12-2023

Solar Flare Prediction From Extremely Imbalanced Multivariate Time Series Data Using Minimally Random Convolutional Kernel Transform

Kartik Saini
Utah State University, kartik.saini@usu.edu

SOLAR FLARE PREDICTION FROM EXTREMELY IMBALANCED MULTIVARIATE TIME SERIES DATA USING MINIMALLY RANDOM CONVOLUTIONAL KERNEL TRANSFORM

by

Kartik Saini

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTER OF SCIENCE

in

Computer Science

Approved:

Shah Muhammad Hamdi, Ph.D. Major Professor	Soukaina Filali Boubrahimi, Ph.D. Committee Member
Steve Petruzza, Ph.D. Committee Member	D. Richard Cutler, Ph.D. Vice Provost of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2023

ABSTRACT

Solar Flare Prediction from Extremely Imbalanced Multivariate Time Series Data using Minimally Random Convolutional Kernel Transform

by

Kartik Saini, MASTER OF SCIENCE

Utah State University, 2023

Major Professor: Shah Muhammad Hamdi, Ph.D.
Department: Computer Science

Solar flares are characterized by sudden bursts of electromagnetic radiation from the Sun's surface, and caused by the changes in magnetic field states in solar active regions. Earth and its surrounding space environment can suffer from various negative impacts caused by solar flares ranging from electronic communication disruption to radiation exposure-based health risks to the astronauts. In this paper, we address the solar flare prediction problem from magnetic field parameter-based multivariate time series (MVTS) data using multiple state-of-the-art machine learning classifiers that include MINImally RandOm Convolutional KErnel Transform (MINIROCKET), Support Vector Machine (SVM), Canonical Interval Forest (CIF), Multiple Representations SEQuence Learner (MR-SEQL), Long Short-Term Memory (LSTM)-based deep learning model, and the Transformer model. We showed our results on the Space Weather ANalytics for Solar Flares (SWAN-SF) benchmark data set, a partitioned collection of MVTS data of active region magnetic field parameters spanning over 9 years of operation of the Solar Dynamics Observatory (SDO). The MVTS instances of the SWAN-SF dataset are labeled by GOES X-ray flux-based flare class labels, and attributed to extreme class imbalance because of the rarity of the major flaring events (e.g., X and M). To minimize the dimensionality of the data, we also included data

preprocessing activities such as statistical summarization. We used the true skill statistic (TSS) and realizations of the Heidke Skill Score (HSS; HSS2) score as a performance validation metric in this class-imbalanced dataset. Finally, we demonstrate the advantages of the MVTS learning algorithm MINIROCKET, which produces better results than other classifiers without the need for essential data preprocessing steps such as normalization, statistical summarization, and class imbalance handling heuristics.

(62 pages)

PUBLIC ABSTRACT

Solar Flare Prediction from Extremely Imbalanced Multivariate Time Series Data using Minimally Random Convolutional Kernel Transform

Kartik Saini

Solar flares are characterized by sudden bursts of electromagnetic radiation from the Sun's surface, and caused by the changes in magnetic field states in solar active regions. Earth and its surrounding space environment can suffer from various negative impacts caused by solar flares ranging from electronic communication disruption to radiation exposure-based health risks to the astronauts. In this paper, we address the solar flare prediction problem from magnetic field parameter-based multivariate time series (MVTS) data using multiple state-of-the-art machine learning classifiers that include MINImally RandOm Convolutional KErnel Transform (MINIROCKET), Support Vector Machine (SVM), Canonical Interval Forest (CIF), Multiple Representations SEQuence Learner (MR-SEQL), Long Short-Term Memory (LSTM)-based deep learning model, and the Transformer model. We showed our results on the Space Weather ANalytics for Solar Flares (SWAN-SF) benchmark data set, a partitioned collection of MVTS data of active region magnetic field parameters spanning over 9 years of operation of the Solar Dynamics Observatory (SDO). The MVTS instances of the SWAN-SF dataset are labeled by GOES X-ray flux-based flare class labels, and attributed to extreme class imbalance because of the rarity of the major flaring events (e.g., X and M). To minimize the dimensionality of the data, we also included data preprocessing activities such as statistical summarization. We used the true skill statistic (TSS) and realizations of the Heidke Skill Score (HSS; HSS2) score as a performance validation metric in this class-imbalanced dataset. Finally, we demonstrate the advantages of the MVTS learning algorithm MINIROCKET, which produces better results than other

classifiers without the need for essential data preprocessing steps such as normalization, statistical summarization, and class imbalance handling heuristics.

To the experiences we never expected and the paths that were redirected.
To the friends and family we found along the way.

ACKNOWLEDGMENTS

First and foremost, I extend my heartfelt gratitude to my major professor, Dr. Shah Muhammad Hamdi, for believing in me and granting me the opportunity to excel. His invaluable advice, unwavering encouragement, and remarkable patience throughout my Master's studies have profoundly shaped me into a more diligent student, a more receptive learner, and an overall better individual.

I would also like to express my sincere appreciation to Dr. Soukaina Filali Boubrahimi and Dr. Steve Petruzza for graciously accepting the responsibility of being part of my committee and providing unwavering support. Their guidance and insights have been instrumental in the development of my research work.

Finally, I am deeply indebted to my friends and family for their immeasurable support and relentless encouragement. Their unwavering belief in my abilities has been the driving force behind the successful completion of my studies.

