. This is also true for time series data, as indicated

by [22], where the meta-learning approach did not significantly differ from the DTW-based 1-NN classifier when the meta-training and meta-testing data exhibit different distributions.

In contrast, FewSig is designed to train with a few initial positive instances and some negative instances from the target task, eliminating the need for either pre-training or meta-learning.

2.3 Aftershock Detection

The existing aftershock detection works can be grouped into two categories, one is template matching [27], and the other one is deep neural networks [5, 37]. The template matching techniques have high interpretability, but the performance relies on the quality and the quantity of the templates, which are very challenging to acquire in a short period after the main shock. For instance, Seiscorr [27] needed hundreds of aftershocks following a large earthquake, which occurred over the first six hours of data. That requirement makes this method inadequate for online monitoring applications.

The problem is exacerbated in deep learning solutions. Supervised deep neural networks suffer when the training set has only a few instances of aftershocks which leads to a long waiting time to collect enough data. Consider Zhang et al. [37], where authors trained on thousands of aftershocks from the first 25 days to achieve only a 67.5% F1-score at the highest on the last five days of aftershocks. In comparison, FewSig achieves the high nineties even with a few positive instances shortly after the main shock. Realizing this caveat in using historical data, other researchers [5] have developed PNN(paired neural network) to learn a similarity function from historical events, however, the trained model is yet to be evaluated on real earthquake events, while we assess on multiple large scale real seismic events. The problem is too complex to be solved using available historical data, which prevents us from exploiting attractive alternatives such as rule-based classifiers, support vector machines, etc.

3 ONLINE FEW-SHOT TIME SERIES CLASSIFICATION

In this section, we formulate the problem and then introduce the technical framework of FewSig. Table 1 summarizes the notations and abbreviations.

We define the online few-shot classification task as follows; there are three datasets, a labeled set L, an online testing set O, and a training set T. $L = \{(t^1, y^1), (t^2, y^2), ..., (t^l, y^l)\}$ with true labels available, only a few (i.e., five) times series in L belong to c^+, and the remaining belong to c^-; O is the online testing set that saves all the new time series pending to be classified; T is the training set for training a model. Initially, O is empty, and the model is trained on T, and $T = L$. During the online testing phase, each new time series t^j will be added to O, $O = O \cup \{t^j\}$, and the model predicts its label $\hat{y}^j$. Finally, the model performance is evaluated by comparing the predicted label with the true label for all time series in O. During the online evaluation process, the model can be retrained at any moment with T that contains L and any number of time series from O that have already been classified. Note that $\hat{y}^i$ is immutable, which means the model cannot leverage the current time series to update the previously predicted results.

Table 1: Symbols and notations.

Symbol	Definition
t^i	Time series i
$t^i_{s,m}$	A sub-sequence of length m in t^i that contains $t^i[s : s+m-1]$
	$m \leq$length of time series, assuming equal length for all time series
y^i	True label for t^i
$\hat{y}^i$	Predicted label for t^i
T	Training set.
N	Size of the training set T, which is the number of time series in T
L	Labeled set
O	Online testing set
$d_{i,j}$	Distance between t^i and t^j.
D^i	Distance feature array of t^i, $D^i = [d_{i,1}, d_{i,2}, ..., d_{i,N}]$
c^+	Positive class
c^-	Negative class
d_i^{k+}	Average distance between t^i and k nearest neighbors from class c^+
d_i^{k-}	Average distance between t^i and k nearest neighbors from class c^-
tFPR	Target FPR for the selective model

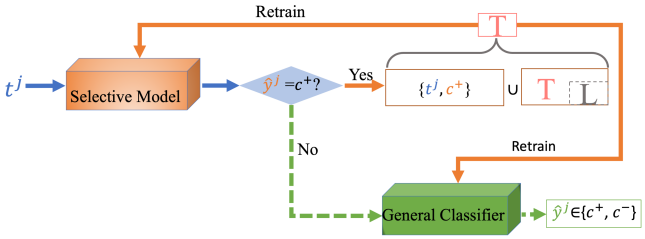

Figure 1: Online classification with self-training.

We propose a general framework shown in Figure 1 for this task. There are two models: a selective model to identify the high-confidence positive time series and a general classifier model to re-evaluate the rest of the time series, the high-confidence positive time series will be added to T for retraining. Initially, both models are trained on T and $T = L$. For each new time series t^j, the solid orange path in Figure 1 represents the case when t^j is identified as a high-confidence positive by the selective model, T will be expanded with $\{(t^j, c^+)\}$ and both models will be retrained on the updated T and finally $\hat{y}^j = c^+$; The green dotted path demonstrates the other case when t^j is not a high-confidence positive, the general classifier will evaluate t^j independently for the second time and yield the final label $\hat{y}^j$.

The framework does not add an explicit stopping criteria for the growth of the training set, T. This is rather an empirical choice because the frequency of aftershocks is the highest immediately after the main shock and decreases rapidly over days. Thus, although the training set is growing, the time between events to exploit the training set is also increasing. Hence, we avoid limiting the growth of the T. If event frequency stays high for longer, we recommend limiting the training set to maintain the required rate.

3.1 Distance/Dissimilarity as features

We use sliding Dynamic Time Warping (DTW) as a shift and warping invariant distance metric to compute the distance among time series. The sliding Euclidean distances effectively correct alignment

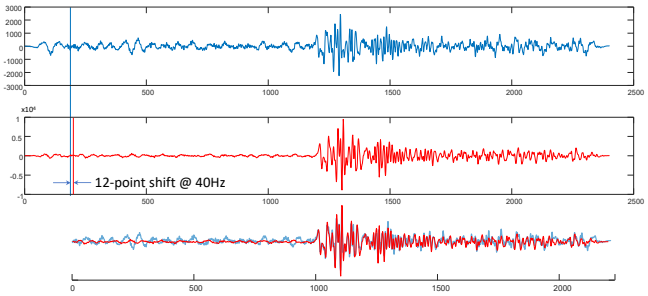

Figure 2: Two real aftershock signals show the highest correlation of 80.84% when one is shifted in time relative to the other to correct for human error in picking, and near 0% correlation if not shifted.

