Solar Pre-Flare Classification with Time Series Profiling

Ruizhe Ma*, Azim Ahmadzadeh*, Soukaïna Filali Boubrahimi*,
Manolis K. Georgoulis†‡, and Rafal A. Angryk*
*Department of Compute Science, Georgia State University, Atlanta, GA, USA
†Department of Physics & Astronomy, Georgia State University, Atlanta, GA, USA
‡RCAAM of the Academy of Athens, Athens, Greece
Email: *{rma1,aahmadzadeh1,sfilaliboubrahimi1}@student.gsu.edu, rangryk@cs.gsu.edu
†manolis.georgoulis@phy-astr.gsu.edu

***Abstract*—Space weather encapsulates the impact of variable solar activity on the vicinity of Earth and elsewhere in the solar system. A major agent of space weather, with significant effort already devoted to its prediction, is solar flares. Most existing analysis in this direction focus on the instantaneous (point-in-time) magnitude of various pre-flare parameters in flare host locations, solar active regions. Nonetheless, a recent trend places data-intensive studies, focusing on the pre-flare time series of these parameters, to the forefront. We take on this task in this study, focusing on the shape of pre-flare active region parameter time series by introducing a data-driven class profiling and clustering of these time series. We rely on data provided by the Space Weather ANalytics for Solar Flares (SWAN-SF) benchmark dataset. Our results indicate some potentially interesting temporal patterns that are unrelated to parameter magnitudes and may be used, both in tandem and independently from magnitudes, for future flare forecasting efforts. Our analysis also provides flexibility to define custom flare classes relying on pre-flare time series behavior and relate them to the existing, conventional NOAA / GOES flare classes.**

***Index Terms*—solar flare, time series, event profile**

I. Introduction

In recent years, solar flare prediction has attracted interdisciplinary researchers with a passion to forecast singular, rare natural events. The incentive for detailed, accurate solar flare forecasts lies in the severity and impact of the phenomenon on humans and infrastructure beyond Earth's protective atmosphere. Solar flares are known as sudden enhancements of high-frequency electromagnetic radiation from the Sun, including extreme ultraviolet, X- and even γ-rays for the largest events (see, for example, [1] and references therein). They are purely magnetic phenomena and inherently local, originating from intense accumulations of magnetic flux in the Sun's lower atmosphere, known as active regions (e.g., [2]). If associated with coronal mass ejections (CMEs) [3], which are expulsions of plasma into the interplanetary space, they are called eruptive flares. The solar weather puzzle is complete with solar energetic particles (SEPs) that relate to both flares and CMEs and are accelerated to speeds comparable to the speed of light.

Instantaneous impact of intense flares and SEPs on humans pertains to astronauts engaging in extravehicular activities (e.g., spacewalks, future Moon landings, etc.). Strong CMEs, if directed towards Earth, may cause magnetic storms that disrupt Earth's magnetosphere for days. For extreme space weather events, the impact on infrastructure, both on Earth's surface and in orbit, could be devastating (see the updated 2019 National Space Weather Strategy and Action Plan), ranging from large-scale power grid failures, to degradation of radio and satellite communications, to Global Positioning System (GPS)/Galileo outages and failures, as well as enhanced radiation levels for passengers and crew flying close to the poles. Details on the projected financial impact of extreme space weather can be found in [4]–[6].

A number of physics-based models exist for predicting solar flares. However, while the flare event data are recorded as time series, due to the high complexity and the related costs, they are more commonly used as generalized point-in-time values, meaning single value per parameter per solar flare event. In this paper, we approach the problem from a more data-driven perspective. Our novelty comes from the use of time series analysis. Through the unsupervised learning process of clustering, we hope to identify natural patterns within solar flares and generate viable pre-flare profiles.

A key application of pre-flare profiles could be toward flare prediction. If we can identify distinct trends profiles of different parameters prior to the occurrence of different flare classes, we would be able to make predictions as the measurements come in, in near realtime. Another possible application of pre-flare profiles is identifying possible sub-classes within existing flare classes. If we were to find different pre-flare profiles within existing flare class labels, this would insinuate the existence of physical sub-classes within the current flare class definitions. Both applications above may be hard to achieve using point-in-time parameter values, and the adoption of time series data and shape-based analysis could set the stage in this direction. The main goal is to identify the possibility of using shapes and not values to establish flares.

The rest of this paper is organized as follows: Section II introduces the background information. Section III discusses the commonly used time series normalization methods. Section IV presents the solar flare time series data used in our experiments. Section V shows the pre-flare profile results. Finally, Section VI presents our conclusion.

978-1-7281-0858-2/19/$31.00 ©2019 IEEE

II. Background

A. Solar Flare Forecasting

The importance of flares themselves, and also their role in the prediction of CMEs and SEPs resulted in literally hundreds of studies, ranging from standardizing practices for appropriate input data collection to benchmarking and performance verification metrics (see, for example, [7]–[9] and references therein). Prediction methods *per se* can be grouped in numerous different categories (see, for example, [10], [11] with recent important additions being machine and deep learning techniques of increasing sophistication [12]–[20]. A quick perusal of these studies reveals both the complexity of the problem at hand and how far we have come toward a better understanding of its challenges, which up until recently has relied exclusively within the realm of solar physics. Building upon all this work, new studies constantly discover new avenues for tackling the problem, albeit still with outcomes far from perfect.

In the direction of benchmarking, ongoing work by [21], [22], a benchmark dataset was released that allows a seamless study of flare occurrence relying on the pre-flare time series of numerous solar active region parameters. This benchmark dataset has been coined Space Weather ANalytics for Solar Flares (SWAN-SF) and provides times series of physical parameters ranging from the appearance of active regions in the eastern solar limb all the way to their rotation beyond the western limb (an approximate total of 14 days). With the severe projection effects close to the limbs, there are two options to follow: either use the values that correspond to CEA magnetic fields all the way to the limbs in an effort to implement full-disk prediction or implement thresholds in the central meridian distance to ignore values at the limbs. Here, we choose the first option, taking into account all available metadata. This said, as each point in the time series of a SHARP parameter is associated with a location of the HARP box in the solar disk, future works may well follow the second option.