Kartik Saini

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

ACRONYMS

AR	Active Region
CIF	Canonical Interval Fores
GOES	Geostationary Operational Environmental Satellite
HMI	Helioseismic Magnetic Imager
HSS2	Heidke Skill Score
kNN	K Nearest Neighbor
LSTM	Long Short-Term Memory
MDI	Michelson Doppler Imager
MINIROCKET	Minimally Random Convolutional Kernel Transform
MR-SEQL	Multiple Representations Sequence Learner
MVTS	Multivariate Time Series
NASA	National Aeronautics and Space Administration
NOAA	National Oceanic and Atmospheric Administration
PPV	Percentage of Positive Values
RF	Random Forest
RNN	Recurrent Neural Network
SAX	Symbol Aggregation Approximation
SDO	Solar Dynamics Observatory
SEC	Space Environment Center
SGD	Solar Geophysical Data
SHARP	Space weather HMI Active Region Patch
SFA	Symbol Fourier Approximation
SVM	Support Vector Machine
SWAN-SF	Space Weather Analytics for Solar Flares
TSS	True Skill Statistic

CHAPTER 1

INTRODUCTION

Solar flares are strong outbursts of radiation that result from the sudden release of magnetic energy that has been stored in the Sun. The length of time that a solar flare lasts can range anywhere from a few minutes to 24-48 hours. Since 1974, the Geostationary Operational Environmental Satellite (GOES) that are operated by the National Oceanic and Atmospheric Administration (NOAA) have been able to detect and categorize the X-ray flux that is produced by flare events in the 1-8 wavelength range. Based on their peak soft X-ray emission in this range, flares are grouped logarithmically as A, B, C, M, and X, ascending from less powerful to more powerful, starting 10^{-8} W m^{-2} [1]. As a direct consequence of this, the peak X-ray flux of an X-class flare is typically one hundred times stronger than that of a C-class flare and ten times stronger than that of an M-class flare. Each class can be broken down into one of nine sub-classes ranging from, e.g., C1 to C9, M1 to M9, and X1 to X9. When the X-ray level is high, it is sometimes difficult or even impossible to detect flares of the A and B classes. However, flares of the C class and higher are identified the vast majority of the time, particularly above level C2. Because of the potential damage they could cause, flares of the M and X classes, which are the most severe, are typically the focus of space weather forecasting.

Flares of the X-class and the M-class have the potential to cause radio blackouts across the Earth and initiate persistent radiation storms in the upper atmosphere. It's possible to put astronauts, flight attendants, and passengers in danger. According to [2], the devastation that was caused by this solar flare could end up costing more than a trillion dollars to repair and replace the damaged infrastructure. By taking the necessary precautions and utilizing an effective solar flare prediction system, the majority of damage can be mitigated.

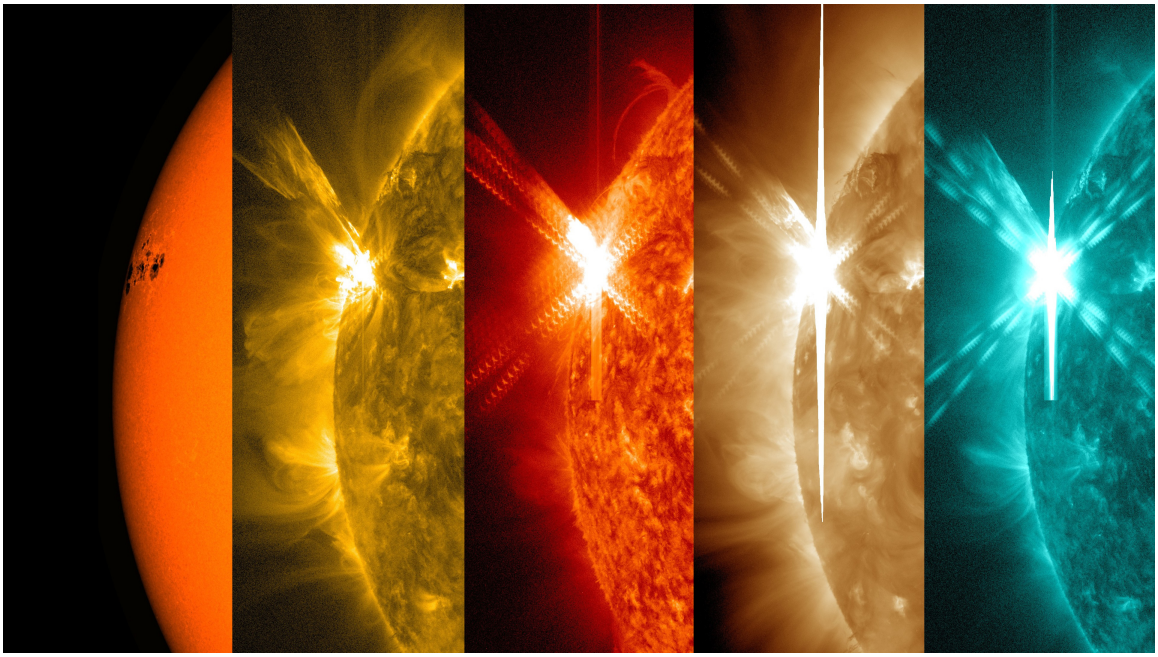

Fig. 1.1: On May 5, 2015, NASA's Solar Dynamics Observatory captured these images of an X2.7-class solar flare – as seen in the bright flash on the left. Each image shows a different wavelength of extreme ultraviolet light that highlights a different temperature of material on the sun. From left to right, the wavelengths are: visible light, 171 angstroms, 304 angstroms, 193 angstroms and 131 angstroms. Each wavelength has been colorized. By comparing different images, scientists can better understand the movement of solar matter and energy during a flare. Credits: NASA/SDO/Wiessinger

The rarity of the event, however, is the most difficult part of addressing solar flare classification. As stated by NASA, the solar cycle, which lasts 11 years, influences the frequency with which solar flares occur. During times of solar maximum, it may occur multiple times each day, and during times of solar minimum, it may occur less than once per week. In addition, stronger flares occur less frequently than their less intense counterparts. For instance, flares of the X10 class, which are considered severe, occur on average approximately eight times every cycle, making it a very rare phenomenon. Flares of the M1 class, which are considered small, occur on average around two thousand times per cycle. Since there is a significant imbalance in the class distribution, traditional classifiers struggle to predict the minority class with high accuracy. The majority of classification-based machine learning algorithms were built on the assumption of an equal ratio of samples for each class [3].