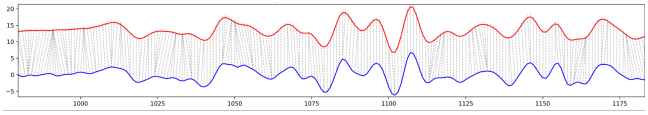

Figure 3: The DTW alignment between the middle segments of signals in Figure 2, the warping band is limited to 3 points. The blue signal exhibits warping due to the variation in wave propagation from the event origin to the seismometer. The correlation after correcting warping is 89.85%.

errors [21]. Still, it is very sensitive to small mismatches [13] due to signal warping. We use sliding DTW distances to address two sources of errors in the time series. The first is due to errors in alignment. For example, human analysts often pick the onset time of a seismic phase's arrival in seismograms, and any picking error can alter results dramatically, as shown in Figure 2. The second error is due to the inherent complexity of the system being monitored. For example, the earth's non-uniformity changes the waves' propagation from neighboring events. Figure 3 shows how DTW deals with warping by allowing one to many alignments to correct tiny variations in the time series.

$$d_{i,j} = min\{SS(t^i, t^j, s_1, s_2, q, m, r), \quad SS(t^j, t^i, s_1, s_2, q, m, r)\} \quad (1)$$

$$SS(t^i, t^j, s_1, s_2, q, m, r) = min\{DTW(t^i_{s,m}, t^j_{q,m}, r), s_1 \le s \le s_2\} \quad (2)$$

We formalize $d_{i,j}$ as the distance between t^i and t^j in Equation 1. $d_{i,j}$ is the smallest DTW distance between two subsequences $t^i_{q,m}$ in t^i and $t^j_{s,m}$ in t^j for given q, m, s_1, s_2, r. We perform a similarity search with a sliding window to find this subsequence pair. The search process defined in Equation 2 is the same approach employed in [26]. We take one subsequence from a time series as a query and search for the most similar one in another time series and vice versa, $d_{i,j}$ is the minimum distance during this process. r in Equation 2 is the Sakoe-Chiba Band [19], q, m defines the query sub-sequence, and s_1, s_2 defines the search boundaries. We recommend readers to tutorials [19, 20] for details of these parameters.

$$d_i^{k+} = \frac{\sum \text{k\_smallest}(\{d_{i,j} | j \neq i \ \& \ t^j \in T \ \& \ y^j = c^+\})}{k} \quad (3)$$

$$\texttt{DistLoss} = \sum_{t^i \in T} \begin{cases} max(0, d_i^{k+} - d_i^{k-})/m & \text{if } y^i = c^+ \\ max(0, d_i^{k-} - d_i^{k+})/m & \text{if } y^i = c^- \end{cases} \quad (4)$$

We perform a search with some candidate values to find the optimal parameter configuration $\{k^*, q^*, m^*, s_1^*, s_2^*, r^*\}$ that yields the lowest loss defined in Equation 4 for a given dataset. The training set only contains instances from L with true labels available at this stage. d_i^{k+} defined in Equation 3 is the mean value of the k smallest distances between t^i and other positive time series in T; d_i^{k-} is defined in a similar way. To handle the bias of different query lengths m, each distance is normalized by m when computing the loss. The optimal parameter configuration remains constant during the online evaluation phase. We set $k^* = 1$ for simplicity, the candidate values for the query are the middle subsequences with a length of 80% to 90% of the entire time series, the searching boundaries are a few seconds wider than the query on both ends, and the Sakoe-Chiba Band in a few samples range, i.e., from 0 to 10. More parameter candidates will increase the model initiation time, and some domain knowledge can be applied here for a smaller set of parameter candidates that yield faster model launching time.

3.2 Selective model

$$\hat{y}^i = \begin{cases} c^+ & \text{if } d_i^{k+} < h_+^* \ \& \ d_i^{k-} >= h_-^* \\ c^- & \text{other cases} \end{cases} \quad (5)$$

The selective model must have a very low FPR since any false-positive instance can have long-term detrimental effects. We propose a model that classifies events based on the conditions expressed in Equation 5. Two thresholds h_+^* and h_-^* are established based on the training set for a given $tFPR$ (target FPR) value.

We demonstrate the learning process with the entire T of size N. We enumerate all possible split values, $\{h_+^i, 0 \le i \le N\}$ and $\{h_-^i, 0 \le i \le N\}$. h_+^i and h_-^i are defined in Equation 6, $ds_i^{k+}, 1 \le i \le N$ is the ith smallest value in the sorted d_i^{k+} array for all $t^i \in T$, δ is a very small number. The learning process can be summarized in Equation 7 as finding two thresholds h_+^*, h_-^* such that the FPR is closest to $tFPR$ while TPR is highest when evaluated with the entire training set. t^i will be classified as positive only when $d_i^{k+} < h_+^*$ **and** $d_i^{k-} \ge h_-^*$.

$$h_+^i = \begin{cases} ds_1^{k+} - \delta & i == 0 \\ (ds_{i+1}^{k+} + ds_i^{k+})/2 & 0 < i < N \\ ds_N^{k+} + \delta & i == N \end{cases} \quad (6)$$

$$h_+^*, h_-^* = \underset{\{h_+^i, h_-^i\}}{\arg\max} \, TPR(1 - |FPR - tFPR|) \quad (7)$$

The optimal k^* in both d_i^{k+} and d_i^{k-} are selected from a candidate set $k = \{1, 2, ..., 8\}$ with highest F1 score by performing *Leave-One-Out* evaluation on T. Once k^* is estimated, h_+^*, h_-^* are selected again with loss function in Equation 7 on the entire training set T. Note that the k^* needs to be re-calculated each time when retraining the model.

3.3 General time series classifier

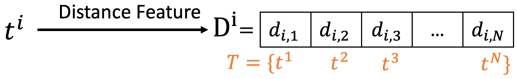

Figure 4: Distance features for a time series t^i.