The active regions correspond to the current solar cycle and are captured via HMI Active Region Patches (HARPs) enhanced for space weather analysis (Space Weather HARPs, or SHARPS [23]). The HARP/SHARP data product has a 12-minute cadence and stems from the Helioseismic and Magnetic Imager (HMI) [24] telescope onboard the Solar Dynamics Observatory. By focusing entirely on time series analysis, the SWAN-SF is virtually a unique benchmark dataset, given that an overwhelming majority of flare forecasts rely on instantaneous, point-in-time values of predictive parameters. Given that flares are an inherently nonlinear dynamical phenomenon, with reasonably well-defined pre-flare and post-flare phases, time series analysis enables us to look at flare prediction from a new perspective. It is imperative to see whether this perspective can put some of the long-standing questions on flare occurrence and triggering to rest.

Flare forecasting using the SWAN-SF dataset is technically a classification (i.e., a supervised) problem. It is, therefore, important to also understand and utilize the automated labeling process of flares. Instances of this type are automatically detected and classified by the National Oceanic and Atmospheric Administration's (NOAA) constellation of GOES satellites. This is achieved by dividing the logarithmic domain of flares' peak flux in soft X-ray wavelengths, into five sections, or *classes*. From the weakest to strongest, these flare classes are A, B, C, M, and X. Each of these categories is divided into a logarithmic scale from 1 to 9. The X-class category is slightly different and does not stop at 9. Those flares are sometimes called Super X-class flares. For example, a C2.0-class flare is, in terms of peak X-ray flux, two times stronger than a C1.0-class flare, 20 times stronger than a B1.0-class flare and ten times weaker than an M2.0-class flare. Subsequently, each flare class can be further divided into ten smaller subcategories, and allow labels such as C1.2, which is two times of a B1.0-class flare stronger than C1.0.

The particular mapping between the flares' peak flux and categorical labels is done for convenience; otherwise, the threshold on the peak flux that separates, for instance, C-class flares from B- or X-class flares, could be anywhere else, as long as the order is preserved. Nonetheless, all of the existing categorical flare forecasting models, measure their models' performance by relying on the correct prediction of those synthetic labels. In this study, using the power of clustering algorithms on the historical time series of flares, provided by SWAN-SF, we aim to find clues to a more data-driven set of flare classes. While this would not change the complexity of this forecasting problem, it may allow shedding light on some of the challenging parts of the problem.

B. Distance Measure

Time series corresponds to a widely used sequential data format, desirable in cases where analysis of real-valued, continuous data are important, and potentially more meaningful than independent parameter values. Considering the descriptive nature of time series data, it is natural for small discrepancies to occur in time series describing the same class of events.

When working with time series, it is important to appreciate the difference between lock-step and elastic distance measures. Given time series Q and C, a lock-step distance measure pertains to the Minkowski distance raised to the power of p, also known as the L_p norm. Typically, Minkowski distance is used with $p = 1$ or 2, which corresponds to the Manhattan distance or the Euclidean distance, respectively. Lock-step measures enforce triangle inequality and imply that the i-th element in one sequence is always mapped to the i-th element in the compared sequence. Euclidean distance is one of the most popular and commonly applied distance measures: it is the straight-line distance between two points in Euclidean space. When Euclidean distance is applied towards time series data, usually equal length is required, else, pre-processing needs to take place before applying Euclidean distance. Therefore, Euclidean distance is best applied to point-in-time, rather than sequential distances.

Conversely, for elastic measures, a one-to-one mapping is only one possibility; one-to-many mappings are also allowed [25]. A widely applied elastic distance measure, Dynamic Time Warping (DTW), is used for measuring the similarity between two time series, not necessarily of equal length. Originally, DTW was used in speech recognition; later, it was adapted to various real-world data mining problems. Generally, this is a method that enables computers to find an optimal match between two given sequences under certain constraints. DTW has been accepted as an efficient measure for time series data [26]–[29]. Its advantage lies in the flexibility of its one-to-many mappings between two sequences.

Euclidean and DTW distances [30] of time series Q and C are shown in Eq. 1 and 2, respectively, where $Q = \{q_1, q_2, ..., q_i, ..., q_n\}$, and $C = \{c_1, c_2, ..., c_j, ..., c_m\}$. Trivially, the Euclidean distance is the sum of distances between each of the one-to-one mapping of elements q_i and c_i, with $N = min\{m,n\}$, N being the number of sum terms. In the case of DTW, however, a $n \times m$ distance matrix is first constructed containing all possible distances for each q_i and c_i pairing. Then, among the numerous warping paths, demonstrated as $W = w_1, w_2, ..., w_k, ..., w_K$, the optimal warping path $min\{W\}$ is chosen if it minimizes the mapping between time series Q and C.

$$Dist(Euclidean) = \sqrt{\sum_{i=1}^{N}(q_i - c_i)^2} \tag{1}$$

$$Dist(DTW) = min\{W(Q,C)\} \tag{2}$$

DTW's effectiveness in finding similar shapes in data is contributed to the algorithm's ability to look within an allowed range for better mapping of data points, which corresponds to a certain step pattern. Various weighting patterns can also be implemented on the warping path. Eq. 3 could be viewed as a basic and commonly used step pattern. Here the cumulative distance $D(Q_i, C_j)$ is the sum of the current distance $d(q_i, c_j)$ and the minimum distance from the adjacent elements.

$$D(Q_i, C_j) = d(q_i, c_j) + min \left\{ \begin{array}{l} d(Q_i, C_{j-1}) \\ d(Q_{i-1}, C_{j-1}) \\ d(Q_{i-1}, C_j) \end{array} \right\} \tag{3}$$

For time series averaging, a widely applied technique is the DTW Barycenter Averaging (DBA) [31]. Instead of dividing the summation, DBA uses DTW to minimize Within Group Sum of Squares (WGSS). Simply put, given a time series set of $\mathbb{S} = \{S_1, S_2, ..., S_n\}$, the time series $C = \{c_1, c_2, ..., c_t\}$ is considered an average of $\mathbb{S}$ if it minimizes:

$$WGSS(C) = \sum_{k=1}^{n} dtw(C, S_n)^2 \tag{4}$$

C. Time Series Similarity

Here we propose three key elements of time series similarity, which are *range value similarity*, *duration similarity*, and *shape similarity*. Individually, none of these three elements can guarantee similarity for time series. Time series need to achieve high similarity in all these three aspects to be considered truly similar. In particular:

- Range value similarity refers to the absolute value range of time series. Range value similarity can be considered as vertical similarity; however, it does not promise actual similarity, but rather a comparable range. This can serve as a rough categorization, or threshold to differentiate large amount of data.
- Duration similarity can be considered a horizontal range value, referring to the time series measurement duration. Two events may have similar range values and shapes; however, if one lasted only seconds while the other lasted decades, then in most circumstances, they are not considered similar, unless a specific task calls for such usage.
- Shape similarity refers to the contour or shape of the time series. The shape similarity is an important part of the similarity elements; while used alone, it cannot guarantee similarity, there is most likely no similarity between time series when shape similarity is not present. In practice, shape similarity can be paired with range value similarity or duration similarity or both.

