Chapter 4

SRF Diagnosis with Augmentation and Early classification

4.1 Introduction

The data acquisition is one of the challenging tasks of RFD as the faulty data specific to SRF is not publicly available compared to bearing or gear fault data. Hence, the researchers depend on rotor testbeds to acquire SRF data, but it will be challenging to run the testbeds for a long time in structural fault environments. As a result, the testbed often generates insufficient and/or imbalanced datasets. Moreover, the commonly available testbeds generate unrealistic data because of the limitations in simulating the frequently changing RPMs, load, and other environmental conditions like noise, irregular sensor triggering, etc. The machinery most often runs under a healthy state in a real production process but rarely gives the data that effectively represent a fault. This causes lousy generalization performance when a system trained with such data, exposed to actual test conditions. This limits the scope of RFD-specific research improvements, which causes a proportional falling off in the percentage of AI works that utilized rotor fault-specific parameters in the literature.

This scenario opens a new research direction of data generation or augmentation

for generating a sustainable fault diagnosis model. Data augmentation is a process by which synthetic examples are created to enrich the dataset and enhance the parameter learning of the model. From the classifier's point of view, the data augmentation helps to reduce the variance of a classifier so that the classification error is minimized. The augmented data delves into the input space that is unexplored and thereby improves the generalization capability of the model, preventing overfitting. The methods to address unlabelled datasets or scarce labeled datasets always demand more research in small datasets situations.

Augmenting the dataset is familiar to the image and speech processing research community but not significantly explored in TS data processing. The commonly used augmentation techniques in image processing are cropping, warping, scaling, and rotating, and those used in or speech processing are acceleration, slowing down, etc. But these methods are not successful beyond a limit in augmenting the TS data, and hence the TS augmentations commonly depend on window slicing, permutations, and random sampling methods. However, these techniques have proven to be readily implementable but have specific issues. Firstly, there is no guarantee that the generated data follows the same distribution as the original data. Secondly, in no way, it ensures the TS properties (trend, seasonality, etc.) present in the newly synthesized data in most cases. It is also challenging to confirm the label of newly created samples in augmentation, opening a new direction to the research community. Hence, we propose an augmentation scheme that addresses the TS properties of original data and generates synthesized samples with very few original samples, even with non-uniform lengths. The augmentation is based on finding the consensus sequence from a set of sequences based on DTW distance matric. Specifically, a weighted soft-DTW-based data augmentation scheme is introduced and enhanced with fault information content (FIC), ensures the TS property, and solves the other practical data collection issues.

Moreover, from the the machine health monitoring perspective, making informed

decisions as early as possible without waiting for full-length TS data is highly desirable. In the literature, early class prediction of TS, based on partially observed data points is called an early classification (EC) [192, 193]. On the other hand, the traditional classification approach performs class prediction when full-length TS becomes available, which delay the decision and also increase the response time. In recent times, EC became an exciting research topic among researchers of various fields [194]. In some early attempts, Bregón et al. [195] presented an idea of classifying incomplete TS for fault classification in dynamic system. Xing et al. [192] proposed an instance-based EC approach by introducing the concept of the minimum required length and also addressed the trade-off between the earliness and the accuracy. Further, they introduced a feature-based early distinctive shapelet classification method, which produced highly interpretable results to the end-users [196]. Mori et. al. [193] proposed an EC method by learning the reliability threshold and also discriminating the classes over time. Similarly, the confidence threshold was defined by fusing the classifier's true prediction capability at successive time steps in [197]. An optimization-based EC approach on multivariate TS has been introduced by learning optimal decision rule in [198]. Besides, EC approach has been adopted in many useful applications in various domains such as early drought prediction [199], early disease prediction [200], early malware detection [201], and early transportation mode detection [202, 203]. EC approach is highly beneficial for the prediction of machinery faults, but it is still unexplored in SRF literature. Thus, an EC-based deep learning approach is introduced for SRF diagnosis in this work.

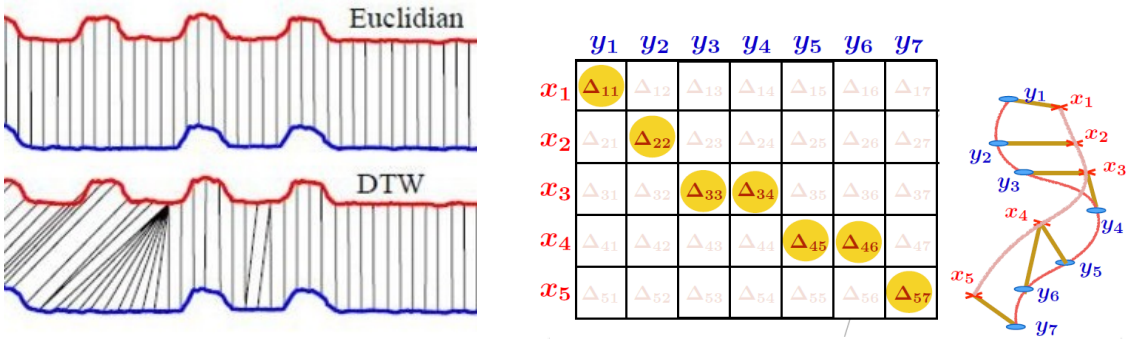
4.2 Soft-DTW Based Augmentation

Averaging is an essential data mining operation for finding the representative data item from a set of data items. When it comes to the computation of the average of a set of sequences, the order of elements in each sequence has to be considered, and hence it is not a trivial task. The complexity of sequence averaging depends on the distance

metric used, and experiments reveal that the commonly used Euclidean distance metric is not appropriate for the sequences. DTW [204, 205] is one preferred distance measuring technique for such cases, since it computes the best possible alignment between two sequences. Similarly, soft-DTW is a variation of DTW with the added differentiability property to the DTW calculation. The consensus sequence problem is related to finding the sequence representative, either the longest common subsequence, the medoid sequence, or the average sequence of the set of sequences. Since the longest common subsequence does not cover the whole data and provides only summary data, it is not preferred to the medoid or average sequence. The latter two methods select the sequence at the center of the sequences, where the average sequence finds the center as the object minimizing the sum of squared distances to objects of the set. But for medoid sequence, such a selection is made from the dataset itself. We use the average sequence definition that minimizes the sum of squared distances to sequences of a set, using DTW or soft-DTW as the distance metric.

4.2.1 Theoretical background

4.2.1.1 DTW



(a) DTW Vs Euclidian (b) DTW Alignment

Figure 4.1: DTW graphical representation

DTW has been using for many applications to find the flexible similarities between two different length sequences. The popular Euclidean distance cannot capture the flexible similarities even for the sequences representing similar trajectories as shown in Fig. 4.1(a). DTW is built on the Levenshtein distance or edit distance which finds the optimal coupling between two sequences by aligning their coordinates as shown in Fig. 4.1(b). Let $X = \langle x_1, x_2, ..., x_n \rangle$ and $Y = \langle y_1, y_2, ..., y_m \rangle$ be two sequences and d represents the pairwise distance between them, then the cost of finding the optimal alignment by recursive computation is given by:

$$\mathcal{D}(X_i, Y_j) = d(x_i, y_j) + \begin{Bmatrix} \mathcal{D}(X_{i-1}, Y_{j-1}) \\ \mathcal{D}(X_i, Y_{j-1}) \\ \mathcal{D}(X_{i-1}, Y_j) \end{Bmatrix} \tag{4.1}$$

where X_i and Y_j are the subsequences $\langle x_1, x_2, ..., x_i \rangle$ and $\langle y_1, y_2, ..., y_j \rangle$, respectively. This alignment is the optimal path matrix represented as $\mathcal{P} \in P_{n \times m}$, which is found by Eq. 4.2, and the algorithm to solve this problem results in exponential time complexity.

$$DTW_{(x,y)} = \min_{\mathcal{P} \in P_{n \times m}} \langle \mathcal{P}, \mathcal{D}(x, y) \rangle \tag{4.2}$$

where $\mathcal{D}(x, y) = [d(x_i, y_j)]_{ij} \in R^{n \times m}$ and $P_{n \times m}$ is the path matrix.

But since the recursive solution possesses the nature of overlapping sub problems, and as it allows for the memorization of partial results, it can be solved by dynamic programming that costs $|X| . |Y|$ operations with time and space complexity of $O(|X| . |Y|)$. The sequence of computations for DTW distance calculation is shown in Algorithm 4.1.