Furthermore, the dataset comprises a range of time series parameters obtained from solar photospheric magnetograms, in addition to NOAA's record of flares in active regions (Refer table 3.1). It also includes physics-based magnetic field parameters, originally acquired through the Space Weather HMI Active Region Patches (SHARP) data product [4]. This high-dimensional time series data poses another challenge because of the curse of dimensionality and the fact that many feature vectors can turn out to be noise. As a result, if we do not minimize the dimension of the original input and reduce the feature set, we will almost certainly overfit. Classifiers that achieve credible accuracy on imbalanced time series data are quite computationally expensive. Even with small data sets, they can take a substantial amount of time to train.

Ahmadzadeh et al. [1] comprehensively presented the challenges involved in handling the SWAN-SF Dataset, which is the largest dataset to date on solar flares. The dataset is based on MVTS-based photospheric magnetic field parameters of solar active regions. The authors discussed the extreme class imbalance in the data as well as the temporal coherence and proposed different remedies to tackle these problems. They began by extracting the statistical features of each magnetic field parameter time series, such as median, standard deviation, skewness, and kurtosis. Additionally, they included the last value of each time series, which reduced the dimensionality of the dataset and made it scalable. They then used an SVM classifier to test the performance of flare prediction. To address the class imbalance issue, they employed both undersampling and oversampling techniques as mandatory preprocessing steps. At the classifier level, they tuned the misclassification weighting parameter to minimize false positives and false negatives. For temporal coherence, they utilized 20 pairs of testing and training data from different partitions to avoid overlapping the sampled MVTS sequence. To evaluate the forecast performance, they used True Skill Statistics (TSS) and realizations of the Heidke Skill Score (HSS; HSS2) to assess the robustness of the SVM model. However, these experimental settings had some limitations. Firstly, the calculated five statistical features may not accurately represent the complete properties of the

time series data, potentially leading to inaccurate predictions. Furthermore, an important data preprocessing step, which involves undersampling or oversampling of training data, is computationally expensive and complicates the application of the proposed ML model in real-time flare forecasting systems.

To tackle the aforementioned drawbacks, we utilize the MINImally RandOm Convolutional KErnel Transform, MINIROCKET [5] time series classifier to predict solar flares in real-time with minimal data manipulation and preprocessing. MINIROCKET is based on the state-of-the-art ROCKET classifier [6], which achieves high precision at a fraction of the computing cost compared to most existing approaches. It achieves this by transforming input time series using random convolutional kernels and training a linear classifier with the transformed features. MINIROCKET is a (nearly) deterministic reformulation of ROCKET that performs significantly faster on larger datasets while maintaining comparable accuracy. Additionally, we trained the Canonical Interval Forest (CIF) [7], Multiple Representations SEQuence Learner (MR-SEQL), and Long Short-Term Memory (LSTM) deep learning models to compare MINIROCKET's performance with other classifiers.

Additionally, we implemented the Transformer model introduced by [8] on the reduced dataset. The transformer model is a neural network architecture based solely on self-attention mechanisms, revolutionizing the field of natural language processing (NLP) and serving as the foundation for subsequent advancements, including state-of-the-art models like BERT and GPT [9]. The key advantage of the Transformer model is its ability to efficiently capture long-range dependencies in parallel, leading to faster training and inference times compared to previous models. Given its effectiveness, the Transformer model can be a powerful choice for MVTS classification, leveraging its ability to capture long-range dependencies and handle multi-variable, temporal data effectively. We aimed to explore an alternative approach using Attention/Transformer Model-based techniques. By harnessing the power of self-attention mechanisms in Transformers, we strive to capture temporal

dependencies and correlations among magnetic field parameters in the MVTS data. Ultimately, this approach aims to improve solar flare classification performance and deepen our understanding of these potentially catastrophic events. To assess the performance of our model, we compared it with six other MVTS-based flare prediction baselines: Flattened vector method (FLT), Vector of last timestamp (LTV), Time series summarization-based MVTS representation (TS-SUM), Long-short term memory (LSTM), Recurrent Neural Network (RNN), and Random Convolutional Kernel Transform (ROCKET). As this is still a work in progress, all these 7 models are trained on the reduced dataset.

CHAPTER 2

RELATED WORK

Theo, [10], was among the first systems to predict flares. It was an expert system that required human input. It used a set of sunspots and magnetic field parameters to forecast different flare classifications. Rule-based flare prediction using Theo was adopted by National Oceanic and Atmospheric Administration's (NOAA) Space Environment Center (SEC) in 1987. The current methods of flare prediction are data-driven and are divided into two categories: linear statistical and nonlinear statistical. They can be further divided into line-of-sight magnetogram-based models and vector magnetogram-based models. The continuous stream of vector magnetograms is always considered a better means for parameterizing the Active Regions as they contain the full-disk magnetic field data as mentioned in [11]. However, it was not easily available before the launch of the Solar Dynamics Observatory (SDO) by the National Aeronautics and Space Administration (NASA) in 2010, and solar physicists had to depend on line-of-sight magnetic data for flare prediction.

Linear statistical studies aim to identify the AR magnetic properties that are correlated with flares. [12]) used line-of-sight magnetograms to parameterize Active Regions and studied the correlation between AR parameters and flare occurrences. From many SOHO/MDI longitudinal magnetograms, they evaluated three physical measures: the maximum horizontal gradient, the length of the neutral line, and the number of singular points. Properties of the photospheric magnetic field, such as non-potentiality and complexity, thought to be highly related to solar flares, have been identified using these evaluated measures. Their statistical analysis concluded that solar flare productivity increases with non-potentiality and complexity. In a similar study, using line-of-sight Michelson Doppler Imager (MDI) magnetograms of 89 active regions and Solar Geophysical Data (SGD) flare reports, [13] assessed the magnitude-scaling correlations between three parameters of magnetic fields and

the flare productivity of solar active regions. The mean value of spatial magnetic gradients at strong-gradient magnetic neutral lines (NL), the length of strong-gradient magnetic neutral lines (LGNL), and the total magnetic energy were the parameters studied. Active region MDI magnetograms used in their research were found to be relatively close to the solar central meridian. In particular, they revealed strong positive linkages between the parameters and both the total flare productivity of active regions and the potential of following flare production. Their findings confirmed the dependence of flare productivity on the degree of non-potentiality of active regions. [14] was the first to determine AR parameters from vector magnetograms. They conducted statistical tests based on discriminant analysis on a variety of photospheric magnetic parameters to identify those properties that are critical for the production of energetic events such as solar flares. They concluded that while the factors evaluated singly had minimal power to differentiate between flaring and flare-quiet groups, the populations could be separated using multi-variable combinations.