When all the elements of similarities mentioned above are high, we can conclude that the examined time series are highly similar. This is not to say that they have to occur simultaneously to be meaningful: each aspect of similarity could be individually significant in certain contexts. For example, when credit card companies analyze customer spending, both a big spender and an average spender would likely have higher spending in the holiday season, which could generate similar spending trends. While the shape can be useful when identifying seasonal spending trends, if the credit card company is interested in the customers' spending power, then the range value is of higher priority than the shape of the spending trend. Conversely, in this paper, we focus on the shape similarity or profile of pre-flare phases corresponding to different flares. Therefore, assuming that each flare prediction window has the same duration, we use normalization to reduce range value dissimilarity and focus on the shape similarities.

D. Clustering

The Distance Density Clustering (DDC) method [32] was developed for time series clustering. Here we use it to cluster pre-flare parameter time series with different normalization methods. DDC is divisive in structure, meaning that performance generally increases as more clusters are introduced. In the extreme case of each event forming its own cluster, the method degenerates to a k-Nearest Neighbors algorithm with $k = 1$ (i.e., 1NN), where each instance of testing data is compared to all the existing training data, and assigned the label of its single closest neighbor. While setting k to 1 can drastically improve the classification accuracy, conceptually, 1NN is a memorization process and not a generalization process. Memorization processes are less powerful in real-world applications, as comparing the current pre-flare measurements

against all historical instances is extremely time consuming, to the point of being unrealistic. Our ultimate goal is to provide generalized pre-flare profiles depending on standard or customized flare classes. In practice, this means that new data needs only to be compared against a handful of profiles and not the entire historical set of pre-flare records. The computation of comparing near-real-time measurements against generalized profiles is doable in real-time. Therefore a clustering of pre-flare time series data that generates shape summarization of the overall behavior could be crucial.

In order to generate the pre-flare profiles, we first need to generate the innate clusters from the flares time series data. While many existing clustering algorithms can be applied to time series, the effect is often limited. The DDC algorithm, on the other hand, has shown promise with more intuitive results. Algorithm 1 shows the main steps of the DDC. Initially, the furthest time series is first identified and serves as the initial cluster seed. The furthest time series being the time series that is the furthest from the most number of other time series. Then the distances between all instances and the cluster seed are computed and sorted. The most significant increase in the sorted distances is considered as a virtual sparse region and is used to divide the dataset. Then new cluster seeds are identified, and the cluster assignment is re-balanced based on time series similarity. This process is iterated until no more clusters can be found, or the process has reached a user-defined threshold, such as a certain number of clusters have been generated. Finally, all the identified cluster seeds and their respective cluster elements are obtained.

Algorithm 1 Distance Density Clustering Algorithm

Require:
$E = \{e_1, ..., e_n\}$ is the time series events to be clustered
$C_{k-1} = \{c_1, ..., c_{k-1}\}$ is the set of cluster seeds
k is number of seeds
L_k is the cluster set of events based on the number of groups

$L_{k-1} \leftarrow Cluster(C_{k-1})$
$ar[1,2,...,k-1] = DistSort(L_{k-1})$
$value[i] \leftarrow max(ar[2]-ar[1], ..., ar[k-1]-ar[k-2])$
if $ar[n]-ar[n-1] == max(value[i])$ **then**
    $location[i] = n$
end if
if **then**$i \leftarrow max(value[1,...,k-1])$
    $l(i_1, i_2) \leftarrow l(i), (c_{i_1}, c_{i_2}) \leftarrow c_i$
end if
return $L_n = \{1,2,...,i_1,i_2,...,n\} \leftarrow C_k\{(c_1, c_2, ..., c_{i_1}, c_{i_2}, ..., c_n)\}$
for $e_i \in E$ **do**
    $(c'_1, c'_2, ..., c'_k) \leftarrow DBA(c_1, c_2, ..., c_{i_1}, c_{i_2}, ..., c_{k-1})$
    $UpdateClusterDBA(C_k)$
end for
return $C'_k = \{c'_1, ..., c'_k\}$ **as set of cluster seeds**
return $L_n = \{l(e) \mid = 1,2,...,n\}$ **set of cluster labels of** E

III. Normalization Methods

In the context of data mining, normalization refers to the scaling of data attributes so that the data are restricted to a smaller vertical range. Normalization is generally required when we work on attributes with different scales. For solar flares, while the magnitude of different attribute values would signify the labeling of different flare classes, the progression of solar flares, or the shape of measurements, is often overlooked. Moreover, some generalized measurement values, such as average values, are available only after an event has completed, as opposed to the actual time series of the physical parameters. This makes the real-time forecasting task difficult. Shapes, on the other hand, are different. Even when the values are small, certain behavior of the time series in the past could be a good indication of certain behaviors in the future. By using normalization on flare data, we focus on the shapes of the physical parameters in the pre-flare phase.

In addition to the scale adjustment explained above, time series normalization also refers to the shifting and scaling of data to eliminate the effect of gross value influences. The four most commonly applied normalization techniques for time series data are *Offset Translation*, *Amplitude Scaling*, *Trend Removal*, and *Smoothing*. When a certain normalization is applied, the same normalization is applied to all the time series in the dataset.

A. Offset Translation

Offset translation means the shifting of time series. Offset is a signal processing term, used when sequences are similar in shape but are within different ranges. Shown in equation 5, offset translation means subtracting the mean from the original time series, namely,

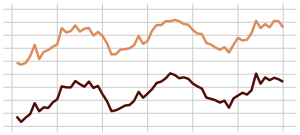
$$ts = ts - mean(ts) \tag{5}$$

Here the mean value is computed for each time series individually and is simply the average over all the values in that specific time series. The translation of the offset can be useful for similarity comparisons. However, an immediate drawback of this operation is that the range values would be overlooked since the value differences are removed. For stored (not real-time) time series, the offset can be removed by subtracting the mean amplitude from each sample. Although the mean is used, it is for the training process only; once the pre-flare profiles are established, the mean of new time series does not need to be computed. An example of an offset translation operation on a time series is shown in Fig. 1.