Figure 4.1 referred from www.dinhanhthi.com

Algorithm 4.1: DTW_Distance

Input: X, Y: sequences of length n and m, respectively.
Output: D: matrix of size $n \times m$

```
for i ← 0 to n do
    for j ← 0 to m do
        D(i, j) ← ∞
    end for
end for
D(0, 0) ← 0
for i ← 1 to n do
    for j ← 1 to m do
        cost = D(X[i], Y[j])
        D[i, j] = cost + min(D[i − 1, j], D[i, j − 1], D[i − 1, j − 1])
    end for
end for
return D
```

4.2.1.2 Soft-DTW

The application of DTW is limited due to its non-differentiability, as the *min* operator is not continuous, limiting gradient, or subgradient calculation. Soft-DTW [206] is a smoothed formulation of DTW, which considers all the alignment costs and finds their soft minimum. It is a differentiable loss function with quadratic time and space complexity to compute value and gradient. The end-to-end differentiable property of soft-DTW enables it to use in generative and predictive models. From the classification perspective, increased performance is observed with soft-DTW compared to over DTW as it can be used with kernel machines. Mit Shah et al. [207] used a soft-*min* operation for solving the non-differentiability issue of DTW. A differentiable generalized *min* operator with a smoothening parameter β is defined by the unified formulation of DTW (Eq. 4.2) and global alignment kernel (k_{GA}^{β}) [208] is expressed as:

$$\min^{\beta} \{p_1, \ldots, p_n\} = \begin{cases} \min_{i \leq n} p_i, \beta = 0 \\ -\beta \log \sum_{i=1}^{n} e^{-p_i/\beta}, \beta > 0 \end{cases} \tag{4.3}$$

The global alignment kernel is defined by:

$$\mathcal{K}^{\beta}(x,y) = \sum_{\mathcal{P}\in P_{n\times m}} e^{-\langle \mathcal{P},\mathcal{D}(x,y)\rangle/\beta} \tag{4.4}$$

Then, soft-DTW with β can be defined as:

$$dtw_{\beta}(x,y) = \min^{\beta}\{\langle \mathcal{P}, \mathcal{D}(x,y)\rangle, \mathcal{P}\in P_{n\times m}\} \tag{4.5}$$

where $\mathcal{D}(x,y)$ be the pairwise distance matrix and $\mathcal{P}\in P_{n\times m}$ be the path alignment matrix in the basic DTW calculation. When $\beta = 0$, the original DTW score is given and setting $\beta > 0$ recovers the dtw_{β} value.

The gradient of soft-DTW can be calculated by with the chain rule [206], and it is shown as:

$$\nabla_x dtw_{\beta}(x,y) = \left(\frac{\partial \mathcal{D}(x,y)}{\partial x}\right)^{T} \frac{\sum_{\mathcal{P}\in P_{n\times m}} e^{-\langle \mathcal{P},\mathcal{D}(x,y)\rangle/\beta}}{\mathcal{K}^{\beta}(x,y)}, \tag{4.6}$$

where $\frac{\sum_{\mathcal{P}\in P_{n\times m}} e^{-\langle \mathcal{P},\mathcal{D}(x,y)\rangle/\beta}}{\mathcal{K}^{\beta}(x,y)}$ is the average alignment matrix and it is denoted by $E_{\beta}[\mathcal{P}]$.

To find the differentiation of $dtw_{\beta}(x,y)$ algorithmically, first, the forward pass by Bellman's equation is performed. This stores the intermediary computations which results in $F = [f_{(i,j)}]$. Thereafter, the chain rule is used to find the impact of change in $f_{(i,j)}$ affecting the end result $f_{(m,n)}$ of the forward pass. In backward pass, the entire matrix $\mathcal{B} = [b_{(i,j)}]$ is calculated starting from $b_{(m,n)}$ down to $b_{1,1}$.

4.2.2 Proposed method

A soft-DTW based data augmentation scheme is adopted in this work. This scheme is based on DTW barycenter averaging (DBA) [209] algorithm, which uses an approximation approach to find the consensus sequence from a set of sequences. Forestier et al. [210], changed the objective function of DBA, incorporating a weight factor with each sequence to generate the average sequence. We have modified this scheme for augmenting TS signals in the subsampled space in the following ways:

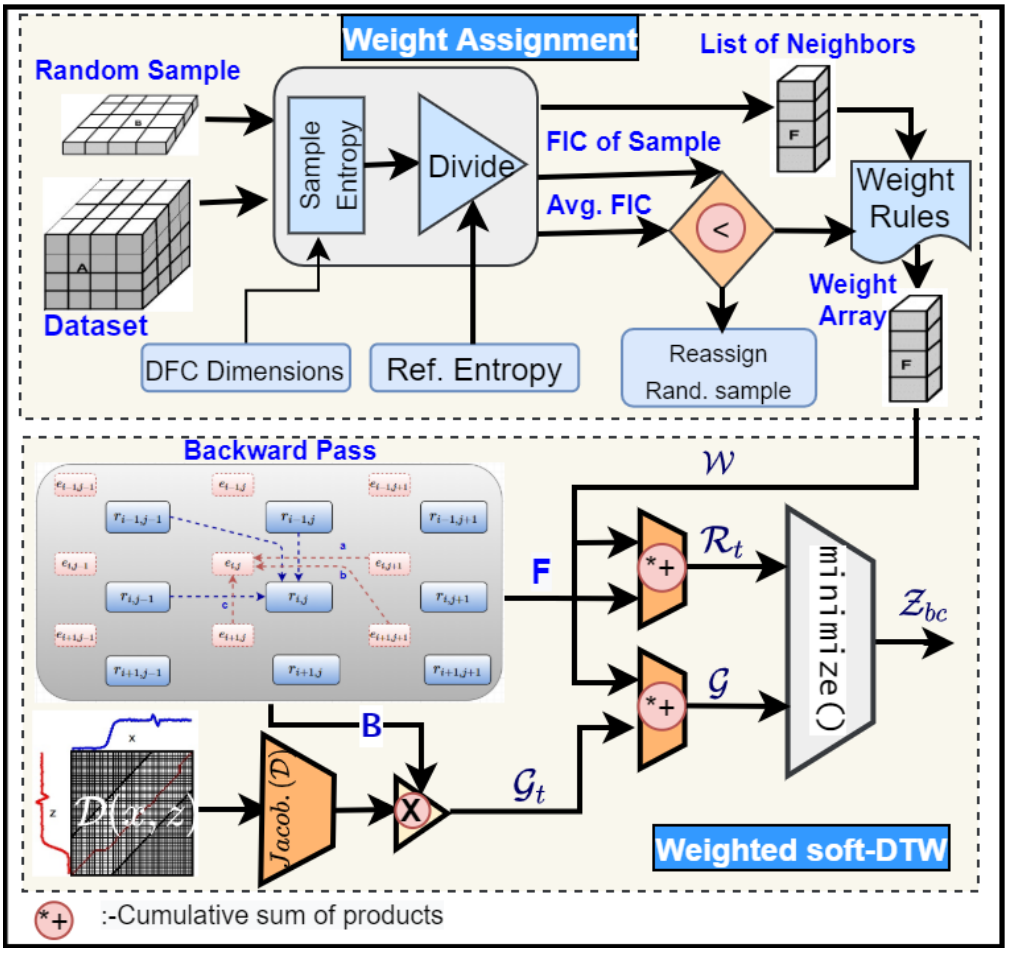


Figure 4.2: Soft-DTW barycenter approach

1. The proposed augmentation scheme incorporates the DFC of SRF in weight distribution, thus making it more domain-specific. It utilizes FIC to select the contributing samples for synthesizing more prudent samples, increasing the classification accuracy.
2. The soft-DTW barycenter approach is adopted to achieve smoother barycenters for effective learning [206].

This augmentation scheme preserves the sequential property and class discriminative ability of synthesized samples, as well as it deals with fewer reference samples and able to work with different length series. The Algorithm 4.2 along with Fig. 4.2, demonstrates soft-DTW barycenter generation. It has two main processes, the first one is the weight assignment process, and the second is the soft-DTW barycentre generation using these

weights.

The weight assignment process is based on the sample entropy [211]. The entropies of the DFC columns in a sample are averaged to find the representative entropy of a sample. The ratio of the representative entropies of faulty and healthy samples is then calculated to find the FIC of a particular sample. This process is given in Algorithm 4.2 from steps 1 to 9. Next, in order to assign weights, initially, a TS X_i has been randomly selected from the dataset of a particular fault. Then its FIC is compared with the average FIC of the whole dataset. If it is less, then a new random sample is selected, and the process is repeated. This process has been given in step 10 to step 13 in Algorithm 4.2 as well as in the first part of Fig. 4.2.