Nonlinear statistical models are commonly implemented using traditional machine learning classifiers. In the context of classification models, several approaches have been explored. [15] utilized a C4.5 decision tree, while [16] employed logistic regression. [17] opted for an artificial neural network, and [18] utilized a relevance vector machine. Additionally, [19] investigated the performance of three classifiers—k-NN, SVM, and Extremely Randomized Tree—utilizing both line-of-sight and vector magnetograms.

The pioneering work by [20] marked the first instance of employing machine learning algorithms on HMI vector magnetograms. They employed a Support Vector Machine (SVM) classifier, leveraging four years of data from the Solar Dynamics Observatory's (SDO) Helioseismic and Magnetic Imager (HMI) to forecast M- and X-class solar flares. Their work was groundbreaking as it involved a vast dataset of vector magnetograms for flare prediction. The authors curated a catalog of flaring and non-flaring active regions from a database containing 2071 active regions, comprising 1.5 million active region patches of vector magnetic

field data. Using 25 parameters, each active region was classified. Additionally, they employed a feature selection algorithm to identify the most effective features for distinguishing between flaring and non-flaring active zones. To address the class imbalance problem, they utilized a cost function to limit false negatives.

Efficiently addressing the solar flare prediction task, [21] framed solar flare classification as a binary classification problem, distinguishing between flaring and non-flaring Active Regions. They meticulously extracted time series samples of the Active Region parameters and developed a flare prediction method utilizing k-NN classification on univariate time series. Interestingly, their research revealed that employing a statistical summarization technique on a specific Active Region parameter known as "total unsigned current helicity" outperformed using all Active Region parameters at a single point in time. Furthermore, by exploring the time series properties of the AR parameters, the researchers identified the most influential parameter, thereby simplifying the problem to a single-variate time series classification. They proposed a novel approach of using a statistical summarization method on the time series, allowing the top AR parameter to serve as the vector-based representation of flaring/non-flaring Active Regions. By applying the k-nearest neighbors (k-NN) classifier within this reduced vector space, they achieved significant computational and time savings. Importantly, they also demonstrated that including C-class flares in the positive class did not improve classification performance.

Angryk et al. [22] presented a comprehensive multivariate time series (MVTS) dataset derived from solar photospheric vector magnetograms in the Space weather HMI Active Region Patch (SHARP) series. The dataset encompassed 4,098 MVTS data instances collected from active regions between May 2010 and December 2018. It included 51 flare-predictive parameters and over 10,000 flare reports. The dataset served as a valuable test bed for solar physicists and machine learning practitioners, providing a cleansed, integrated, and readily available dataset with data verified from multiple sources. The study incorporated data

from the GOES flare catalog, SSW and XRT flares, and NOAA AR locations to enhance, verify, and cleanse the dataset. The authors recalculated magnetic field parameters from individual region patches and transformed them into multivariate time series spanning the entire length of a given HARP series. They further addressed dataset cleaning, accounting for empty SHARPs, location-based filtering, and missing values. Subsequently, the dataset was partitioned into target classes using flare intensity threshold criteria, while observation window, latency, and prediction window concepts were utilized for custom slicing and labeling.

Ahmadzadeh et al. [1] discussed the challenges posed by the SWAN-SF dataset introduced by Angryk et al. [22]. They highlighted the extreme class imbalance within the data, as well as the temporal coherence. To tackle these issues, the researchers began by extracting statistical features from the time series, thereby reducing the dimensionality of the dataset. They employed an SVM machine learning classifier to conduct their experiments and addressed class imbalance through a combination of undersampling and oversampling techniques at the data level. Additionally, they fine-tuned the misclassification weighting parameter at the classifier level to minimize false positives and false negatives. To account for temporal coherence, they employed testing and training data from different partitions to prevent overlapping of data points.

The Transformer model, introduced by Vaswani et al. [8], offers several strengths, including its ability to capture long-range dependencies, handle parallel computation, and learn contextual relationships without relying on explicit sequential processing. In the context of multivariate time series (MVTS) classification, the benefits of the Transformer and self-attention mechanism can be utilized due to the sequential nature of the data.

CHAPTER 3

BENCHMARK DATASET

3.0.1 SWAN-SF

The benchmark dataset Space Weather ANalytics for Solar Flares (SWAN-SF) by Angryk et al. [22] serves as an illustrative example of a multivariate time series, aiming to achieve unbiased flare forecasting and classification. This dataset encompasses five distinct flare classes, ranging from the most powerful X-class and M-class flares to the smaller B-class and C-class flares. Additionally, it includes a non-flaring class denoted as the F class. In this paper, we refer to the flaring (M and X class flares) and non-flaring (F, B, and C class flares) as the positive and negative classes, respectively. To ensure temporal segmentation, the dataset has been divided into five partitions, each containing approximately equal proportions of X- and M-class flares (Figure 3.1).

The dataset comprises time series features derived from solar photospheric magnetograms, alongside NOAA's active region flare history. The Solar Dynamics Observatory's (Pesnell et al. [23]) HMI Active Region Patches (HARP) data product provides magnetograms (Hoeksema et al. [24]). While the magnetic field parameters are initially derived from the Space weather HMI Active Region Patches (SHARP) data product (Bobra et al. [4]), they were recalculated and augmented with additional parameters for validation purposes, including parameters not found in SHARPs (refer to Table 1 in Angryk et al. [22]). The dataset consists of sliding time series slices, with each instance representing 24 physical magnetic field parameters (see Table 3.1). These time series instances are logged at 12-minute intervals over a total of 12 hours (60-time steps). The instances of the multivariate time series are classified based on the most intense solar flare occurring within the preceding 12 hours.