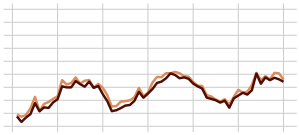
B. Amplitude Scaling

Amplitude is also a term borrowed from signal processing. It measures how far, and in which direction, does a variable differ from a defined baseline. Scaling of a signal's amplitude means changing the strength of the signal. With time series data, we remove the different amplitudes in hopes of finding similarity by excluding the physical parameter's strength.

$$ts = (ts - mean(ts))/std(ts) \tag{6}$$



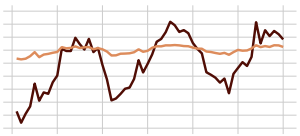
(a) Before offset translation



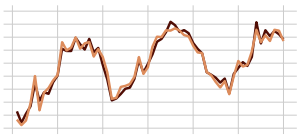
(b) After offset translation

Figure 1: Offset translation shifts the range of time series to focus on the shape similarities. (a) Shows the original time series, (b) shows the normalized result after offset translation is applied.

Shown in equation 6 and illustrated in Fig. 2, amplitude scaling is achieved by first moving time series by its mean and normalizing the amplitude by the standard deviation. Which means that in a way, offset translation is included in amplitude scaling. In fact, when $std(ts) = 1$, the two methods are identical.



(a) Before amplitude scaling



(b) After amplitude scaling

Figure 2: Amplitude scaling changes the strength of the signal (time series) to make shape similarities easier to identify. (a) Shows the original time series, (b) shows the time series after the amplitude removal.

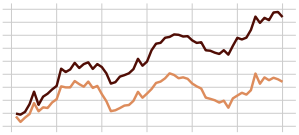
C. Trend Removal

Trend removal is mostly used in prediction. Trends represent long-term movements in sequences. In identifying patterns in sequence data, trends may become distracting, and therefore, it is often justified to remove them for revealing possible oscillations. To this end, the regression line of the time series needs to be identified and then subtracted from the time series. An example of linear trend removal is shown in Fig. 3. However, unlike offset translation and amplitude scaling, trend removal is not a straightforward operation. There can be different types of trends or even multiple trends. In our experiments, we only consider linear and logarithmic trends.

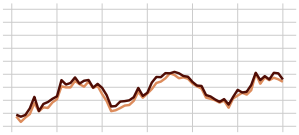
D. Smoothing

Smoothing is usually performed with a moving window on the time series to obtain the average values of each data point with those of its neighbors. It can eliminate some irregular movements, but is sensitive to outliers and also invalidates data at the beginning and end of the time series.

In our dataset, the time series are relatively short in length (i.e., 60 data points) and is also noisy in nature. An effective



(a) Before trend removal



(b) After trend removal

Figure 3: Trend removal removes the patterns, or trends, from time series data in the hope of focusing on the value changes without the long-term trend distraction. (a) Shows the original time series with a linear trend and (b) is where the trend is removed to reveal the similarity between the time series.

smoothing window is often relatively large, and would, therefore, shorten the time series excessively, rendering the result ineffective. For this reason, we will not utilize smoothing in our experiments.

IV. SWAN-SF Dataset

In this study, we use the Space Weather ANalytics for Solar Flares (SWAN-SF) [21], which is a benchmark dataset of multivariate time series (MVTS), spanning over the 9-year period (2010-2018) within the solar cycle 24. The time series in the data, as illustrated in Fig. 4, are the result of a sliding temporal window spanning over 12 hours of observation prior to the occurrence of an event. The sliding window extracts a list of physical (magnetic field) parameters from regions of interest (RoI) and produces an MVTS. The label assigned to each MVTS corresponds to the strongest flare in the temporal observation window, among the multiple flares that may co-occur. Of course, the presence of multiple flares in an observation window impacts the general flares' profiles. However, an operation-ready forecast system also needs to predict based on the characteristics of flare clusters, and not singled-out flares. This is simply a design choice in SWAN-SF, made to closely mimic the data that any real-time forecasting model should eventually base their predictions on. If no flares were reported during that period, the corresponding MVTS would be labeled as flare-quiet (FQ). Note that each MVTS is unique to a particular active region. However, due to the use of a sliding window for slicing time series, multiple observation windows may be attributed to a single flare occurring in that region. This behavior is reflected in Fig. 4 by the top-tree blue bars, representing three different MVTS, all corresponding to one flare of magnitude M1.0.

This data benchmark comprises of five partitions in such a way that there is approximately an equal number of strong (GOES M- and X-class) flares, distributed in each partition. The partitions are temporally separated. This provides an easy way for users to split the data into training, validation, and testing sets, without having to worry about unwanted biases that their sampling methodology may impose on the problem. Table. I shows the number of instances (pre-flare time series) labeled in compliance with the settings of Fig. 4: a 12-hour

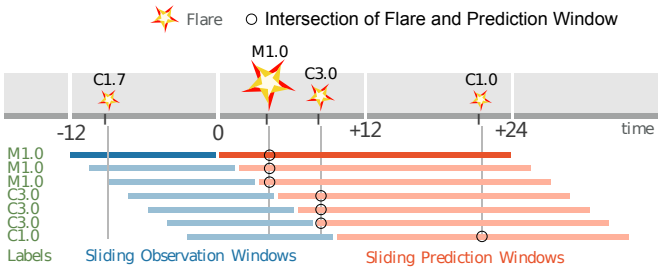


Figure 4: Visualization of the slicing process employed by [21] for the collection of multivariate time series in the SWAN-SF benchmark dataset. For a given active region, each observation window (blue bars) is labeled (the text is on the left, in green) according to the intersection (black circles) of its prediction window (red bars) with the magnitude of the largest flare reported for this active region in that time window. Note that each observation window does not produce one time series, but rather one MVTS.

Table I: Labeled flare instances as per flare class for the five partitions of the SWAN-SF benchmark dataset. Also shown is the approximate class imbalance ratio between labels corresponding to M- and X-class flares and all the other labels.

Partition	X	M	C	B	FQ	Ratio
1	172	1130	6250	4999	64222	1:58
2	48	1279	8444	4194	78517	1:69
3	160	1152	3350	108	22236	1:20
4	165	1153	6487	832	52689	1:51
5	21	1071	6419	832	89400	1:95

observation window and a 24-hour forecast window, with zero latency.

Our analysis is based on the notion that flare data do not only differ in value, but also in the development process, and it is the latter that is our emphasize in this study. As this paper is simply a proof of concept, we did not want to emphasize the quantity of experimental data. Instead, we selected 100 unique C- and M-class instances from Partitions 4 and 5, respectively. Moreover, because X-class instances are rare events, and it is important to have balanced datasets for cluster analysis, we selected X-class instances from across all five Partitions. For obtaining 100 X-class flare instances, if they have similar active region id, we limited our sample space to time series that are as spread out as possible. This is to avoid the impact of auto-correlation caused by the time series coming from the same active regions.