The weight distribution process is shown in step 15 of the algorithm. In the process, the random sample that meets the minimum FIC criteria is assigned a weight of 0.3. After that its $N \times 0.1$ (10.0% of total samples) nearest indexed neighbors are selected and ranked based on the sample FIC. The first two neighbors with the highest FIC are assigned a weight of 0.15 each, and then the next two neighbors are given the weight 0.1 each. To ensure the normalized sum of weights, the rest of the samples in the neighboring subset shares the remaining 0.2 weight equally to create the weight array. These weight values are identified to keep a balance between two objectives, i.e. diversity and the discriminative properties of the synthesized samples. Moreover, the approach of Fawaz et al. [212] is also considered with some modification to give the first four neighbors more weightage. The described scheme assists in selecting the most significant participating samples, and thereby a proportional reduction in computational cost has been achieved.

The weight assignment process follows the barycenter calculation by DBA with the help of DTW, which uses sequential computation. The forward and backward passes of soft-DTW are specified in line numbers 21 to 31 in Algorithm 4.2. The forward pass is required to facilitating the differentiation of DTW algorithmically. Line number 23 of

Algorithm 4.2: FIC based weighted soft-DTW

Input: X^f is a set of N^f TS of fault type f, and $z = X^f[k]$, where k is a random number $1 \leq k \leq N^f$, $z \in \mathbb{R}^{(L \times M)}$

Output: $\mathcal{Z}_{bc}$: optimal average sequence

Initialization :

Let $W^f \in \mathbb{R}^{N^f}$ is the weight vector initialized to 0, D_ϕ^{if} be the d^{if} DFC columns of i^{th} sample of X^f, ε^h be the entropy of reference sample, d be the distance fuction and set $S = 0$, $\mathcal{G} = 0$, $\mathcal{R}_t = 0$.

for $i \leftarrow 1$ to N^f **do**
  for $j \leftarrow 1$ to d^{if} **do**
    $S \leftarrow S +$ ***Sample_entropy***$(D_\phi^{if}[j])$
  end for
  $\varepsilon^{if} \leftarrow S/d^{if}$ /*Entropy of i^{th} sample of fault f
  $\varepsilon^f[i] \leftarrow \varepsilon^{if}/\varepsilon^h$. /*FIC assigned per sample
  $E[i, 0] \leftarrow \varepsilon^f[i]$
  $E[i, 1] \leftarrow i$
end for
$\varepsilon_{cls} \leftarrow \varepsilon^f/N^f$ /* Average FIC of fault class
if $(E[k, 0] < \varepsilon_{cls})$ **then**
  Find new random index k and reassign $X^f[k]$ to z.
end if
/*Sort sample indices based on the key FIC
$E \leftarrow Sort\,(E, key : E[:, 0])$
Distribute weights in W^f w.r.t index position k
$\mathcal{Z}_{bc} \leftarrow minimize(Fun\_sdtw())$
return $\mathcal{Z}_{bc}$

Procedure *Fun_sdtw()*

---

for $l \leftarrow 1$ to N^f and $W^f[l] \neq 0$ **do**
  $x \leftarrow X_l^f$
  Compute the cost matrix $\mathcal{D}\,(x, z) \in \mathbb{R}^{|x| \times |z|}$
  /*Forward pass to compute $F \leftarrow dtw_\beta(x, z)$
  for $i \leftarrow 1$ to $|x|$ **do**
    for $j \leftarrow 1$ to $|z|$ **do**
      $f_{i,j} \leftarrow d\,(x, z) + \min^\beta \{f_{i-1,j-1}, f_{i-1,j}, f_{i,j-1}\}$
    end for
  end for
  /*Backward pass to compute error B, assuming $b_{i,|x|+1} = b_{|z|+1,j} = 0$ and $b_{|x|+1,|z|+1} = 1$ for $i \in |x|, j \in |z|$
  for $i \leftarrow |x|$ to 1 **do**
    for $j \leftarrow |z|$ to 1 **do**
      $a_1 \leftarrow e^{\frac{1}{\beta}(f_{i+1,j} - f_{i,j} - d_{i+1,j})}$,
      $a_2 \leftarrow e^{\frac{1}{\beta}(f_{i,j+1} - f_{i,j} - d_{i,j+1})}$, and $a_3 \leftarrow e^{\frac{1}{\beta}(f_{i+1,j+1} - f_{i,j} - d_{i+1,j+1})}$
      $b_{i,j} \leftarrow b_{i+1,j}.a_1 + b_{i,j+1}.a_2 + b_{i+1,j+1}.a_3$
    end for
  end for
  $\mathcal{G}_t \leftarrow Jacobian(\mathcal{D}).B$
  $\mathcal{G} \leftarrow \mathcal{G} + W^f[l] * \mathcal{G}_t$
  $\mathcal{R}_t \leftarrow \mathcal{R}_t + W^f[l] * F$
end for
return $\mathcal{R}_t, \mathcal{G}$

the algorithm shows a forward pass of Bellman's equation and saves all the intermediate computations. Thereafter, the backward pass find the impact of change in $f_{(i,j)}$ affecting the end result $f_{(m,n)}$ considering the terms influencing $f_{(i,j)}$, i.e. $f_{i-1,j-1}$, $f_{i-1,j}$, and $f_{i,j-1}$. Then chain rule gives the backward recursion $b_{i,j} = \frac{\partial f_{m,n}}{\partial f_{i,j}}$, and the expansion of the same gives $b_{i+1,j}.\frac{\partial f_{i+1,j}}{\partial f_{i,j}} + b_{i,j+1}.\frac{\partial f_{i,j+1}}{\partial f_{i,j}} + b_{i+1,j+1}.\frac{\partial f_{i+1,j+1}}{\partial f_{i,j}}$. Here $\frac{\partial f_{i+1,j}}{\partial f_{i,j}}$ yields a ratio $e^{-f_{i,j}/\beta} / \left(e^{-f_{i,j-1}/\beta} + e^{-f_{i,j}/\beta} + e^{-f_{i+1,j-1}/\beta}\right)$. The logarithm of that derivative results in $\beta \log \frac{\partial f_{i+1,j}}{\partial f_{i,j}} = f_{i+1,j} - f_{i,j} - d_{i+1,j}$. Similarly we derive $\beta \log \frac{\partial f_{i,j+1}}{\partial f_{i,j}} = f_{i,j+1} - f_{i,j} - d_{i,j+1}$ and $\beta \log \frac{\partial f_{i+1,j+1}}{\partial f_{i,j}} = f_{i+1,j+1} - f_{i,j} - d_{i+1,j+1}$.

The backward recursion gives the entire matrix $B = [b_{i,j}]$, starting from $b_{n,m}$ down to $b_{1,1}$. The derivatives w.r.t. the values in the cost matrix $\mathcal{D}$ are computed by $\frac{\partial f_{n,m}}{\partial d_{i,j}} = \frac{\partial f_{n,m}}{\partial f_{i,j}} \frac{\partial f_{i,j}}{\partial d_{i,j}} = b_{i,j}.1 = b_{i,j}$ and hence we have

$$\nabla_x dtw_\beta(x, y) = \left(\frac{\partial \Delta(x, y)}{\partial_x}\right)^T B \tag{4.7}$$

Where B is the average alignment matrix shown in Eq. 4.6, which is also referred as $E_\beta[\mathcal{P}]$. The heterogeneity of the synthesized samples is ensured by the randomness of the samples and weights used in the weight assignment process. The vectorized operation of applying these weights to generate diversity in barycenters are given in line numbers 32 to 34 in the algorithm. These operations are shown in the second part of Fig.4.2. Finally, the minimization function returns the optimal soft-DTW barycenter.