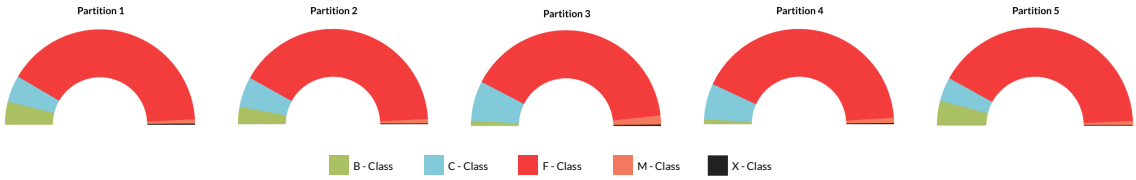

Fig. 3.1: Distribution of classes in five partitions

Each solar active region exhibits a different range of flare classes or maintains a region of tranquil activity within a prediction window. These changes are reflected in the representation of solar event i as $mvts\_i$, a multivariate time series instance, along with its corresponding class label, y_i. The term y_i characterizes the various flare classes. Comprising N magnetic field parameters, the multivariate time series instance $mvts_i \in R^{T*N}$ encompasses multiple time series with periodic observations over an interval of T. The t-th timestamp value is denoted as $x^{<t>} \in R^N$, while the j-th parameter time series is denoted as $P_j \in R^T$. The event is classified based on the active region's state after the observation time T and the subsequent prediction interval L. NOAA records of flare events are utilized to determine the state of a given timestamp.

When the population of one or more data classes is significantly smaller than the majority classes, the dataset is considered class imbalanced. The minority classes consist of data points from the smaller group, while the other group is referred to as the majority classes. Figure 3.1 illustrates the substantial class imbalance present in the SWAN-SF dataset. Traditional machine learning classifiers tend to favor the majority class, as highlighted by [25]. This becomes especially concerning in solar flare classification, where the focus lies on a minority of cases. Class imbalance significantly impacts various performance metrics, including accuracy, precision, and the F1 score. This is due to the metrics disregarding the number of misclassifications. For instance, a model that assigns all instances to the majority class may achieve high accuracy while learning very little about the minority class. In

the following sections, we discuss TSS and HSS2 evaluation metrics, which are commonly employed in such class imbalance scenarios to measure model performance.

Table 3.1: List of AR magnetic field parameters

Abbreviation	Description	Formula
ABSNJZH [14]	Absolute value of the net current helicity	$H_{c_{abs}} \propto \lvert\sum B_z \cdot J_z\rvert$
EPSX [26]	Sum of x-component of normalized Lorentz force	$\delta F_x \propto \frac{\sum B_x B_z}{\sum B^2}$
EPSY [26]	Sum of y-component of normalized Lorentz force	$\delta F_y \propto \frac{-\sum B_y B_z}{\sum B^2}$
EPSZ [26]	Sum of z-component of normalized Lorentz force	$\delta F_z \propto \frac{\sum\left(B_x^2+B_y^2-B_z^2\right)}{\sum B^2}$
MEANALP [27]	Mean characteristic twist parameter, α	$\alpha_{\text{total}} \propto \frac{\sum J_z \cdot B_z}{\sum B_z^2}$
MEANGAM [14]	Mean angle of field from radial	$\overline{\gamma} = \frac{1}{N}\sum \arctan\left(\frac{B_h}{B_z}\right)$
MEANGBH [14]	Mean gradient of horizontal field	$\overline{\lvert\nabla B_h\rvert} = \frac{1}{N}\sum\sqrt{\left(\frac{\partial B_h}{\partial x}\right)^2+\left(\frac{\partial B_h}{\partial y}\right)^2}$
MEANGBT [14]	Mean gradient of total field	$\overline{\lvert\nabla B_{tot}\rvert} = \frac{1}{N}\sum\sqrt{\left(\frac{\partial B}{\partial x}\right)^2+\left(\frac{\partial B}{\partial y}\right)^2}$
MEANGBZ [14]	Mean gradient of vertical field	$\overline{\lvert\nabla B_z\rvert} = \frac{1}{N}\sum\sqrt{\left(\frac{\partial B_z}{\partial x}\right)^2+\left(\frac{\partial B_z}{\partial y}\right)^2}$
MEANJZD [14]	Mean vertical current density	$\overline{J_z} \propto \frac{1}{N}\sum\left(\frac{\partial B_y}{\partial x}-\frac{\partial B_x}{\partial y}\right)$
MEANJZH [14]	Mean current helicity (B_z contribution)	$\overline{H_c} \propto \frac{1}{N}\sum B_z \cdot J_z$
MEANPOT [28]	Mean photospheric magnetic free energy	$\overline{\rho} \propto \frac{1}{N}\sum\left(\boldsymbol{B}^{Obs}-\boldsymbol{B}^{Pot}\right)^2$
MEANSHR [28]	Mean shear angle	$\overline{\Gamma} = \frac{1}{N}\sum \arccos\left(\frac{\boldsymbol{B}^{Obs}\cdot\boldsymbol{B}^{\text{Pot}}}{\lvert B^{Obs}\rvert\lvert B^{Pot}\rvert}\right)$
R_VALUE [29]	Sum of flux near polarity inversion line	$\Phi = \Sigma\,\lvert B_{\text{LoS}}\rvert\,dA$ (within R mask)
SAVNCPP [14]	Sum of the modulus of the net current per polarity	$J_{z_{sum}} \propto \lvert\sum^{B_z^+} J_z dA\rvert + \lvert\sum^{B_z^-} J_z dA\rvert$
SHRGT45 [14]	Fraction of Area with shear $> 45°$	Area with shear $> 45°$/ total area
TOTBSQ [26]	Total magnitude of Lorentz force	$F \propto \sum B^2$
TOTFX [26]	Sum of x-component of Lorentz force	$F_x \propto -\sum B_x B_z dA$
TOTFY [26]	Sum of y-component of Lorentz force	$F_y \propto \sum B_y B_z dA$
TOTFZ [26]	Sum of z-component of Lorentz force	$F_z \propto \sum\left(B_x^2+B_y^2-B_z^2\right)dA$
TOTPOT [14]	Total photospheric magnetic free energy density	$\rho_{tot} \propto \sum(\boldsymbol{B}^{Obs}-\boldsymbol{B}^{Pot})^2 dA$
TOTUSJH [14]	Total unsigned current helicity	$H_{c_{\text{total}}} \propto \sum B_z \cdot J_z$
TOTUSJZ [14]	Total unsigned vertical current	$J_{z_{total}} = \sum \lvert J_z\rvert\,dA$
USFLUX [14]	Total unsigned flux	$\Phi = \sum \lvert B_z\rvert\,dA$