V. Experiments and Results

In this section, we present our findings in clustering normalized pre-flare time series data. In order to eliminate the randomness of performance, we performed a balanced 5-fold cross-validation on the curated dataset of a total of 300 C-, M-, and X-class instances. Cross-validation is a statistical evaluation method used to evaluate machine learning models on a limited data sample fairly. For each fold, the testing data is never included in the training process to avoid bias.

Table II: Nine parameters selected by domain experts of which solar pre-flare time series are evaluated.

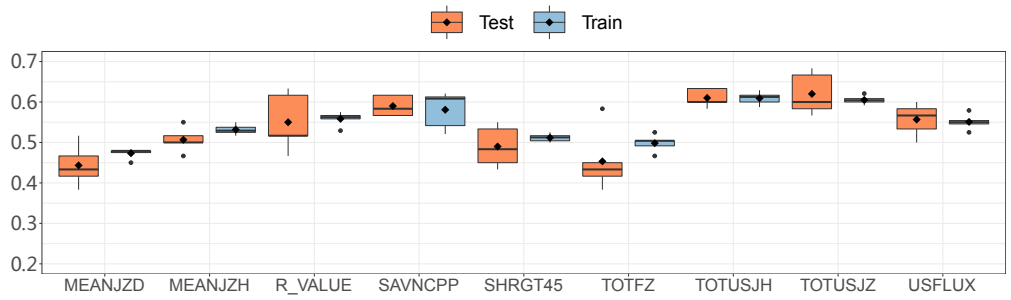
	Keyword	Description
1	MEANJZD	Mean vertical current density
2	MEANJZH	Mean current helicity
3	R_VALUE	Sum of flux near polarity inversion line
4	SAVNCPP	Sum of the modulus of the net current per polarity
5	SHRGT45	Fraction of area with shear angle $> 45°$
6	TOTFZ	Sum of z-component of Lorentz force
7	TOTUSJH	Total unsigned current helicity
8	TOTUSJZ	Total unsigned vertical current
9	USFLUX	Total unsigned flux

The first step is to obtain and compare clustering results. Distance Density Clustering is performed on the normalized time series of all the partitions on nine SHARP parameters: MEANJZD, MEANJZH, R_VALUE, SAVNCPP, SHRGT45, TOTFZ, TOTUSJH, TOTUSJZ, and USFLUX. Described briefly in Table II, the nine parameters are selected by domain experts from the list of parameters discussed by Bobra et al. [16]. The different parameters are simply different measurements of solar flares, and should not hinder non-domain experts from understanding the results of this paper.

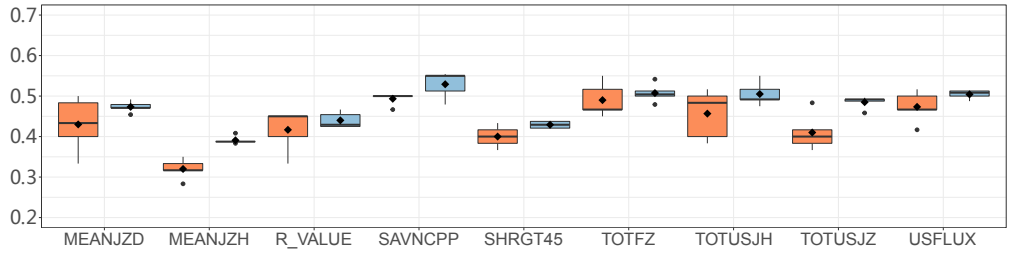
Fig. 5 shows the boxplot of cluster accuracy on training and testing data. In each partition for each normalization method, the data is clustered to 10 clusters. The label of a cluster is determined by the majority event labels, and multiple clusters can have the same flare class label. The x-axis shows the nine parameters, and the y-axis is the accuracy value. The accuracy at cluster 10 is computed based on the matches between predicted labels and actual labels for all five partitions from the 5-fold cross-validation. For each parameter, the left boxplot is the testing accuracy of each normalization method, and the right boxplot is the training accuracy of the respective normalization methods. Parameters R_VALUE, SAVNCPP, TOTUSJH, and TOTUSJZ generally have better accuracy performance. Normalization accuracy is typically lower than the original data cluster accuracy, with offset translation, difference detrend, and logarithmic detrend showing similar parameter performance patterns as the original data clustering accuracy.

While some machine learning algorithms achieve better results from normalization, this is not the case for pre-flare time series clustering. Although most of the time, we are able to obtain more shape details with normalization, the accuracy often declines. This is caused by the fact that when normalization is applied, only two out of three similarity elements are met, the duration and shape similarity, the range value similarity is lost, and this would have a negative impact on clustering accuracy. Therefore, in our experiments, the accuracy values are only used as a reference, not a quality indicator.

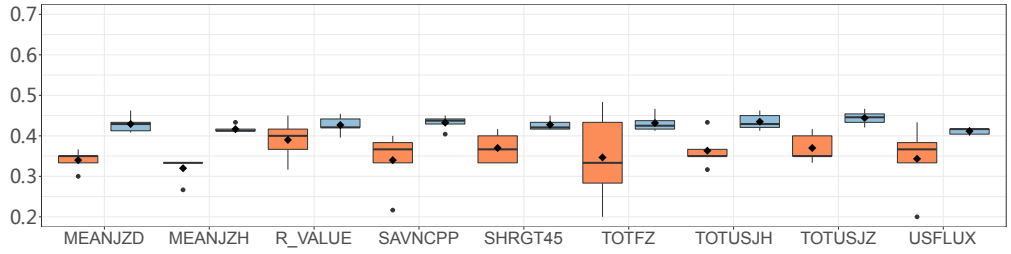
Due to space limitations, we only select one parameter to demonstrate the details of further investigation. TOTUSJZ is chosen as it generally has good performance and clear cut clusters. We first look at the original, unnormalized, pre-flare time series of GOES class C, M, and X on parameter