4.3 Early Classification

The classification strategy which specifies how early a TS data can be labeled with the given supervised dataset is called early classification. It is more desirable in situations where decisions are time critical like clinical diagnosis, transportation, fault prediction, etc. Traditional TS classification algorithms assume that the full TS has been received prior to predict the class, however, the time-sensitive applications necessitates the pre-

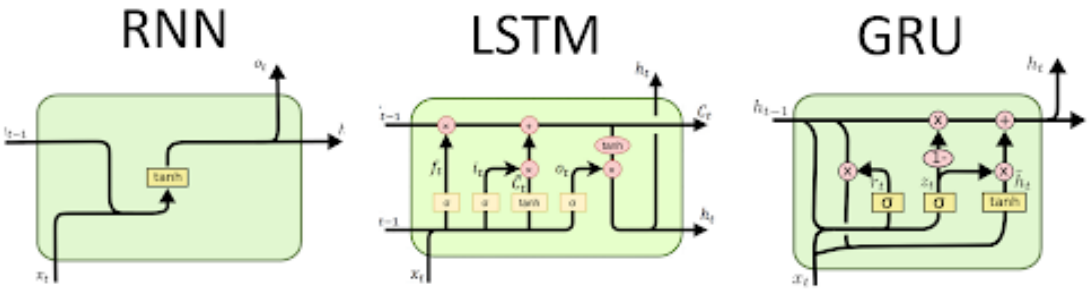


Figure 4.3: Sequential DL models

dictions to be made before the entire series has been observed. It has to sacrifice the accuracy of classification in the support of earlier predictions, accounting earliness sufficient to take an action. The main problem in an EC is to decide the time step or the number of data points with enough information to predict a class maintaining an expected level of accuracy. Mean time the accuracy and earliness are two contradictory objectives which forms a multi-objective optimization problems with task-dependent trade-offs. One of the main challenges in EC is that even though the full TS is labeled, it is not defined whether it can be labeled at an intermediate time step. i.e., an inherently unsupervised sub-task is there within a supervised task. Obviously, the other challenge is maintaining a decent trade-off between the two conflicting objectives; the accuracy and the earliness. The third challenge is regarding the multivariate TS where different variables will assume the class labels at different time steps. In fault diagnosis applications the early decisions are always welcomed as the diagnosis system can take care of the anticipated fault, thereby reduce the chance of unexpected shutdowns and losses. In the proposed SRF fault diagnosis model, at the fault classification phase, we used an EC approach for SRF that predicts the faults with an acceptable trade-off between earliness and accuracy. In this line, firstly, a sequential deep learning classifier is developed by considering accuracy only as an objective. Then, early decision policy is defined by taking accuracy and earliness into account.

4.3.1 Sequential DL models for early classification

In practice, most of the existing works gave the least importance to the temporal properties of TS data in SRF classification. They have been developed based on global features set, which seems to produce unrealistic solutions. The implementation of the early classification strategy on TS requires sequential DL models. The commonly used sequential DL models like simple RNN, LSTM [213], and GRU [214] have the sequential connections between their nodes. It makes them capable of learning temporal dynamic behavior for the input time sequence. The simple RNN replaces the whole activation, while LSTM and GRU regulate the information in each cell, which helps the latter models to tackle vanishing/exploding gradient problems. With this motivation, we proposed a deep learning architecture that consists of L number of recurrent layers for the implementation of EC. These layers capture the temporal information and there long term relationships from the signal. Thereafter, fully connected layer is added, which provides a higher level representation of data and also helpful in discriminating the classes well. Finally, the last layer performs the fault classification. We developed two EC models $\mathcal{M}_1$ and $\mathcal{M}_2$, respectively based on LSTM and GRU architectures.

4.3.2 Proposed method

In this work, the proposed early classification model (ECM) for fault diagnosis follows a two-fold process. In the first fold, the base classifier $\mathcal{F}$ is developed by taking accuracy only as the objective similar to the traditional classification approach in which class prediction is made once the complete TS becomes available. In the second fold, the class-wise confidence threshold is defined by taking the reliability of prediction and earliness into account.

Algorithm 4.3: ECM prediction process

Input: X: incoming TS, $\mathcal{F}$: trained classifier, $\{\theta_1, \theta_2 \dots \theta_k\}$: confidence thresholds
Output: $\hat{y}$: predicted class label, t^*: time point at which prediction is made

```
for t=1 to T do
    X_t ← paddingDataWithMean(X_t)
    ŷ ← F(X_t)
    compute θ̂ = ψ(ŷ),  using Eq. 4.9
    if θ̂ ≥ θ_ŷ then
        return ŷ and t* ← t
    end if
end for
```

4.3.2.1 Confidence threshold

The proposed ECM defines the class-wise confidence threshold (θ) to take the early decision on incoming TS. The confidence threshold is used to measure the reliability of class prediction that analyzes whether the observed sequence is sufficient enough for class prediction or not. Basically ECM process the incoming time-series X at every time step t and compute the confidence of predicted class $\hat{y} = \mathcal{F}(X_t)$, denoted by $\hat{\theta}_t$. Moreover, ECM predicts the class label only if computed $\hat{\theta}_t$ is higher than the predefined confidence threshold θ.

4.3.2.2 Training phase

In this phase, firstly, the base classifier $\mathcal{F} \in \{\mathcal{M}_1, \mathcal{M}_2\}$ is trained using full-length summary data. To learn the more accurate model, the training set is increased by augmentation with the proposed approach, discussed in section 4.2.2. The classifier $\mathcal{F}$ is trained with complete training set by considering accuracy only as objective. Next ECM learns the confidence thresholds $\{\theta_1, \theta_2 \dots \theta_k\}$ and follow a similar approach as defined in [197]. However, the proposed ECM learns the class-wise threshold, which is more adaptable for SRF diagnosis.

To learn the confidence threshold, we consider $\mathcal{F}$ as a pretrained-model, and compute the performance of classifier $\mathcal{F}$ at each timestep t, denoted by $\mathcal{F}_t^p$. Basically $\mathcal{F}_t^p$ measures the possibility of class prediction y while predicted class label is $\hat{y}$. It is

formally defined as:

$$\mathcal{F}_t^p(y, \hat{y}) = \frac{\parallel \{X^i | (y^i = y) \wedge (\mathcal{F}(X^i) = \hat{y})\} \parallel}{\parallel \{X^i | \mathcal{F}(X^i) = \hat{y}\} \parallel} \tag{4.8}$$

Thus, the confidence in prediction ($\hat{y}$) at single time point t is computed as $F_t^p(y = \hat{y}, \hat{y})$. To evaluate the $\mathcal{F}_t^p$, partial data $\mathcal{D}_t$ is used in which each $X_t \in \mathcal{D}_t$ contains only t data points. As the classifier $\mathcal{F}$ has been trained using full-length (T), X_t is not acceptable by $\mathcal{F}$. To deal with this problem, each X_t is padded with current mean in prefix manner. It fulfills the two requirements. Firstly, it makes the input acceptable to the pre-trained model. Secondly, It captures the mean distribution of current signals for unobserved series.

As the TS data is collected over time, we can get the class prediction at each time step t. Therefore, all the predictions up to t are utilized to compute the composite confidence for class prediction, which is defined as:

$$\psi(\hat{y}_t) = 1 - \prod_{j=1}^{t} (1 - \mathcal{F}_j^P(\hat{y}_t | \hat{y}_j)) \tag{4.9}$$

Finally, we define the optimal θ_c for each class of SRF that supports reliable class prediction early in time. In this process, all possible threshold candidates ($\theta_t^i, i \in [1, N], t \in [1, T]$) are computed for the training set. Next, the best class wise threshold is selected by balancing the trade-off between the reliability score of predictions and earliness.

4.3.2.3 Prediction phase

In the prediction process, the ECM processes the incoming X at each time point t and computes the class label $\hat{y}_t = \mathcal{F}(X_t)$. Further, it computes the confidence of $\hat{y}_t$ by Eq. 4.9. If incoming X at time t, satisfy the reliability threshold condition, ECM will predict the class label, otherwise ECM will wait to add more data points in the TS

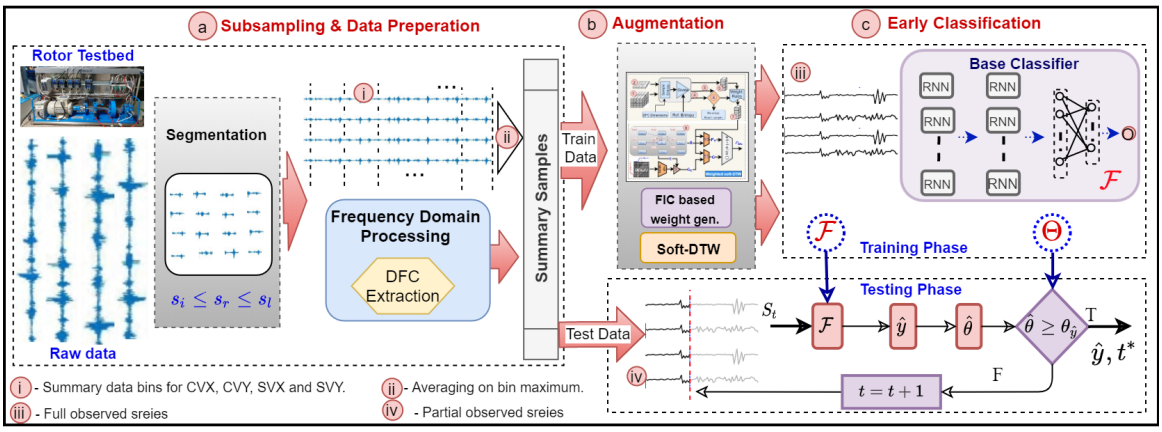


Figure 4.4: Overall framework

and repeats the process. The complete prediction process of ECM is demonstrated in Algorithm 4.3.