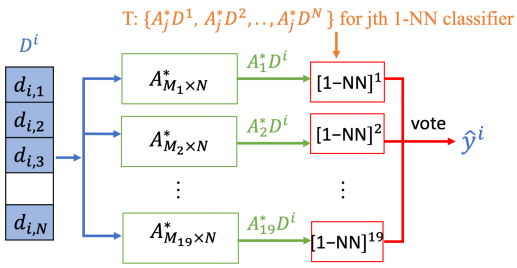

Figure 5: Proposed general time series classifier.

We utilize the distance feature D^i to represent t^i as shown in Figure 4 for our proposed general classifier enlightened by the existing time series classification works [8, 11, 13]. Each element in D^i is the distance between t^i and a time series in T. The length of D^i is dependent on the size of T. If a new time series t^j is added to T, then all the time series in T need to update the features by appending $d_{i,j}$ at the tail. The feature for each upcoming new time series is computed with the latest T.

Jain pointed out in [11] that not all the dimensions in D^i contribute equally during the classification, thus we consider removing less significant dimensions and boosting significant dimensions by performing Neighborhood Component Analysis (NCA [9]) with Focal-Loss [14]. This is achieved by learning a linear transformation matrix A such that the 1-NN classifier performs well under this transformed space. We define the learning process in Equations 8, 9,10,11. FL is Focal loss, p_{t_i} is the probability that t^i is correctly classified. $\alpha \in [0, 1]$ is a balancing factor for addressing the class imbalance. γ is the focusing parameter that can make learning more focused on the hard misclassified positive ones rather than numerous simple negative ones. We use $\alpha = 0.5, \gamma = 2$ recommend in the original work [14] across all the experiments.

$$A^*_{M \times N} = \underset{A_{M \times N}}{\operatorname{argmin}} \sum_{t^i \in T} FL(t_i) \tag{8}$$

$$FL(t_i) = -\alpha(1 - p_{t_i})^\gamma \log p_{t_i} \tag{9}$$

$$p_{t_i} = \begin{cases} p_i & \text{if } y^i = c^+ \\ 1 - p_i & \text{if } y^i = c^- \end{cases} \tag{10}$$

$$p_i = \sum_{y^j = c^+, j \neq i} \frac{e^{-\|AD^i - AD^j\|^2}}{\sum_{k \neq i} e^{-\|AD^i - AD^k\|^2}} \tag{11}$$

$$\overline{M} = \{a \lfloor \frac{N}{20} \rfloor, a \in \{1, 2, 3, ..., 19\}\} \tag{12}$$

Instead of using only one matrix $A \in \mathbb{R}^{M \times N}$, N is number of time series in T, and M is the reduced feature dimension, we learn nineteen A with various dimensions $\overline{M} = \{M_1, M_2, ..., M_{19}\}$. $\overline{M}$ can be estimated based on the best *Cross Validation* performance on the training set or based on a certain level of randomness. For this work, we applied the same approach described in [11] that can be expressed in Equation 12. Each matrix A is trained separately, and each space will have an independent 1-NN classifier trained with transformed distance features of T. The final predicted label $\hat{y}^i$ is voted on among the 19 1-NN classifiers. $vote = v$ means the $\hat{y}^i = c^+$ as long as there are v 1-NN classifiers classifying t^i as c^+. The final structure with our proposed model is shown in Figure 5.

4 EXPERIMENTAL EVALUATION

We compare FewSig with four semi-supervised models adapted for the online few-shot classification settings on 68 datasets from the UEA/UCR Time Series Repository [3] covering various domains. All our experiments are reproducible, the source code, data, and additional results are available on our supporting website [38]. We performed all the experiments on an AMD EPYC 7402 server (24 cores) with a 4xRTX3090 GPU and 128GB RAM.

4.1 Models for comparison

We select three traditional semi-supervised time series classification models: Wei's model [34], DTWD [4], SUCCESS [16] and one state-of-the-art semi-supervised model SSTSC [35] based on deep learning. Next, we briefly describe the original algorithms and how we adopt them in the online setting.

Wei's model uses a one-nearest-neighbor (1-NN) classifier as the base model. Initially, L contains only a few labeled positive time series and $T = L$. To expand T, the algorithm iteratively selects the high-confidence positive time series from the unlabeled set measured by the nearest neighbor Euclidean distance. When iteration stops, all time series in T are labeled as c^+, and the remaining ones in the unlabeled set are c^-.

We take the following procedures to adapt the model for the online setting: 1) Initially $T = L$, L includes both c^+ and c^-. 2) The 1-NN classifier will classify each new time series t^j, if $\hat{y}^j = c^+$, t^j will be added to T and retrain the model. 3) If the model is updated, then all the previous $t^i \in O$ that are not in T will be reconsidered whether to add in T. 4) The model will be retrained immediately after T is expanded in step 3, and step 3 is repeated. 5) The iteration will stop when T is not expanded in step 3.

DTWD applied a similar approach for augmenting T via self-training on an unlabeled set. It employs a new distance measure and a one-class classifier. The distance measure is the ratio of DTW distance to Euclidean distance. The one-class classifier relies on the entire unlabeled set to pick the optimal parameters however, this is impractical in the online scenario. Thus, we consider Wei's approach with the distance measure employed by DTWD.

SUCCESS is based on the constrained single-linkage hierarchical agglomerative clustering algorithm with DTW distance. One constraint when linking the instances is that instances from L can not be linked. The final label of a cluster is decided by the majority class of instances in L. The unlabeled instances in a cluster share the same cluster label. Then a 1-NN classifier will be trained for testing. To work in the online setting, we first utilize SUCCESS to classify a new time series t^j, then rebuild the cluster hierarchy

with L and the existing O including t^j, and finally retrain the 1-NN classifier.

SSTSC learns an encoder that can capture the temporal context based on the unlabeled dataset in a self-supervised manner. This encoder will be used as a backbone for the supervised TSC module that is trained on the labeled dataset L. We consider the same approach employed for SUCCESS to adapt SSTSC for the online setting.

4.2 Online experimental protocols

We select 68 out of 128 univariate time series datasets from the UEA/UCR repository [3]. The selection is based on the size of the datasets, which should be less than 800 time series since `FewSig` trains on five positive time series, and increasing the number of test instances does not significantly impact the performance. Hence, we chose a cut-off for the datasets. For each dataset, we re-assign the labels, the minority class is considered as c^+ for unbalanced datasets, and a randomly chosen class as c^+ for balanced datasets. All other classes are considered as c^-.