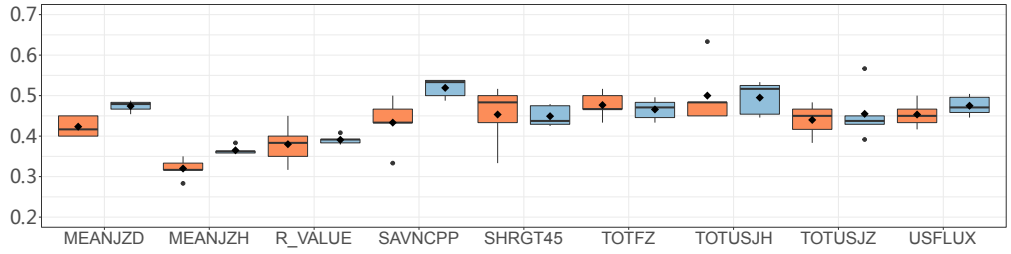
(a) Original cluster accuracy



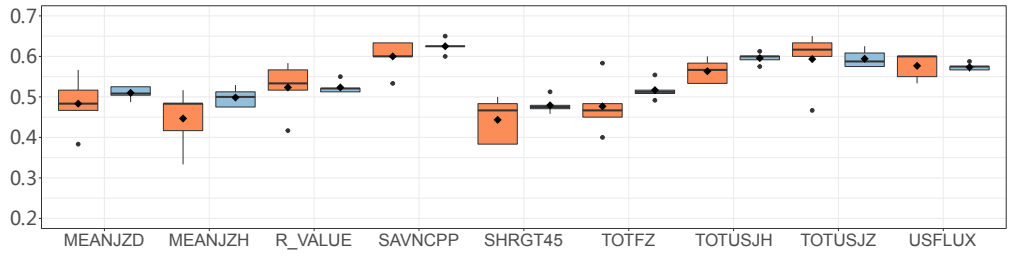
(b) Offset translation cluster accuracy boxplot



(c) Amplitude scaling cluster accuracy boxplot

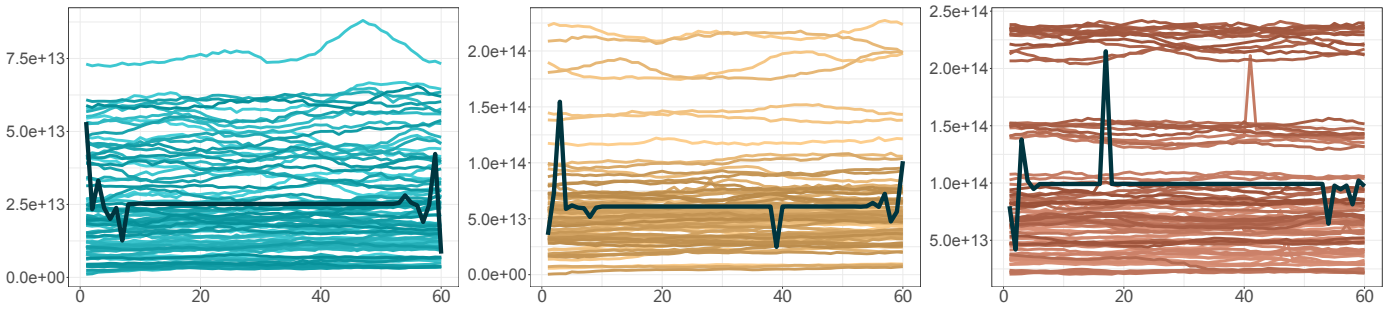


(d) Difference detrend cluster accuracy boxplot



(e) Logarithmic detrend cluster accuracy boxplot

Figure 5: Boxplots with (a) no normalization (original data), (b) offset translation, (c) amplitude scaling, (d) detrend with difference and (e) detrend with log. The horizontal axis is the corresponding parameters, and the vertical axis is the accuracy. The left (orange) boxplot of each parameter is the testing boxplot, the right (blue) boxplot is the training boxplot; each boxplot contains the accuracy information from all five partitions.



(a) Time series for pre-flare class C. (b) Time series for pre-flare class M. (c) Time series for pre-flare class X.

Figure 6: In the first partition of the 5-fold cross-validation, the original pre-flare time series of (a) C-class, (b) M-class, and (c) X-class of parameter TOTUSJZ.

TOTUSJZ. Fig. 6 shows the C-, M-, and X-class pre-flare time series, along with the time series average with DBA for each class (black lines). We can observe more spikes at the beginning and end of the average time series, likely due to common spikes in many averaged time series. Later on, as the evens progress, there is more diversity in the time series events shapes (values). The value differences of the pre-flare time series for different classes can be easily seen. However, the shape of different pre-flare classes is not apparent; the shape differences are likely made visually insignificant by the large value differences within the same class. The general flat shape of the average time series suggests a lack of patterns to be identified.

In regards to the three aspects of similarity, since we already have duration similarity in place (i.e., all time series of SWAN-SF are of length 60), we try to emphasize shape similarity while reducing range value similarities. Fig. 7 shows one of the clusters for parameter TOTUSJZ when we generate 5 clusters, the blue lines are C-class, yellow lines are M-class, and orange lines are X-class. In Fig. 7 (a) the clustered data are all normalized with offset translation, (b),(c), and (d) shows the normalized C-, M-, and X-class pre-flare time series respectively. Fig. 7 (e) shows the original (unnormalized) time series which appear in Fig. 7 (a), and (f), (g), and (h) are the original time series of C-, M-, and X-class pre-flares respectively. Similar to Fig. 6, the unnormalized time series are flat, lacking shape similarities or dissimilarities identification.