3.0.2 Reduced dataset

The following dataset is one of the benchmark datasets for MVTS-based solar flare prediction published by Angryk et al. [22]. This dataset consists of multiple MVTS instances, with each instance comprising 25-time series of active region magnetic field parameters (a comprehensive list can be found in Table 1). The time series instances are recorded at 12-minute intervals, spanning a total duration of 12 hours (60-time steps). The dataset is characterized by having 60 observation points (T) and 25 parameters (N). Our experimen-

tal dataset consists of 1,540 MVTS instances, which are evenly distributed across four flare classes: X, M, BC, and Q. Here, "Q" represents flare-quiet events, and "BC" represents a mixture of B and C class events.

CHAPTER 4

METHODOLOGY

4.0.1 MINIROCKET

While machine learning and deep learning-based classifiers for time series classification have achieved impressive levels of accuracy, they often suffer from high computational complexity. This drawback becomes particularly problematic for larger datasets, resulting in longer training times and rendering them impractical. Moreover, many existing techniques focus on specific aspects of the data, such as shape or frequency, neglecting the broader picture. To address these challenges, [6] introduced the RandOm Convolutional KErnels Transform (ROCKET) method. This novel approach leverages the success of convolutional neural networks in time series classification by utilizing random convolutional kernels to extract informative features. These features are then used to train a linear classifier. To further enhance efficiency, [5] proposed a modified version called the MINImally RandOm Convolutional KErnels Transform (MINIROCKET), which achieves faster execution and nearly deterministic behavior.

The ROCKET method transforms time series data by convolving each series with a set of random convolutional kernels. These kernels, similar to those in convolutional neural networks, possess random characteristics such as length, weights, bias, dilation, and padding. They capture a wide range of information and patterns at various frequencies and scales. The output of each kernel undergoes two pooling techniques: global max pooling and percentage of positive values (PPV) pooling. Global max pooling selects the maximum feature value, while PPV pooling evaluates the prevalence of a pattern captured by the kernel. $ppv = 1/n \sum_{i=0}^{n-1} [z_i > 0]$, where z_i is the output of the convolution operation. The fraction of positive values derived from PPV pooling plays a vital role in assessing the

significance of the captured patterns, contributing to the method's high precision. Each kernel produces two features, resulting in a total of 20,000 features per input time series when employing 10,000 random convolutional kernels. These extracted features are then utilized for training a linear classifier.

Hyper-parameters	ROCKET	MINIROCKET
Length	{7, 8, 11}	9
Weights	N(0, 1)	{-1, 2}
Bias	U(-1, 1)	From convolution output
Dilation	Random	Fixed
Padding	Random	Fixed

Fig. 4.1: Difference between ROCKET and MINIROCKET Kernel's hyper-parameters

Both the MINIROCKET and ROCKET methods rely on PPV pooling to assess the convolution values. MINIROCKET further optimizes computational efficiency by employing a fixed set of kernels with specific hyper-parameter settings, refer to figure 4.1. Key modifications include fixing the kernel length at 9, limiting the weight hyper-parameter to a fixed range, adapting the bias hyper-parameter to random convolutional output values, restricting the dilation hyper-parameter, and employing only PPV pooling instead of global max pooling and PPV. These optimizations enable MINIROCKET to generate half as many features as ROCKET while maintaining equivalent precision. MINIROCKET achieves remarkable computational efficiency through a combination of the aforementioned optimizations. It utilizes the mathematical properties of fixed kernels and PPV pooling to compute PPV for both positive and negative weights simultaneously, effectively doubling the number of applied kernels without increasing computations. It also maximizes the reuse of convolution output and avoids multiplications by employing additive operations. Addi-

tionally, MINIROCKET computes all kernels for each dilation at once, further optimizing computation and output reuse. These optimizations significantly improve computational efficiency while preserving the accuracy level of the ROCKET classifier.

In our experiments with the SWAN-SF dataset, MINIROCKET demonstrated superior performance compared to other classifiers. Its computational efficiency and accuracy make it an excellent choice for time series classification tasks. In the following section, we will discuss the outcomes of our experimentation and provide a concise overview of the remaining classifiers evaluated.

4.0.2 TRANSFORMERS

Notations

The solar event instance i is represented by an MVTS instance $mvts_i$. The MVTS instance $mvts_i \in^{T \times N}$ is a collection of individual time series of N magnetic field parameters, where each time series contains periodic observation values of the corresponding parameter for an observation period T. In the MVTS instance $mvts_i = \{v_{t_1}, v_{t_2}, ., ., ., v_{t_T}\}$, where $v_{t_i} \in^N$ represents a timestamp vector.

Data Preprocessing and Normalization

The magnetic field parameter values are recorded in different scales, so we perform z-score normalization. Z-score normalization is a technique used to transform data in such a way that it possesses a mean of zero and a standard deviation of one. By employing this method, we can effectively assess and compare the relative significance of various features within our dataset. Suppose that M number of MVTS instances each with N parameters and T time points are represented by a third-order tensor $\mathcal{X} \in^{M \times N \times T}$, where three modes represent events, parameters, and timestamps. We perform *parameter-level* z-normalization

as a preprocessing step in the following three steps.

1. We perform mode-2 metrication, i.e., reshaping the tensor so that mode-2 entities (parameter) become the columns of the matrix. The matrix is denoted by $X_{(2)} \in^{MT \times N}$. The columns are denoted by $P_1, P_2, \ldots, P_N$.

2. For each column P_j, we perform z-normalization:

$$x_k^{(j)} = \frac{x_k^{(j)} - \mu^{(j)}}{\sigma^{(j)}}$$

   Here, $x_k^{(j)}$ is the k-th value of the column P_j, where $1 \leq k \leq MT$, $\mu^{(j)}$ is the mean of the column P_j, and $\sigma^{(j)}$ is the standard deviation of the column P_j.