We randomly order all the time series in a dataset due to the absence of event time. L consists of the first five positive time series and the first half of the negative time series; O consists of the rest of the time series. The time series in O is fed to the model sequentially in assigned order to simulate online scenarios. The scores are computed by comparing the $\hat{y}^j$ with y^j for all $t^j \in O$. To diminish the random bias, we perform 30 trials for each dataset, and we verify that none of the trials share more than two positive time series in L with other trials. The final scores of a model are averaged among 30 trials.

There is no parameter configuration for Wei's model, DTWD, and SUCCESS. For SSTSC, we use the same configuration described in the original paper [35]. We use the following settings for `FewSig` across all datasets: `tFPR`$=1\%$, $\alpha = 0.5$, $\gamma = 2$, and SDG with $lr = 0.02$, the final F1 score is averaged when `vote`$=\{1, 2, 3, 4\}$ separately.

4.3 Experimental Results

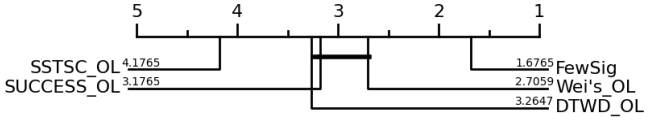

Figure 6: Critical difference diagram of 5 models on the 68 UEA&UCR benchmark datasets. $\rho = 0.05$ for Nemenyi test.

4.3.1 Ranking comparison. We conduct the Friedman test and Nemenyi post-hoc test on all 68 datasets across five models. The ranking is computed with the mean F1 scores of 30 trials per dataset. We show the critical difference diagram in Figure 6 in which `FewSig` ranks at the top with a significant statistical difference to the rival models. The average F1 for 68 datasets between `FewSig` and Wei's model, SSTSC are shown in Figure 7, `FewSig` has a higher F1 for most of the datasets in both cases. The results for each dataset are available on the supporting website [38].

4.3.2 Time complexity. We present the average running time of 68 datasets for each model in Table 2. We consider the time of initial training on L, the time for inferring each instance in O, and the time for model retraining. Since the model does not need to be retrained on each instance $t^j \in O$ for Wei's model, DTWD, and `FewSig`, the time is amortized on the entire testing set.

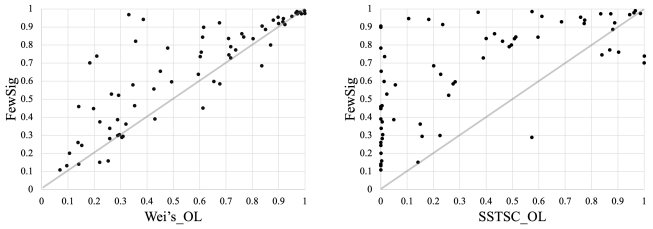

Figure 7: Mean F1 scores of 30 trails on 68 datasets comparison between `FewSig` and Wei's model (left), SSTSC (right).

Table 2: Average running time in seconds. k is the number of candidate parameter sets to find the optimal one.

Model	Initial time		Ave. Time per t^j	
	Distance	Train	Distance	Train+Infer
Wei's	1.49	0.0004	0.019	0.006
DTWD	5.01	0.0005	0.049	0.006
SUCCESS	3.517	0.0003	0.041	0.050
SSTSC	0	26.78	0	36.68
FewSig	8.571*k	14.46	0.057	0.79

Although `FewSig` performs significantly more accurately, `FewSig` is not as efficient as existing methods. This highlights that `FewSig` trades a bit of speed to gain accuracy. However, `FewSig` takes several minutes to initiate and processes roughly one event per second. This speed is sufficient for our target domain because the shortest time between successive events is 16 seconds in our online testing sets for both Nepal and Chiapas aftershock sequences.

4.3.3 Parameter sensitivity. In this section, we discuss the `FewSig` sensitivity to three design parameters: i) different vote numbers for the general classifier, ii) tFPR, and iii) the initial number of positive time series in L.

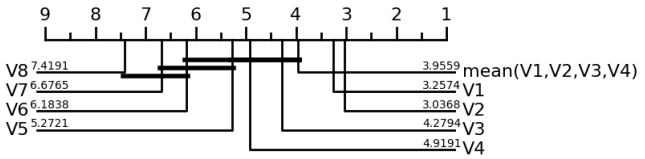

Figure 8: Critical difference of `FewSig` with the different number of votes on 68 datasets, $vote = 2$ yield highest rank.

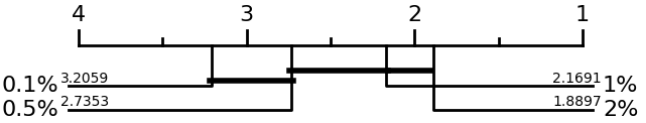

Figure 9: Critical difference diagram of FewSig with four different $tFPR$ values.

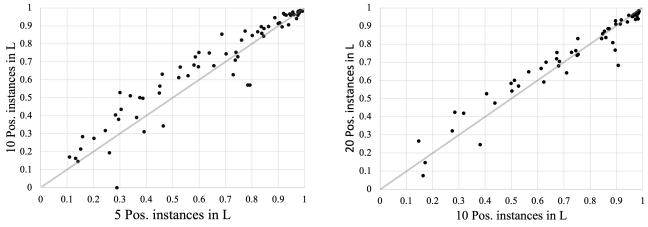

Figure 10: F1 score comparison between labeled sets L with the different number of positive instances for 68 datasets.

i) Figure 8 shows the F1-score ranking of FewSig with the different number of votes for the ensemble. If two classifiers agree on a positive instance, it gives the best overall results across all 68 datasets. ii) Figure 9 shows there is a significant difference in ranking when tFPR goes from one extreme (0.1%) to the other (2%). iii) In Figure 10 we show the F1 comparison with the different number of positive time series in L. The figure shows that doubling this number from 5 to 10 has more impact than doubling from 10 to 20.

5 AFTERSHOCK DETECTION

In this section, we evaluate FewSig performance for the online aftershock detection on two major earthquakes Mw 7.8 Nepal (Gorkha) earthquake in 2015 and the Mw 8.2 Chiapas earthquake in 2017.