4.4 Overall Framework

The overall framework combining the proposed augmentation scheme (section 4.2.2) and early classification scheme (section 4.3.2) is shown in Fig. 4.4. The primary data processing step is identified as the subsampled data preparation phase that transforms the raw data into a TD, and DFC combined feature space without losing the sequential properties of data. This is shown in the first part of the figure indicated by 'a' in the pink circle. It is evident that the first three harmonics of the rotating frequency (1x, 2x, 3x), and its combinations with phase in both radial and axial directions is sufficient to distinguish between different SRF. Hence, this phase follows the DFC extraction process given in section 3.4 to provide DFC features. The summary data values in the TD are generated by finding the maximum values in every bin of size S_r data points and averaging such values from all the bins. It is shown as numbers 'i' and 'ii' in pink circles in the figure. Next, the augmentation process is employed to enhance the training data by generating more diverse samples, which is shown as 'b' in the pink circle. It is placed at the data input part of the ECM shown as the third part of the diagram ('c' in the

pink circle). This part consists of two phases called training ('iii' in the pink circle) and testing ('iv' in the pink circle). In the training phase, ECM learns the base classifier ($\mathcal{F}$) and class-wise reliability threshold $\Theta \in \mathcal{R}^k$. In the testing phase, ECM process the incoming signal in transformed feature space and predict the class label (fault type) when the corresponding reliability threshold is satisfied. A detailed description of each phase of the framework has been given in previous sections.

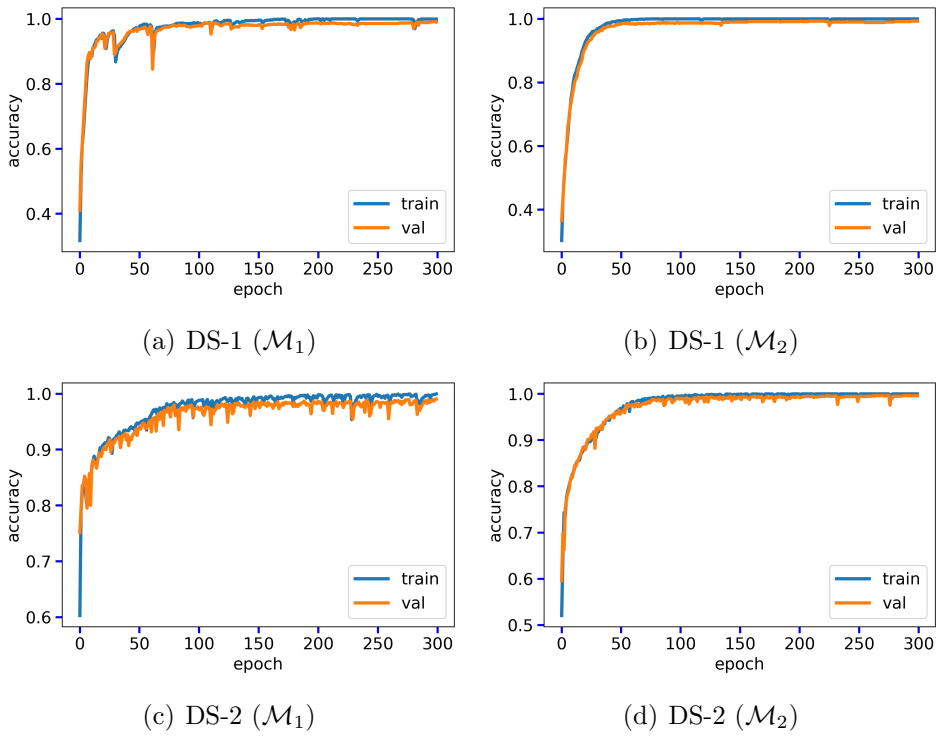
4.5 Results and Discussions

In this section, experimental results are presented and discussed. The original data collected from the rotor testbed, is summarized in the subsampled space at $1s$ duration. The first dataset contains six classes, with five different speeds and, each sample is organized to have 5 minutes duration. Then it is augmented to improve the size of the training data so that the original and the augmented data together form the overall training samples. The same subsampling, but with overlapping, is performed on the MaFaulDa dataset selecting the closest matching speeds of DS-1 with all load conditions.

The effectiveness of the proposed framework is evaluated to attain the earliness of fault classification by keeping a decent trade-off with accuracy. To mine the useful fault pattern information of SRF, the individual and the combined effect of TD and DFC features in the subsampled space with the sequential models are analyzed. Besides, the capacity of the framework in dealing with real plant data is examined with a separate pipeline denoted as PL-2. The performance of the model on PL-2 is compared with the standard data specified by PL-1. PL-2 dataset is designed to simulate unevenly sampled or missed data situation by randomly removing a few data points from subsampled space.

We determine the hyper parameters of the classification model $\mathcal{M}_1$ and $\mathcal{M}_2$ by considering the categorical cross-entropy as a loss function and Adam as a optimizer. Moreover, the proposed model uses *tanh*, *ReLU* and *softmax* as activation functions in

RNN, dense and output layers respectively. We have searched over recurrent layers $L \in \{1, 2, 3\}$, hidden nodes $HN \in \{16, 32, 64, 128\}$ and learning rate $\eta \in \{0.1, 0.01, 0.001\}$ for 300 epoch. Finally the best parameter $L = 2$, $\eta = 0.001$ are considered.



(a) DS-1 ($\mathcal{M}_1$) (b) DS-1 ($\mathcal{M}_2$)

(c) DS-2 ($\mathcal{M}_1$) (d) DS-2 ($\mathcal{M}_2$)

Figure 4.5: Accuracy v/s epoch graph for the model $\mathcal{M}_1$, $\mathcal{M}_2$

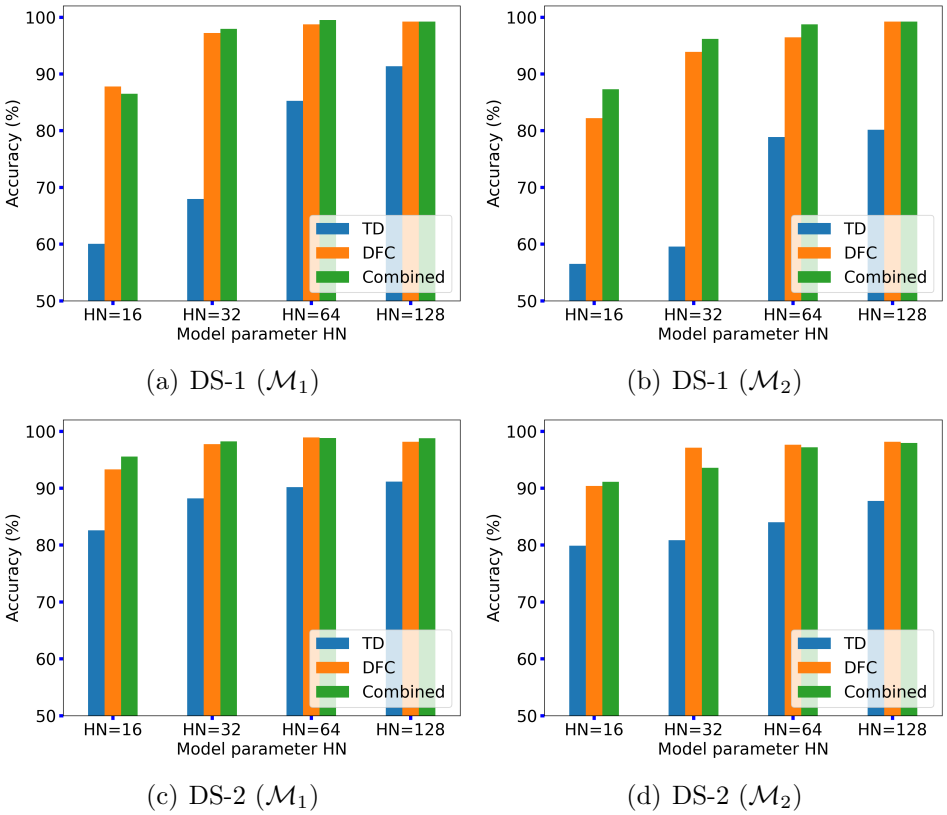
Table 4.1: Training performance

	$\mathcal{M}_1$ (Accuracy %)			$\mathcal{M}_2$ (Accuracy %)		
	Train	Val	Test	Train	Val	Test
DS-1	99.29	97.73	98.73	100.00	99.15	99.49
DS-2	99.82	99.76	97.96	99.93	99.88	98.78

4.5.1 Impact of subsampling and augmentation

Table 4.1 demonstrates the impact of representing the raw data in the sequentially subsampled feature space and the effect of augmentation on overall performance. Also,

Fig. 4.5 notifies the learning of the models on both the datasets. The accuracy results are provided for the combined features as a performance benchmark. Table 4.1 indicates that the $\mathcal{M}_1$ and $\mathcal{M}_2$ achieve satisfactory performance on both the datasets, and it specifies the ability of the subsampled feature space to boost the fault diagnosis results. Comparing the training and validation accuracy of both the models in both the datasets asserts the fact that the models are free from data overfitting, indicating the effectiveness of augmentation.