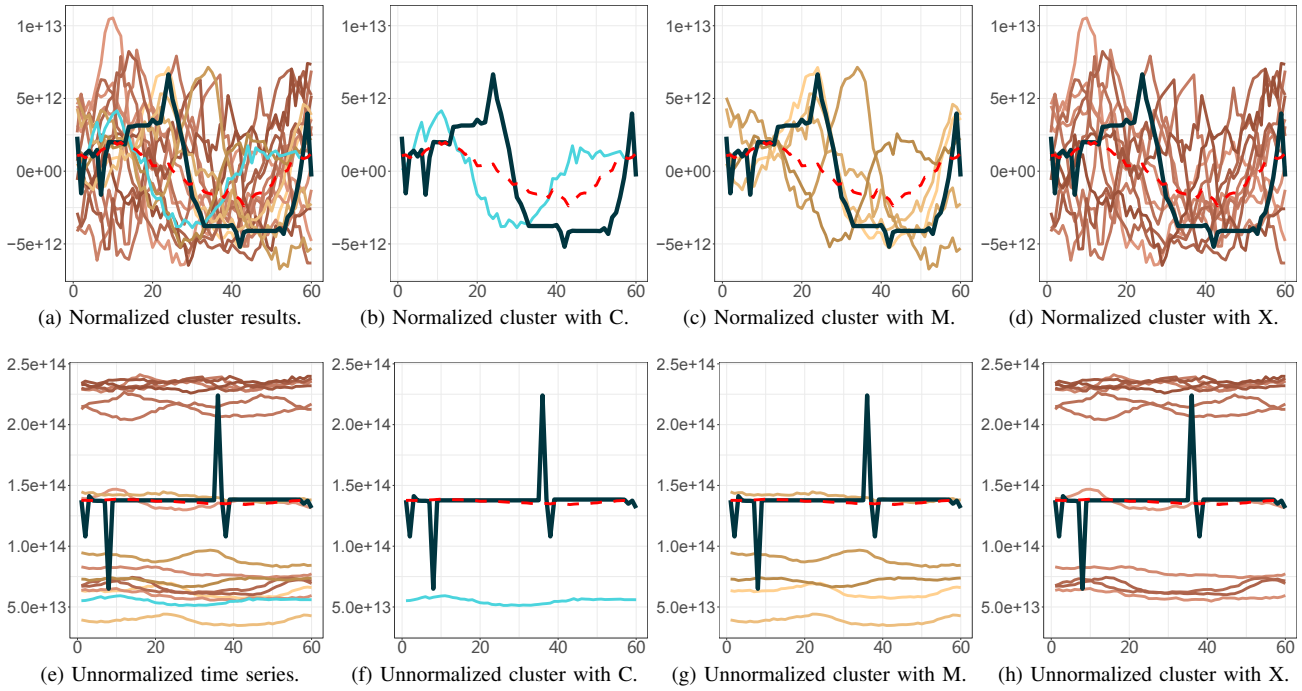
All the solid lines in Fig. 7 show the DBA time series averages, and the red dotted lines are the conventional averages. Conventional averages are simply the sum of all the instances at one time point divided by the total number of instances. When the time series are unnormalized, it is difficult for both the DBA and the conventional average to identify an intuitive representation of the original time series. In the case of normalized time series, it is still difficult for the conventional averaging technique to pick up the representative shape of the time series, which can be due to the difficulty of picking up shape similarities by single time point averaging. Generally, DBA for normalized data is more useful in shape profile identification, and DBAs are more meaningful than conventional average generalization. It is important to note that the shape signatures we identify here are insensitive to the parameter value ranges. When we normalize, we obtain shape intensive information but miss value differences. On the other hand, when we work with time series that are unnormalized, we preserve the parameter values, but overlook shape information. Therefore, the combination of both the shape information as well as the value differences would be worth investigating in the future.

When comparing the normalized and unnormalized time series of time series from the same cluster, we can see similar shapes from different classes with a different value range. The one C-class instance in this cluster has a small value compared to other time series; the M-class time series are all below the average line, and the X-class is demonstrating two concentrations in time series value range distribution. This suggests that different flare classes can demonstrate similar shapes, as well as the possibility of sub-clusters existing within what we currently understand as one class of flares.

VI. Conclusion

The application of normalization could segregate the pre-flare classification problem into value and shape. In this paper, we approached solar pre-flare labeling from a novel shape profile perspective, from which we hope to obtain new interpretations and, perhaps, clues for future flare forecasting efforts. The latter is not yet attempted here. However, this study shows the feasibility of identifying flares not just by parameter values, but by examining pre-flare shapes. By placing emphasis on the shape profiles of solar pre-flares, we are able to identify similar shapes between different class of pre-flares.

Our next step is to perform more comprehensive experiments and obtain more precise pre-flare shape profiles. Then by combining the shape profiles with parameter values, it may become possible to perform flare forecasting. Additionally, pre-flare shape profiles could help us identify sub-classes within what we currently define as one flare class.



(a) Normalized cluster results. (b) Normalized cluster with C. (c) Normalized cluster with M. (d) Normalized cluster with X.

(e) Unnormalized time series. (f) Unnormalized cluster with C. (g) Unnormalized cluster with M. (h) Unnormalized cluster with X.

Figure 7: Cluster details of one of five clusters of parameter TOTUSJZ, in this particular cluster, there are a total of 18 pre-flare time series, 1 C-class, 5 M-class, and 12 X-class. The blue lines are C-class pre-flares, yellow lines are M-class pre-flares, and orange lines are X-class pre-flares. The solid line is the time series average, while the red dotted line is the conventional average. The normalized time series and their respective averages on the top row, (a) the entire cluster time series, (b) C-class pre-flare, (c) M-class pre-flares, (d) X-class pre-flares. The time series in this particular cluster but in the unnormalized form is shown on the bottom row, (e) the entire cluster of unnormalized time series with its corresponding averages, (f) C-class pre-flare, (g) M-class pre-flare, (h) X-class pre-flare.

Acknowledgment

This work was supported in part by two NASA Grant Award [No. NNX11AM13A, and No. NNX15AF39G], and one NSF Grant Awards [No. AC1443061]. The AC1443061 award has been supported by funding from the Division of Advanced Cyber infrastructure within the Directorate for Computer and Information Science and Engineering, the Division of Astronomical Sciences within the Directorate for Mathematical and Physical Sciences, and the Division of Atmospheric and Geospace Sciences within the Directorate for Geosciences.

References

[1] A. O. Benz, "Flare Observations," *Living Reviews in Solar Physics*, vol. 5, no. 1, p. 1, Feb 2008.

[2] K. L. Harvey and C. Zwaan, "Properties and Emergence Patterns of Bipolar Active Regions - Part One," *solphys*, vol. 148, no. 1, pp. 85–118, Nov 1993.

[3] J. Zhang, K. Dere, R. Howard, M. Kundu, and S. White, "On the temporal relationship between coronal mass ejections and flares," *The Astrophysical Journal*, vol. 559, no. 1, p. 452, 2001.

[4] S. S. Board, "Committee on the societal and economic impacts of severe space weather events: A workshop," 2008.

[5] J. Eastwood, E. Biffis, M. Hapgood, L. Green, M. Bisi, R. Bentley, R. Wicks, L.-A. McKinnell, M. Gibbs, and C. Burnett, "The economic impact of space weather: Where do we stand?" *Risk Analysis*, vol. 37, no. 2, pp. 206–218, 2017.

[6] E. J. Oughton, A. Skelton, R. B. Horne, A. W. P. Thomson, and C. T. Gaunt, "Quantifying the daily economic impact of extreme space weather due to failure in electricity transmission infrastructure," *Space Weather*, vol. 15, no. 1, pp. 65–83, Jan 2017.