5.1 Data preparation

We use the Late Event Bulletin (LEB) [1], which provides information about events such as time, location, and magnitude. The seismograms (i.e. time series data) are collected from the International Monitoring System (IMS), operated by the Preparatory Commission of the Comprehensive Nuclear-Test-Ban Treaty Organization (CTBTO). For both major earthquakes, we use local catalogs to confirm the aftershocks. We use the catalog from McNamara et al. [17] for the Nepal earthquake, and the catalog from the Mexican Servicio Sismológico Nacional (SSN) [18] for the Chiapas earthquake. We describe our data preparation procedures in the following steps.

Aftershock association We select the ground truth aftershock events by joining the local catalog with LEB. We call this process aftershock association. The association process is necessary to double confirm that an event was truly an aftershock. We use each aftershock event from the local catalog as a query to search for an associated event in the LEB. The join condition for an association is that the events must be within 200km of their Great-circle distance **and** their origin times are within 5 seconds. We break ties by taking the events that are closest in time.

Non-aftershock selection We consider all the seismic events in the LEB that are outside the geographical region of aftershock events as non-aftershocks. For this experiment, we only select non-aftershock events with similar origin-to-station distances compared to the aftershock events to create challenging cases for the classifier.

Arrival selection We get signal arrival times for the aftershock and non-aftershock events at a target seismometer in the LEB. For this experiment, we only use P phase signals, as P is the most common phase in the LEB.

Waveform processing Based on the arrival time, we extract three 60-second time series for each event from the continuous waveforms of three broadband channels BHZ (vertical), BHN (north-south), BHE (east-west) of a single seismometer. Thus, the time series contains 30 seconds of pre-arrival and 30 seconds of post-arrival signal. This time window is large enough to capture the initial compressional seismic waves generated by any regional earthquake. If the waveforms are sampled at 40Hz, the length of each waveform is 2,400 real numbers. Following conventional seismic signal preprocessing techniques, we remove the linear trend, and the mean value of each waveform, then taper the waveforms before filtering. We applied a 0.4Hz to 10Hz second-order Butterworth bandpass filter in both directions to cancel the phase shift. Next, we compute the Signal to Noise Ratio (SNR) on the filtered waveforms as the ratio of the standard deviation of the signal part (post-30 seconds) over the noise part (pre-30 seconds). For the experiments, we only use waveforms that have $\texttt{SNR} \geq 2$.

5.2 Features for multidimensional time series

$$SS(i, j, s_1, s_2, q, m) = min\Big\{\sum_{\beta \in \{Z,N,E\}} \alpha_\beta^j DTW(t_{s,m}^{i_\beta}, t_{q,m}^{j_\beta}, r), s_1 \leq s \leq s_2\Big\} \quad (13)$$

$$\alpha_\beta^i = \frac{max(\|t^{i_\beta}\|)}{max(\|t^{i_Z}\|)+max(\|t^{i_N}\|)+max(\|t^{i_E}\|)}, \beta \in \{Z, N, E\} \quad (14)$$

To accommodate three channel/dimension time series data, we modify Equation 2 to Equation 13 by combining three weighted distances of time series from the same channel. $t^{i_\beta}, \beta \in \{Z, N, E\}$ represents the time series from β channels for the event i. Equation 14 defines α_β^i which are the proportional weights of each channel.

We applied the same procedures described in section 3.1 to select the optimal parameters $\{q^*, m^*, s_1^*, s_2^*, r^*\}$ for Equation 13. In here we fix $q=5$, $m=50$, $s_1=4$ and $s_2=6$ seconds and compute the loss value defined in Equation 4 for each $r \in \{0, 1, ..., 10\}$ in number of points. We report $\{q^*, m^*, s_1^*, s_2^*, r^*\}$ applied in the result sections.

5.3 Experimental Settings

We reproduced the scenario when an earthquake happens to evaluate the real-world performance of FewSig. Initially, we have the labeled set containing the non-aftershocks before the main shock and the first **five** aftershocks. All the later events are in the online testing set O without knowing their labels. The model will sequentially classify each event in O based on their event time. Finally, we show the performance of each model over time regarding TPR, FPR and F1 scores.

Parameters for FewSig are constant for all the experiments: $\texttt{tFPR} = 0.5\%$, $\texttt{vote} = 2$, $\texttt{learning rate(lr)} = 0.02$, $\texttt{epoch} = 250$, $\gamma = 2$, $\alpha = 0.5$. The warping banding for computing the DTW varies at each station. We report the exact number in corresponding sections.

5.4 Results for the 2015 Nepal Earthquake

The origins of the events in LEB and the local catalog [17] are shown in Figure 11. The MKAR seismic station has the highest number of recorded arrivals in LEB among other IMS stations for the ground truth aftershock events. Thus we fetch all the seismograms for the experiment at MKAR.

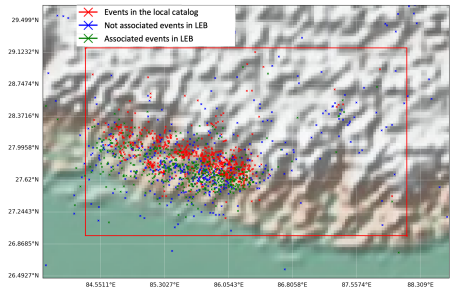

Figure 11: The distribution of origins for events in the LEB and the local catalog [17]. The events in the LEB for the 2015 Nepal earthquake are selected by limiting the origins to the red rectangular region and limiting the origin time from 2015-04-25T06:11:24.290000Z to 2016-05-14T22:45:53.330000Z.

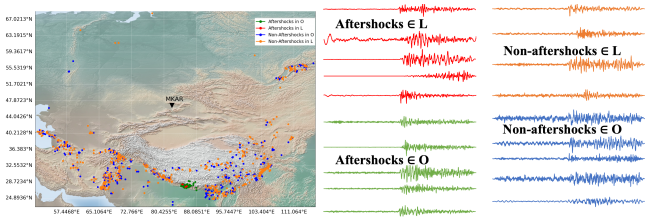

Figure 12: Left figure shows the geographical distribution of origins for extracted arrivals at MKAR. The right figure shows some example waveforms from the BHZ channel at MKAR. The first five aftershocks are in red. They were filtered with a 0.4Hz to 10Hz Butterworth bandpass filter.