(a) DS-1 ($\mathcal{M}_1$) (b) DS-1 ($\mathcal{M}_2$)

(c) DS-2 ($\mathcal{M}_1$) (d) DS-2 ($\mathcal{M}_2$)

Figure 4.6: Model comparison with different nodes

Besides, model $\mathcal{M}_2$ achieves slightly superior performance in comparison to $\mathcal{M}_1$ on both the datasets. It is also observed that both the model able to achieve decent performance with few epochs for each dataset and the learning curve of $\mathcal{M}_2$ is comparatively smoother than $\mathcal{M}_1$. It is worth noting from the results that adding highly discriminative augmented data increases data diversity, which helps improve the parameter

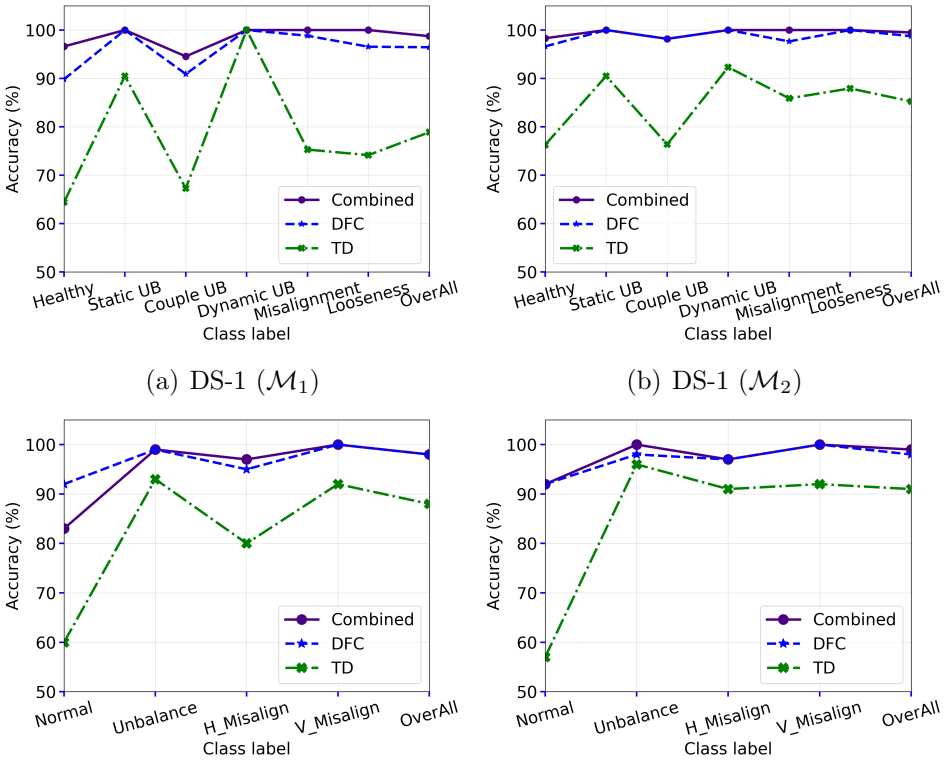
training of the model and shows an increase in accuracy with basic sequential learning models.

4.5.2 Performance analysis of sequential models

Fig. 4.6 shows the performance comparison of classifiers with TD, DFC, and combined features for both the datasets. As the model's structural complexity increases, a corresponding increase in accuracy is observed for both classifiers in both datasets with TD, DFC, and the combined features. It is observed that the combined features provide decent performance, and DFC itself is able to mark its importance in getting significant accuracy. In contrast, TD features proven to be inferior in performance for DS-1, while for DS-2, TD features are managed to give a minimum accuracy of 80.0%. The most important observation from the results of DS-1 is that both the sequential models with 64 HN is sufficient to give a decent accuracy closer to 100.0%. Interestingly, it shows that $\mathcal{M}_2$ model with 64 HN outperformed the $\mathcal{M}_2$ with 128 HN. Thus, it is concluded that the classifiers with 64 HN provide acceptable performance without compromising to higher complexity for negligible performance enhancement for DS-1.

Compared to DS-1, the DS-2 shows consistent improvement in accuracy as the model complexity increases. It is worth noticing that even the least complex model with 16 hidden layers with TD features could produce an accuracy of around 80.0% with both the classifiers. As in DS-1, the supremacy of $\mathcal{M}_2$ over $\mathcal{M}_1$, and dominance combined features over TD or FD features persists with DS-2 also. Regarding the model selection, there is a slight performance improvement for 64 HN model compared 32 HN model for both the classifiers, where 128 HN model could give negligible performance enhancement for $\mathcal{M}_1$ only. Thus, we have chosen the model with 128 HN for comparative analysis. Fig. 4.7 shows the importance of DFC in SRF diagnosis for both the classifiers on the datasets. The fault-wise analysis assists in understanding the significance of DFC in fault diagnosis. The noteworthy fluctuations have been observed for TD features with

different types of faults. Both classifiers have displayed the unacceptable performance for the healthy and couple UB class labels for DS-1. Also, similar observation has been noted for misalignment as well as looseness class using $\mathcal{M}_1$ model. But DFC, as well as the combined features, provide acceptable performance irrespective of the fault type for both the models. Also, $\mathcal{M}_2$ performs slightly better than $\mathcal{M}_1$ in terms of accuracy. It is observed that the class-wise behavior of DFC and combined features in DS-2 is almost similar to that of DS-1, producing no significant fluctuations between different fault types. But TD features show unacceptable performance with the class normal in both the classifiers; at the same time $\mathcal{M}_2$ is able to make up the performance degradation of horizontal misalignment class. However, as with DS-1, $\mathcal{M}_2$ performs better than $\mathcal{M}_1$ for all the classes.



(a) DS-1 ($\mathcal{M}_1$) (b) DS-1 ($\mathcal{M}_2$)

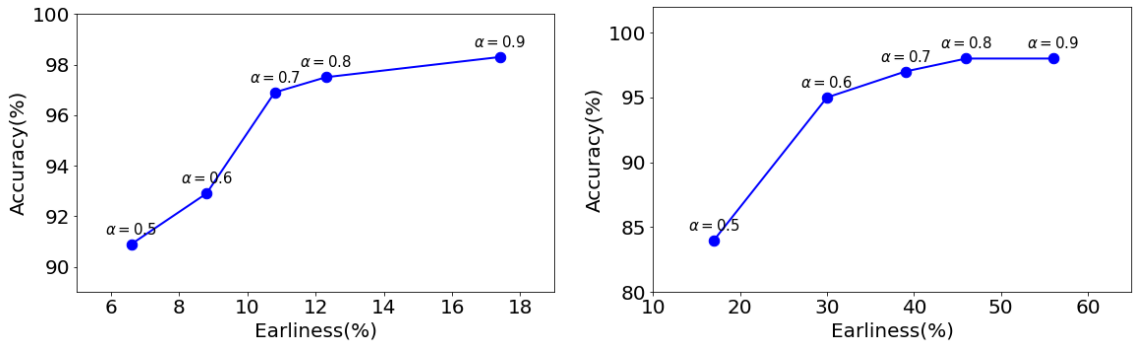
(c) DS-2 ($\mathcal{M}_1$) (d) DS-2 ($\mathcal{M}_2$)

Figure 4.7: Comparative analysis of TD, DFC, and combined features.