3. We reshape the matrix $X_{(2)} \in^{MT \times N}$ back to third-order tensor, $\mathcal{X} \in^{M \times N \times T}$.

Attention-based MVTS Classification Framework

In this study, we use the Transformer/Attention-based model to get better performance in classifying the MVTS solar flare dataset. In our model, we create the transformer encoder block as figure 4.2 shows. In [8] the authors proposed a model architecture called the Transformer. The Transformer consists of an encoder and a decoder, both of which are composed of multiple layers of self-attention and feed-forward neural networks. The encoder processes the input sequence, such as a sentence in machine translation. It consists of a stack of identical layers, where each layer has two sub-layers:

- Self-Attention Layer: This layer performs self-attention, allowing each word in the input sequence to attend to all other words in the same sequence. It captures dependencies between words and generates contextualized representations for each word.

- Feed-Forward Neural Network Layer: After self-attention, a feed-forward neural network layer is applied to each word representation independently. It introduces non-linearity and enables the model to incorporate additional information.

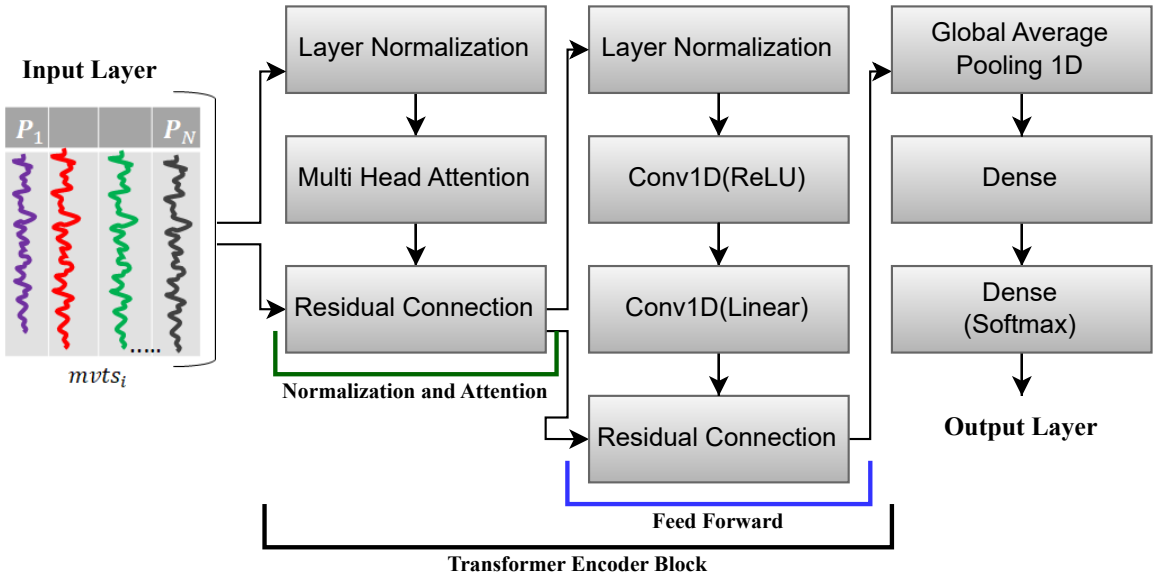

Fig. 4.2: Transformer/Attention Model for MVTS Classification

Algorithm 1 MVTS Transformer Encoder

function TRANSFORMER_ENCODER($inputs$, $head\_size$, num_heads, ff_dim)

  $x \leftarrow \text{LAYERNORMALIZATION}(inputs, \epsilon = 1e-6)$

  $x \leftarrow \text{MULTIHEADATTENTION}(x, x, key\_dim = head\_size, num\_heads = num\_heads)$

  $res \leftarrow x + inputs$

  $x \leftarrow \text{LAYERNORMALIZATION}(res, \epsilon = 1e-6)$

  $x \leftarrow \text{CONV1D}(x, filters = ff\_dim, kernel\_size = 1, activation = \text{"}relu\text{"})$

  $x \leftarrow \text{CONV1D}(x, filters = inputs.shape[-1], kernel\_size = 1)$

  return $x + res$

end function

Algorithm 2 Build MVTS Transformer Model

function BUILD_TRANSFORMER_MODEL($input\_shape$, $head\_size, num\_heads, ff\_dim$, $num\_transformer\_blocks$, mlp_units)
    $n\_classes \leftarrow$ LENGTH($unique\_y\_train$)
    $inputs \leftarrow$ INPUT($shape = input\_shape$)
    $x \leftarrow inputs$
    for $i \leftarrow 1$ **to** $num\_transformer\_blocks$ **do**
        $x \leftarrow$ TRANSFORMER_ENCODER(x, $head\_size$, num_heads, ff_dim)
    end for
    $x \leftarrow$ GLOBALAVERAGEPOOLING1D(x, $data\_format = "channels\_first"$)
    for dim **in** mlp_units **do**
        $x \leftarrow$ DENSE($x, dim, activation = "relu"$)
    end for
    $outputs \leftarrow$ DENSE($x, n\_classes, activation = "softmax"$)
    return MODEL($inputs, outputs$)
end function

In our model, we utilize the transformer encoder block and use the benefits of the Multi-Head Attention architecture which is a crucial component of the Transformer model. It allows the model to focus on different parts of the input sequence simultaneously, enhancing its ability to capture complex temporal dependencies and extract relevant features. By using multiple attention heads, the model can learn different representations and attend to different aspects of the input data in parallel. In the context of MVTS data classification, Multi-Head Attention offers several advantages:

- Enhanced Representational Capacity: By attending to different parts of the input sequence simultaneously, Multi-Head Attention allows the model to capture both local and global dependencies effectively. This enables the model to learn complex patterns within the time series data, leading to improved classification performance.

- Robustness to Variable-Length Sequences: MVTS data often consists of sequences with varying lengths. Multi-Head Attention can handle variable-length sequences efficiently by assigning different attention weights to different parts of the input. This flexibility enables the model to adapt to sequences of different lengths without compromising its classification accuracy.