We extracted 217 aftershock arrivals and 1182 non-aftershock arrivals at the MKAR based on the conditions described in the data preparation section, each arrival has three corresponding time series from three channels. Next, we put 5 aftershocks and 705 non-aftershocks in L, and 212 aftershocks and 477 non-aftershocks in O based on the event time. Figure 12 shows the origins of the extracted arrivals. We use $q = 5$, $m = 50$, $s_1 = 4$, $s_2 = 6$ seconds and $r = 1$ point for Equation 13.

We demonstrate the performance of `FewSig` and reference models over time regarding F1, TPR, and FPR in Figure 13 and 14. We can conclude that `FewSig` consistently leads the other models by around 0.2 on F1, 20% on TPR, and essentially maintains the lowest FPR. The F1-score of `FewSig` rapidly increases when more aftershocks are selected for the training set. F1 reaches 89% at the peak point after 157.2 hours since the main shock, then it gradually decreases when there are more non-aftershock events. Finally, the F1-score levels out at 0.85, the TPR does so at 82.55%, and the FPR at 5.4% for `FewSig`. During the online training process, FewSig is retrained 127 times. The origins of false positives and false negatives obtained by `FewSig` are demonstrated in Figure 15. False positives are harmful because human analysts will have to spend time correcting such errors if indeed they detect them. If they don't detect them, then target events could be missed. The scattering of false positives in multiple locations suggests `FewSig` making unbiased mistakes.

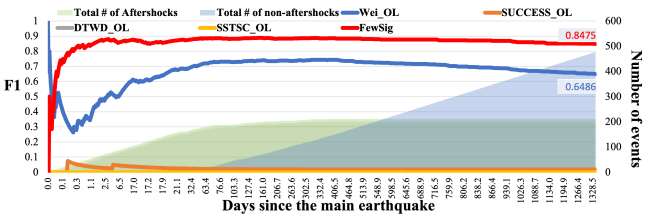

Figure 13: Online performance for classifying the Nepal aftershock sequence at MKAR. F1 scores of different models are shown on the solid curves. A point on a curve shows the score when testing the events at and before the time on the x-axis. The accumulated number of testing aftershocks and non-aftershocks are represented by the light green and blue shaded areas respectively. Both SUCCESS_OL and SSTSC_OL have F1 scores of zero throughout. We only use Z channel time series data for SSTSC_OL since it only supports univariate time series.

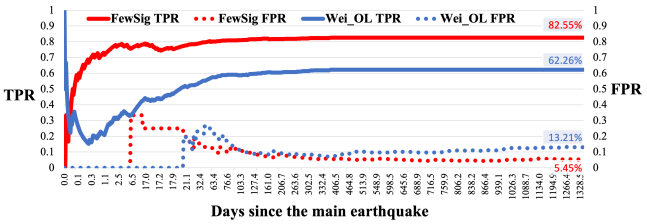

Figure 14: Online performance for classifying the Nepal aftershock sequence at MKAR. TPR and FPR scores of `FewSig` and Wei_OL are shown on the solid and dotted curves.

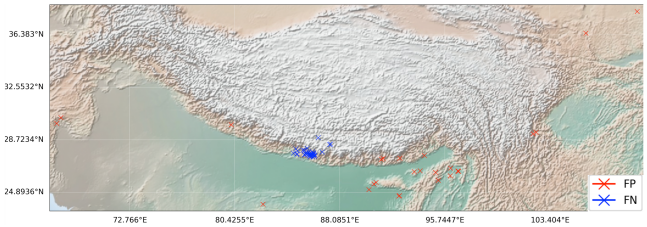

Figure 15: The origins of 26 false positives (red) and 37 false negatives (blue) obtained by `FewSig` overall.

5.5 Results for the 2017 Chiapas Earthquake

To test the universality of `FewSig`, we further examine the 2017 Chiapas aftershock sequence. We use the catalog of the Mexican Servicio Sismológico Nacional (SSN) [18] as the local catalog. The selection criteria we made on this catalog are: 1. The period from 2017-09-08 to 2018-03-08. 2. The latitude is from 14 to 17 and the longitude is from -96 to -93. The origins of events in the local catalog and LEB are shown in Figure 16. We choose the station TXAR for this earthquake since it has the most number of arrivals.

We balanced the number of non-aftershocks in L and O due to insufficient non-aftershock events after the main shock in the LEB. Then we have 5 aftershocks and 280 non-aftershocks in L, and

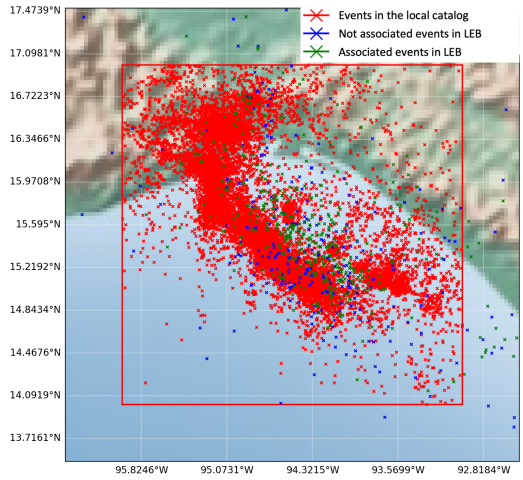

Figure 16: The origin distribution of events in LEB and the local catalog [18]. The events in LEB are selected by limiting the origins to the red rectangular region and limiting the origin time from 2017-09-08T04:49:17.000000Z to 2018-03-08T22:18:24.000000Z.

134 aftershocks and 280 non-aftershocks in O after balancing. The distribution of origins is shown in Figure 17.