[7] D. S. Bloomfield, P. A. Higgins, R. J. McAteer, and P. T. Gallagher, "Toward reliable benchmarking of solar flare forecasting methods," *The Astrophysical Journal Letters*, vol. 747, no. 2, p. L41, 2012.

[8] G. Barnes *et al.*, "A comparison of flare forecasting methods. i. results from the "all-clear "workshop," *The Astrophysical Journal*, vol. 829, no. 2, p. 89, Sep 2016. [Online]. Available: https://iopscience.iop.org/article/10.3847/0004-637X/829/2/89

[9] K. Leka, S.-H. Park, K. Kusano, J. Andries, G. Barnes, S. Bingham, D. S. Bloomfield, A. E. McCloskey, V. Delouille, D. Falconer *et al.*, "A comparison of flare forecasting methods. ii. benchmarks, metrics, and performance results for operational solar flare forecasting systems," *The Astrophysical Journal Supplement Series*, vol. 243, no. 2, p. 36, 2019.

[10] M. K. Georgoulis, "On Our Ability to Predict Major Solar Flares," *Astrophysics and Space Science Proceedings*, vol. 30, p. 93, Jan 2012.

[11] K. Leka and G. Barnes, "Chapter 3 - solar flare forecasting: Present methods and challenges," in *Extreme Events in Geospace*, N. Buzulukova, Ed. Elsevier, 2018, pp. 65 – 98. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128127001000030

[12] R. Qahwaji and T. Colak, "Automatic Short-Term Solar Flare Prediction Using Machine Learning and Sunspot Associations," *solphys*, vol. 241, no. 1, pp. 195–211, Mar 2007.

[13] T. Colak and R. Qahwaji, "Automated solar activity prediction: a hybrid computer platform using machine learning and solar imaging for automated prediction of solar flares," *Space Weather*, vol. 7, no. 6, 2009.

[14] J. Mason and J. Hoeksema, "Testing automated solar flare forecasting with 13 years of michelson doppler imager magnetograms," *The Astrophysical Journal*, vol. 723, no. 1, p. 634, 2010.

[15] O. W. Ahmed, R. Qahwaji, T. Colak, P. A. Higgins, P. T. Gallagher, and D. S. Bloomfield, "Solar flare prediction using advanced feature extraction, machine learning, and feature selection," *Solar Physics*, vol. 283, no. 1, pp. 157–175, 2013.

[16] M. G. Bobra and S. Couvidat, "Solar flare prediction using sdo/hmi vector magnetic field data with a machine-learning algorithm," *The Astrophysical Journal*, vol. 798, no. 2, p. 135, 2015.

[17] A. M. Massone, M. Piana, and F. Consortium, "Chapter 14 - machine learning for flare forecasting," in *Machine Learning Techniques for Space Weather*, E. Camporeale, S. Wing, and J. R. Johnson, Eds. Elsevier, 2018, pp. 355 – 364. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780128117880000147

[18] K. Florios, I. Kontogiannis, S.-H. Park, J. A. Guerra, F. Benvenuto, D. S. Bloomfield, and M. K. Georgoulis, "Forecasting solar flares using magnetogram-based predictors and machine learning," *Solar Physics*, vol. 293, no. 2, p. 28, Jan 2018. [Online]. Available: https://doi.org/10.1007/s11207-018-1250-4

[19] N. Nishizuka, K. Sugiura, Y. Kubo, M. Den, and M. Ishii, "Deep Flare Net (DeFN) Model for Solar Flare Prediction," *apj*, vol. 858, no. 2, p. 113, May 2018.

[20] C. Campi, F. Benvenuto, A. M. Massone, D. S. Bloomfield, M. K. Georgoulis, and M. Piana, "Feature ranking of active region source properties in solar flare forecasting and the uncompromised stochasticity of flare occurrence," *arXiv e-prints*, p. arXiv:1906.12094, Jun 2019.

[21] B. Aydin *et al.*, "Multivariate time series dataset for space weather data analytics (manuscript submitted for publication)," *Scientific Data*, 2019.

[22] B. Aydin, D. Kempton, S. Mahajan, S. Basodi, A. Ahmadzadeh, S. Filali Boubrahimi, S. M. Hamdi, M. Schuh, M. Georgoulis, P. Martens, and R. Angryk, "Multivariate Time Series Dataset for Space Weather Data Analytics," 2019. [Online]. Available: https://doi.org/10.7910/DVN/EBCFKM

[23] M. G. Bobra, X. Sun, J. T. Hoeksema, M. Turmon, Y. Liu, K. Hayashi, G. Barnes, and K. Leka, "The helioseismic and magnetic imager (hmi) vector magnetic field pipeline: Sharps–space-weather hmi active region patches," *Solar Physics*, vol. 289, no. 9, pp. 3549–3578, 2014.

[24] P. H. Scherrer, J. Schou, R. I. Bush, A. G. Kosovichev, R. S. Bogart, J. T. Hoeksema, Y. Liu, T. L. Duvall, J. Zhao, A. M. Title, C. J. Schrijver, T. D. Tarbell, and S. Tomczyk, "The Helioseismic and Magnetic Imager (HMI) Investigation for the Solar Dynamics Observatory (SDO)," *solphys*, vol. 275, no. 1-2, pp. 207–227, Jan 2012.

[25] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining and Knowledge Discovery*, vol. 26, no. 2, pp. 275–309, 2013.

[26] H. Sakoe, "Dynamic-programming approach to continuous speech recognition," in *1971 Proc. the International Congress of Acoustics, Budapest*, 1971.

[27] C. Myers and L. Rabiner, "A level building dynamic time warping algorithm for connected word recognition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 29, no. 2, pp. 284–297, 1981.

[28] E. Keogh and C. A. Ratanamahatana, "Exact indexing of dynamic time warping," *Knowledge and information systems*, vol. 7, no. 3, pp. 358–386, 2005.

[29] Y.-S. Jeong, M. K. Jeong, and O. A. Omitaomu, "Weighted dynamic time warping for time series classification," *Pattern Recognition*, vol. 44, no. 9, pp. 2231–2240, 2011.

[30] M. Müller, "Dynamic time warping," *Information retrieval for music and motion*, pp. 69–84, 2007.

[31] F. Petitjean, A. Ketterlin, and P. Gançarski, "A global averaging method for dynamic time warping, with applications to clustering," *Pattern Recognition*, vol. 44, no. 3, pp. 678–693, 2011.

[32] R. Ma and R. Angryk, "Distance and density clustering for time series data," in *Data Mining Workshops (ICDMW), 2017 IEEE International Conference on*. IEEE, 2017, pp. 25–32.