4.5.3 Performance analysis of ECM

4.5.3.1 Impact of α parameters in ECM

In the proposed model, parameter α decides the trade-off between accuracy and earliness and holds a value between 0 and 1. Fig. 4.8 demonstrates the trend of $\alpha \in \{0.5, 0.6, 0.7, 0.8, 0.9\}$, and it is observed that the accuracy and earliness substantially increase with the increase in the value of α. Moreover, it is also realized that the trend of α depends on the characteristic of the dataset. For example, change in α from 0.6 to 0.7, accuracy improved by 4.0% and 2.0% for datasets DS-1 and DS-2 respectively, whereas the earliness increased by 2.0% and 9.0% respectively for corresponding datasets. In this experimental work, the value of α is considered to be 0.9, as it exhibits adequate performance in terms of accuracy and earliness for both the datasets. However, any value of α can be selected as per the requirement of the system.



(a) DS-1 (b) DS-2

Figure 4.8: Effect of α

4.5.3.2 Performance of ECM on two datasets

The performance analysis of the proposed ECM on datasets DS-1 and DS-2 are provided in Table 4.2 and Table 4.3, respectively. Along with that, the performance of the PL-2 dataset has also been compared. In Table 4.2, it is observed that the ECM-$\mathcal{M}_2$ achieves the highest accuracy of 99.50% for combined features, whereas ECM-$\mathcal{M}_1$

with DFC features achieves the best earliness of 12.70%. As the TD signals have the least discriminative ability, it requires longer series to get an acceptable accuracy, while DFC and combined signals can make quick conclusions with less than 20.0% earliness without compromising accuracy. Similarly, model $\mathcal{M}_2$ is capable of capturing information in a short length of the series as compared to model $\mathcal{M}_1$. As a result, ECM-$\mathcal{M}_2$ outperformed the ECM-$\mathcal{M}_1$ in both the objectives i.e. accuracy and earliness. A similar conclusion has been drawn for DS-2 also, as shown in Table 4.3. But due to more diversity in speed and load conditions, DS-2 dataset demands more length sequences to make confident classification. Hence, the earliness value of DS-2 is around 60.0%. But still, the overall accuracy of DS-2 is slightly better than DS-1 in this framework.

Table 4.2: Performance of ECM on dataset DS-1

		TD		DFC		Combined	
		Acc	Ear	Acc	Ear	Acc	Ear
PL-1	ECM-$\mathcal{M}_1$	78.40	56.80	96.40	12.70	98.20	20.20
	ECM-$\mathcal{M}_2$	86.30	80.70	98.50	12.80	99.50	14.90
PL-2	ECM-$\mathcal{M}_1$	77.30	60.02	93.30	15.60	96.23	24.34
	ECM-$\mathcal{M}_2$	85.40	82.80	96.01	16.08	97.91	18.19

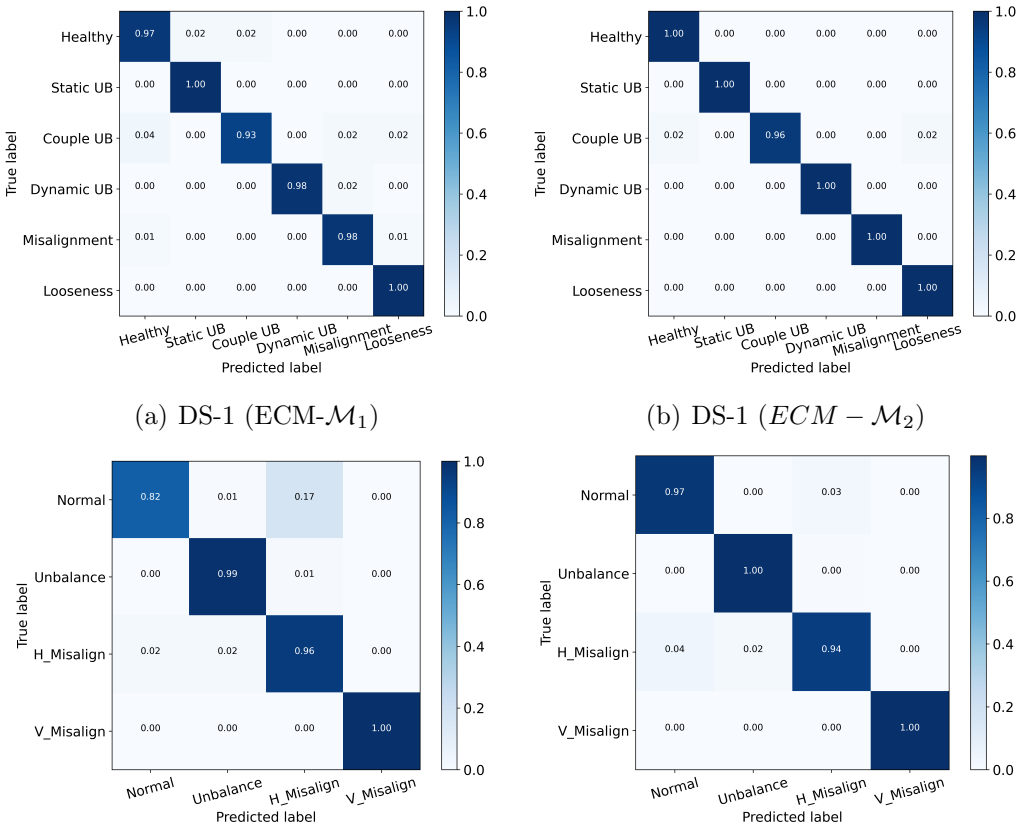
Acc: Accuracy (%), Ear: Earliness (%).

Table 4.3: Performance of ECM on dataset DS-2.

		TD		DFC		Combined	
		Acc	Ear	Acc	Ear	Acc	Ear
PL-1	ECM-$\mathcal{M}_1$	88.53	86.24	97.01	63.07	97.55	62.66
	ECM-$\mathcal{M}_2$	90.29	90.83	97.51	54.21	98.32	55.68
PL-2	ECM-$\mathcal{M}_1$	88.07	86.17	94.65	63.87	95.60	63.17
	ECM-$\mathcal{M}_2$	88.93	90.98	93.06	54.79	96.96	56.66

Acc: Accuracy (%), Ear: Earliness (%) .

Considering the performance of the model with PL-2 data, it is evident from the first observation that there is a trend in decreased accuracy and increased earliness value as compared to PL-1, in both the datasets. Since data acquisition irregularity is created by randomly removing the samples, and the data missed timestamps are filled



(a) DS-1 (ECM-$\mathcal{M}_1$)

(b) DS-1 ($ECM - \mathcal{M}_2$)

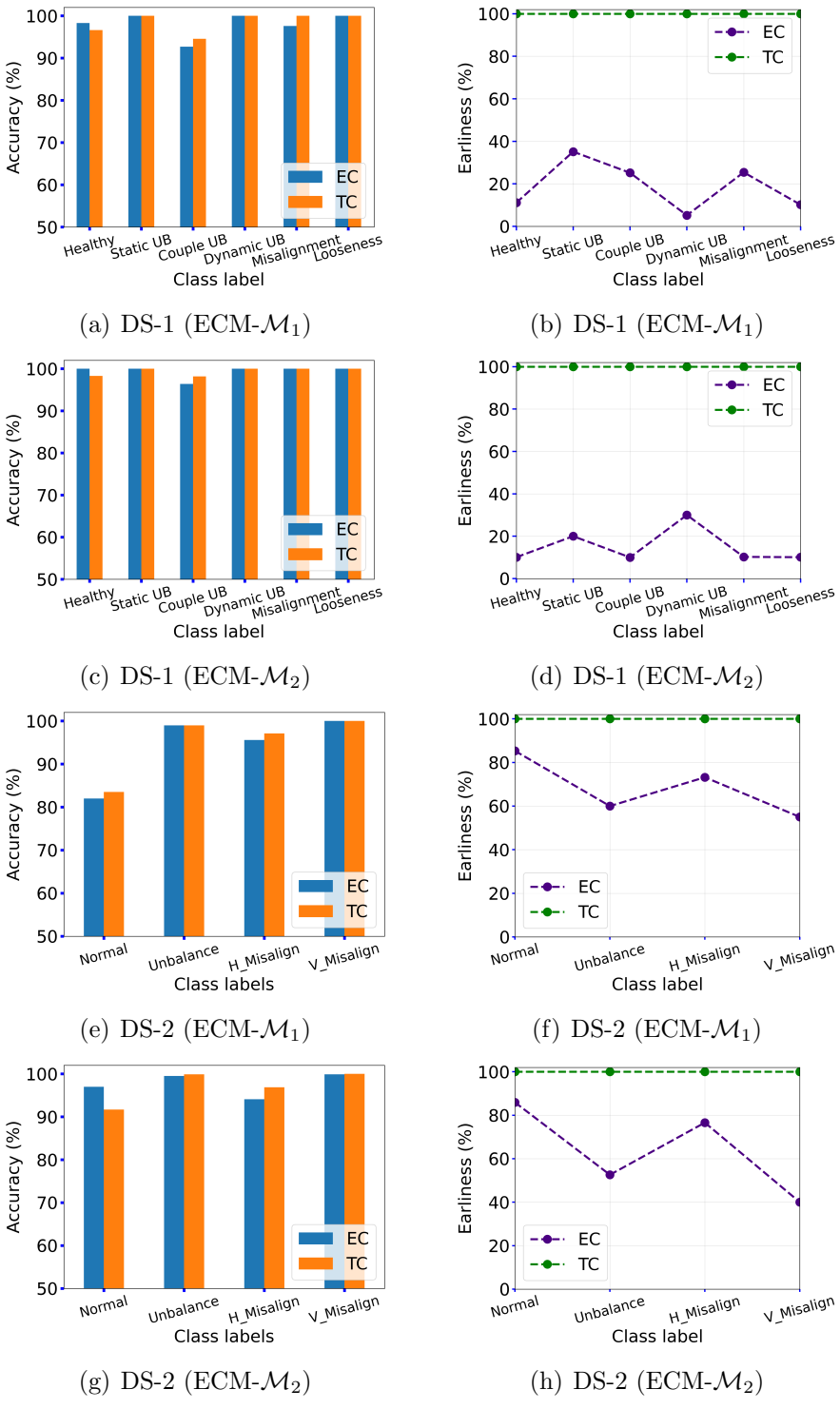
(c) DS-2 ($ECM - \mathcal{M}_1$)