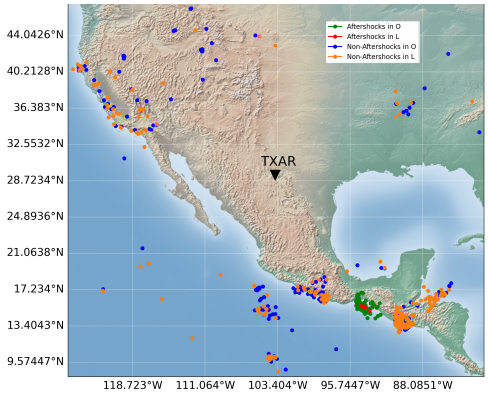

Figure 17: Origins for extracted arrivals at TXAR station.

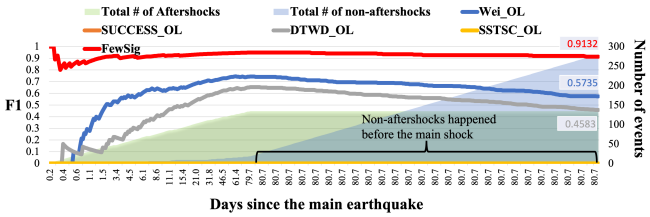

Figure 18: Online performance of each model while classifying the Chiapas aftershock sequence at TXAR.

Figures 18 and 19 are analogous to Figures 13 and 14, but for the Chiapas earthquake, FewSig outperforms the reference models

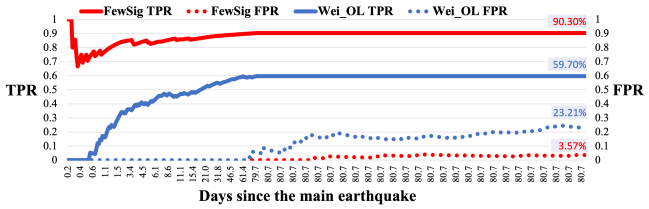

Figure 19: Online performance for classifying Chiapas aftershock sequence at TXAR. TPR, FPR scores of FewSig and Wei_OL are shown on the solid and dotted curves.

Figure 20: The origins of 10 false positives (red) and 13 false negatives (blue) obtained by FewSig overall.

by achieving an overall 0.91 F1 score, 90.3% TPR, and 3.57% FPR. During the online training process, FewSig is retrained 86 times. The overall false positives in Figure 20 are scattered and suggest FewSig is making unbiased errors.

6 CONCLUSION

In this paper, we present an online few-shot classifier for time series data. Our method only requires a few labeled positive instances and some labeled negative instances and can be gradually enhanced with the new unlabeled instances automatically. FewSig outperforms existing methods on 68 datasets from the UEA/UCR repository and achieves adequate performance for the online aftershock detection task when evaluated with two real-world earthquakes. Our model is lightweight and does not rely on a large labeled dataset which makes it applicable for online seismic monitoring systems to reduce the workload of analysts.

ACKNOWLEDGEMENTS AND DISCLAIMER

This material is based on work supported by The Air Force Research Laboratory under contract no. FA9453-21-2-0049 and National Science Foundation under #2104537. The views expressed are those of the author and do not necessarily reflect the official policy or position of the Department of the Air Force, the Department of Defense, or the U.S. government.

REFERENCES

[1] 1999. IDC Documentation, Processing of Seismic, Hydroacoustic, and Infrasonic Data (1999), (IDC/OPS/MA/001/Rev.1).

[2] Rex V Allen. 1978. Automatic earthquake recognition and timing from single traces. *Bulletin of the seismological society of America* 68, 5 (1978), 1521–1532.

[3] William Vickers Anthony Bagnall, Jason Lines and Eamonn Keogh. 2023. The UEA UCR Time Series Classification Repository. Retrieved Jan 19, 2023 from www.timeseriesclassification.com

[4] Yanping Chen, Bing Hu, Eamonn Keogh, and Gustavo E.A.P.A Batista. 2013. DTW-D: Time Series Semi-Supervised Learning from a Single Example. In *Proceedings of the KDD 2013* (Chicago, Illinois, USA) *(KDD '13)*. Association for Computing Machinery, New York, NY, USA, 383–391. https://doi.org/10.1145/2487575.2487633

[5] Andrea Conley, Brendan Donohoe, and Benjamin Greene. 2021. Aftershock Identification Using a Paired Neural Network Applied to Constructed Data. (6 2021). https://doi.org/10.2172/1821802

[6] Haoyi Fan, Fengbin Zhang, Ruidong Wang, Xunhua Huang, and Zuoyong Li. 2021. Semi-Supervised Time Series Classification by Temporal Relation Prediction. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 3545–3549. https://doi.org/10.1109/ICASSP39728.2021.9413883

[7] Chelsea Finn, Pieter Abbeel, and Sergey Levine. 2017. Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks. In *Proceedings of the 34th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 70)*, Doina Precup and Yee Whye Teh (Eds.). PMLR, 1126–1135. https://proceedings.mlr.press/v70/finn17a.html

[8] Rafael Giusti, Diego F. Silva, and Gustavo E. A. P. A. Batista. 2016. Improved Time Series Classification with Representation Diversity and SVM. In *Proceedings of the ICMLA 2016*. 1–6. https://doi.org/10.1109/ICMLA.2016.0010

[9] Jacob Goldberger, Sam Roweis, Geoff Hinton, and Ruslan Salakhutdinov. 2004. Neighbourhood Components Analysis. In *International Conference on Neural Information Processing Systems* (Vancouver, British Columbia, Canada) *(NIPS'04)*. MIT Press, Cambridge, MA, USA, 513–520.

[10] Yunhui Guo, Noel C. Codella, Leonid Karlinsky, James V. Codella, John R. Smith, Kate Saenko, Tajana Rosing, and Rogerio Feris. 2020. A Broader Study of Cross-Domain Few-Shot Learning. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 124–141.

[11] Brijnesh Jain and Stephan Spiegel. 2016. Dimension Reduction in Dissimilarity Spaces for Time Series Classification. In *Advanced Analysis and Learning on Temporal Data*. Springer, Cham, 31–46.

[12] Shayan Jawed, Josif Grabocka, and Lars Schmidt-Thieme. 2020. Self-supervised learning for semi-supervised time series classification. In *Pacific-Asia Conference on KDD*. Springer, 499–511.