(d) DS-2 ($ECM - \mathcal{M}_2$)

Figure 4.9: Confusion matrix of ECM-$\mathcal{M}_1$ & ECM-$\mathcal{M}_2$ on DS-1 & DS-2 datasets

with mean values, in TD data, this trend is not significant. That is, the TD data points in the subsampled space are almost similar to PL-1 data, because of the averaging on maximum bin operation. But the DFC extracted values in the subsampled space of PL-2 are bit different from the DFC data sequence generated in PL-1. This justifies the reduction in accuracy and need for more lengthy pattern (increased value of earliness) in both DFC and combined feature scenarios. But it is worth noticing that the overall model accuracy and earliness of PL-2 can be maintained almost similar to PL-1, because of the training with the proposed augmentation method.

Further, the confusion matrix has been shown in Fig. 4.9 to analyze the performance of ECM for classifying the healthy and faulty components using combined features. The proposed ECM demonstrates the good performance for healthy as well as for faulty



(a) DS-1 (ECM-$\mathcal{M}_1$)

(b) DS-1 (ECM-$\mathcal{M}_1$)

(c) DS-1 (ECM-$\mathcal{M}_2$)

(d) DS-1 (ECM-$\mathcal{M}_2$)

(e) DS-2 (ECM-$\mathcal{M}_1$)

(f) DS-2 (ECM-$\mathcal{M}_1$)

(g) DS-2 (ECM-$\mathcal{M}_2$)

(h) DS-2 (ECM-$\mathcal{M}_2$)

Figure 4.10: Comparative analysis of proposed model with traditional approach

components except for Couple UB as shown in Fig. 4.9(a)-(b). This effect has been observed because the properties of static UB and dynamic UB are almost similar and it creates adversity in classification. For DS-2 dataset also, ECM demonstrated good performance and able to classify unbalance and vertical misalignment accurately as shown in Fig. 4.9 (c)-(d). ECM-$\mathcal{M}_1$ misclassified 17.0% normal samples as horizontal misalignment. However, ECM-$\mathcal{M}_2$ outperformed with 97% accuracy and misclassified only 3.0% normal sample as horizontal misalignment. Overall, the proposed ECM is capable of providing exceptional results for SRF classification.

4.5.3.3 Comparative analysis of ECM with traditional approach

We have compared the proposed ECM with the traditional classification (TC) approach for SRF, which is depicted in Fig. 4.10. The TC approach considers the full-length sequence of data for classification, whereas the proposed ECM is able to predict the fault with partial sequence of data. ECM-$\mathcal{M}_1$ classified dynamic UB fault with 100.0% accuracy by utilizing only 5.19% data points, as shown in Fig. 4.10(a) and Fig. 4.10(b). Moreover, ECM-$\mathcal{M}_1$ classified couple UB using 25.19% of full-length data only as compared to the TC approach with approximately 1.8% deficiency in the accuracy. It is also observed that the proposed ECM achieved better accuracy than the TC approach for a healthy condition, as shown in Fig. 4.10(a) and Fig. 4.10(c). Moreover, ECM-$\mathcal{M}_2$ achieved better earliness compared to ECM-$\mathcal{M}_1$ except for dynamic UB class. For dataset DS-2, ECM-$\mathcal{M}_2$ demonstrated similar or even higher accuracy for normal class, compared to DS-1, by utilizing 85.9% data points as shown in Fig. 4.10(g)-(h). Both models utilized more lengthy sequences with DS-2 compared to DS-1 due to the diverse nature of the dataset. It is also observed that the accuracy and earliness patterns vary for both the classifiers $\mathcal{M}_1$ and $\mathcal{M}_2$ with respect to the faults under consideration. However, the earliness and the accuracy of $\mathcal{M}_2$ is always found superior to that of $\mathcal{M}_1$. Thus, based on the above observation, it is concluded that the proposed EC approach

for SRF is highly efficient as compared to the traditional one.

4.6 Summary

The significance of providing sufficient data to the DL models is well-known for the research community, and hence a variety of data augmentation schemes have been proposed. But augmenting data in the TS domain still left not addressed to its importance, which affected the research progress in the FDPM area. This work presented a domain-specific data augmentation scheme for SRF diagnosis with an advanced EC learning strategy. Thus, this work addresses challenging issues like inadequate and unrealistic faulty data in SRF and the limitations in employing DL and advanced learning strategies. The data scarcity and imbalance problems are handled through an augmentation method using soft-DTW, enhanced by fault information content-based weighing scheme. At the fault classification phase, we proposed an EC approach for SRF that predicts the faults with an acceptable trade-off between earliness and accuracy. Popular sequential learning methods such as LSTM and GRU have been utilized to make a decision with the partially observed data. Thus the early class prediction showed its significance in fault diagnosis of SRF by making fact-based decisions at the earliest possible without having to wait for full-length. The proposed data subsampling method, which incorporated SRF specific DFC and TD features, facilitated the model to perform well irrespective of the industrial data acquisition issues. The soft-DTW-based augmentation enriched the subsampled input training dataset and eliminated the class imbalance issue.

The experiments showed the impact of subsampling and augmentation in SRF diagnosis by providing a decent performance with few epochs for both Meggitt and MaFaulDa datasets. It is evident from the results that adding highly discriminative augmented data increases data diversity, which helps improve the parameter training of the model and shows an increase in accuracy with basic sequential learning mod-

els. Moreover, It is observed that the combined features provide decent performance, and DFC itself is able to mark its importance in getting significant accuracy. Interestingly, it shows simple models with 64 hidden nodes provided acceptable performance, and there is no need to compromise with higher complexity for negligible performance enhancement with the proposed framework. Moreover, it is noteworthy that the TD features show fluctuations in accuracy with different types of faults, but DFC has given consistent performance in fault-wise accuracy comparison, which signifies the impact of normalization in DFC extraction. It is observed that the models achieved the highest accuracy of 99.50% for combined features with an earliness of around 15.0% value. The best earliness of 12.70% produced an accuracy of 96.4%, which shows that the model's tradeoff between accuracy and earliness is well-maintained. DFC and combined signals can make quick conclusions with less than 20.0% earliness without compromising accuracy.

The proposed augmentation scheme preserves the same sequential property of the original signal, due to which the augmented data enhanced the EC training process. Even with the most irregular data (PL-2) created for simulating actual industrial conditions, the model could provide a decent result. The paradigm's advantage is that it offers decent performance early with a minimum input sequence. As a result, it provides a significant amount of time for maintenance activity than the traditional fault diagnosis method. The decision policy is the heart of the early classification paradigm and plays a crucial role in providing reliable performance by controlling the earliness parameter. Hence, the experimental results demonstrated that the proposed framework achieved decent outcomes in both objectives: accuracy and earliness.