[13] Rohit J. Kate. 2016. Using dynamic time warping distances as features for improved time series classification. *Data Mining and Knowledge Discovery* 30 (3 2016), 283–312. Issue 2. https://doi.org/10.1007/s10618-015-0418-x

[14] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. 2020. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42 (2 2020), 318–327. Issue 2. https://doi.org/10.1109/TPAMI.2018.2858826

[15] Anthony Lomax, Claudio Satriano, and Maurizio Vassallo. 2012. Automatic picker developments and optimization. *Seismological Research Letters* 83, 3 (2012), 531–540.

[16] Kristóf Marussy and Krisztian Buza. 2013. SUCCESS: A New Approach for Semi-supervised Classification of Time-Series. In *Artificial Intelligence and Soft Computing*. Springer Berlin Heidelberg, Berlin, Heidelberg, 437–447.

[17] D.E. McNamara, W.L. Yeck, W.D. Barnhart, V. Schulte-Pelkum, E. Bergman, L.B. Adhikari, A. Dixit, S.E. Hough, H.M. Benz, and P.S. Earle. 2017. Source modeling of the 2015 Mw 7.8 Nepal (Gorkha) earthquake sequence. *Tectonophysics* 714-715 (2017), 21–30. https://doi.org/10.1016/j.tecto.2016.08.004 Special Issue on the 25 April 2015 Mw 7.8 Gorkha (Nepal) Earthquake.

[18] MexicoSSN. 2016. Mexico Earthquake catalog. Retrieved Jan 19, 2023 from http://www2.ssn.unam.mx:8080/catalogo

[19] Abdullah Mueen. 2016. Extracting Optimal Performance from Dynamic Time Warping. Retrieved Jan 19, 2023 from https://www.cs.unm.edu/~mueen/DTW.pdf

[20] Abdullah Mueen. 2016. Similarity Search on Time Series Data: Past, Present and Future. Retrieved Jan 19, 2023 from https://www.cs.unm.edu/~mueen/Tutorial/CIKM2016Tutorial.pdf

[21] Abdullah Mueen, Eamonn Keogh, and Neal Young. 2011. Logical-shapelets: an expressive primitive for time series classification. *Proceedings of the KDD 2011* (2011), 1154–1162. https://doi.org/10.1145/2020408.2020587

[22] Jyoti Narwariya, Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, and TV Vishnu. 2020. Meta-learning for few-shot time series classification. In *ACM IKDD CoDS*. 28–36.

[23] Minh Nhut Nguyen, Xiao-Li Li, and See-Kiong Ng. 2011. Positive Unlabeled Learning for Time Series Classification. In *International Joint Conference on Artificial Intelligence* (Barcelona, Catalonia, Spain) *(IJCAI'11)*. AAAI Press, 1421–1426.

[24] Alex Nichol, Joshua Achiam, and John Schulman. 2018. On First-Order Meta-Learning Algorithms. (3 2018).

[25] Brian Pope. 2022. Air Force Technological Advancement Center. Personal Communication. March 9, 2022 at 12:33 PM.

[26] Thanawin Rakthanmanon, Bilson Campana, Abdullah Mueen, Gustavo Batista, Brandon Westover, Qiang Zhu, Jesin Zakaria, and Eamonn Keogh. 2012. Searching and mining trillions of time series subsequences under dynamic time warping. In *Proceedings of the KDD 2012*. 262–270. https://doi.org/10.1145/2339530.2339576

[27] Christos Saragiotis. 2021. Efforts toward automatic aftershock sequences processing at the International Data Centre. In *EGU General Assembly Conference Abstracts (EGU General Assembly Conference Abstracts)*. Article EGU21-5062, EGU21-5062 pages. https://doi.org/10.5194/egusphere-egu21-5062

[28] Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical Networks for Few-Shot Learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 4080–4090.

[29] C. A. R. Sousa, V. M. A. Souza, and Gustavo Batista. 2014. Time series transductive classification on imbalanced data sets: an experimental study. In *International Conference on Pattern Recognition*. IEEE, 3780–3785.

[30] Vinicius M. A. Souza, Rafael G. Rossi, Gustavo Batista, and Solange O Rezende. 2017. Unsupervised active learning techniques for labeling training sets: an experimental evaluation on sequential data. *Intelligent Data Analysis* 21, 5 (2017), 1061–1095.

[31] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. 2018. Learning to compare: Relation network for few-shot learning. In *IEEE conference on computer vision and pattern recognition*. 1199–1208.

[32] Haishuai Wang, Qin Zhang, Jia Wu, Shirui Pan, and Yixin Chen. 2019. Time series feature learning with labeled and unlabeled data. *Pattern Recognition* 89 (2019), 55–66. https://doi.org/10.1016/j.patcog.2018.12.026

[33] Yaqing Wang, Quanming Yao, James T Kwok, and Lionel M Ni. 2020. Generalizing from a few examples: A survey on few-shot learning. *ACM computing surveys* 53, 3 (2020), 1–34.

[34] Li Wei and Eamonn Keogh. 2006. Semi-supervised time series classification. In *Proceedings of the KDD 2006*. ACM Press, New York, New York, USA, 748. https://doi.org/10.1145/1150402.1150498

[35] Liang Xi, Zichao Yun, Han Liu, Ruidong Wang, Xunhua Huang, and Haoyi Fan. 2022. Semi-supervised Time Series Classification Model with Self-supervised Learning. *Engineering Applications of Artificial Intelligence* 116 (2022), 105331. https://doi.org/10.1016/j.engappai.2022.105331

[36] Zhao Xu and Koichi Funaya. 2015. Time series analysis with graph-based semi-supervised learning. In *IEEE international conference on data science and advanced analytics*. IEEE, 1–6.

[37] Qi Zhang, Tong Xu, Hengshu Zhu, Lifu Zhang, Hui Xiong, Enhong Chen, and Qi Liu. 2019. Aftershock Detection with Multi-scale Description Based Neural Network. In *2019 IEEE International Conference on Data Mining (ICDM)*. 886–895. https://doi.org/10.1109/ICDM.2019.00099

[38] Sheng Zhong. 2022. Support Website. Retrieved Jan 19, 2023 from https://sites.google.com/view/onlinefewshot/home