The Multi-Head Attention mechanism is a crucial component of the Transformer model, allowing for the simultaneous capture of different aspects of the input sequence. It involves the computation of multiple attention heads in parallel, enabling the model to effectively process diverse information. The equations governing Multi-Head Attention are as follows:

- Scaled Dot-Product Attention:
  $Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$
  Here, Q, K, and V denote input matrices representing queries, keys, and values, respectively. The dimension of the key and query vectors is denoted by d_k. The attention mechanism computes the weighted sum of values based on the similarity between queries (Q) and keys (K). Linear transformations are applied to the queries, keys, and values before calculating attention weights through the dot-product operation. The softmax function normalizes these weights, and the resulting weights are used to weigh the corresponding values (V) to produce the final output.

- Multi-Head Attention:
  $MultiHead(Q, K, V) = Concatenate(head_1, head_2, .., head_h)W^O$
  where $head_i$ = Attention(QW_i^Q, KW_i^K, VW_i^V). The matrices W_i^Q, W_i^K, and W_i^V represent learnable linear transformation matrices specific to the i-th attention head, while W^O is the linear transformation matrix applied to the concatenated heads. In this step, the input matrices Q, K, and V are linearly transformed separately for each attention head. The Attention function is then applied to obtain the attention outputs for each head. These attention heads are again concatenated and transformed

by matrix W^O to produce the final output of the Multi-Head Attention layer [8].

Our model diagram, as shown in Figure 4.2, can be described as follows. The inputs undergo a series of Transformer Encoder Blocks. Each Transformer Encoder Block comprises a normalization and attention step, followed by a feed-forward step. For each Transformer Encoder Block, the input data is passed through a Layer Normalization (Encoder) block to normalize the inputs. The normalized inputs are then fed into the MultiHead Attention layer, which applies self-attention to capture dependencies between different parts of the input sequence. The output of the MultiHead Attention layer is combined with the original inputs using a Residual Sum operation, preserving the original information. The result is further passed through another Layer Normalization (Encoder) block. The output of the Layer Normalization block is fed into a 1D Convolutional layer with ReLU activation, enabling the capture of local patterns and non-linear relationships in the data. This output then passes through another 1D Convolutional layer. The output of the second 1D Convolutional layer is once again combined with the previous output using a Residual Sum operation. The final output of the transformer encoder function is obtained by summing the previous output with the input data, representing the transformed representation of the inputs. The Transformer Encoder Blocks are repeated multiple times according to the specified parameter.

After the last transformer encoder block, the output is fed into a Global Average Pooling 1D layer to aggregate the features across the time dimension. Subsequently, the output of the Global Average Pooling 1D layer is passed through a series of Dense layers with *ReLU* activation, as determined by the *mlp_units* parameter. The final Dense layer generates the model's outputs, with the number of units corresponding to the number of output classes, and employs the softmax activation function. These outputs represent the predictions made by the model. Algorithm 1 operates as follows:

1. Layer Normalization: The input tensor is first normalized along each feature dimension by passing it through a layer normalization layer.

2. Self-Attention: The normalized tensor is then fed into a multi-head attention layer, where a self-attention mechanism is applied. Each attention head attends to different parts of the input sequence and learns to capture distinct relationships between time steps. The output of the attention layer retains the same shape as the input.

3. Residual Connection: The output of the multi-head attention layer is element-wise added to the original input tensor (inputs). This residual connection facilitates the direct flow of gradients from the input to the output, easing the learning process for the model.

4. Feed-forward: The result of the residual connection is passed through another layer normalization layer.

5. Convolutional Layer: A 1D convolutional layer with ff_dim filters and kernel size 1 is applied to the normalized tensor. This layer acts as a feed-forward neural network layer, applying non-linear transformations independently to each position in the sequence.

6. Second Convolutional Layer: Another 1D convolutional layer with inputs. shape[-1] filters and kernel size 1 is applied to the result obtained from the previous layer.

7. Residual Connection: The output of the second convolutional layer is element-wise added to the result obtained from the first residual connection layer.

8. Final Output: The sum of the previous residual connection and the original input tensor (inputs) is returned as the final output.

Algorithm 2 incorporates several parameters, each described as follows: $input\_shape$ specifies the shape of the input data, $head\_size$ determines the size of each attention

head in the transformer, num_heads denotes the number of attention heads in the transformer, ff_dim represents the dimension of the feed-forward network in the transformer, $num\_transformer\_blocks$ indicates the number of transformer blocks to be stacked, and mlp_units is a list of integers specifying the number of units in each MLP layer. Within the algorithm, it first determines the number of classes ($n\_classes$) based on the unique labels present in the training data. It then defines the input layer and sets it as the current layer, denoted as x. The algorithm proceeds by applying the transformer encoder block through the $transformer\_encoder$ function. After the transformer encoder blocks, a global average pooling layer is applied to reduce the spatial dimensions of the data. Subsequently, a series of MLP layers are implemented as specified by the mlp_units parameter, with each layer employing $ReLU$ activation. Finally, an output layer is added with $n\_classes$ units and a $softmax$ activation function for classification. The algorithm creates and returns a $Keras$ model with the defined inputs and outputs.

CHAPTER 5

EXPERIMENTS

In this section, we present an overview of the various models that we investigated and compare them to the state-of-the-art MINIROCKET. The study was conducted by evaluating the performance of each model under different data configurations and comparing the results. To ensure the validity of the results, one partition was used for training, and the remaining four partitions were utilized for testing. For instance, we employed partition 1 to train the model, while partitions 2, 3, 4, and 5 were used individually for testing. This strategic approach yielded a total of 20 distinct partition pairs. This setting was chosen to align with the methodology used by Ahmadzadeh et al. [1] to prevent data overlap and remedy temporal coherence. The performance of the models was evaluated using the True Skill Statistic (TSS) score and Heidke Skill Score (HSS2), the two most frequently used metrics for flare prediction in class imbalance data.

5.0.1 Performance metrics: TSS Score and HSS2 Score

An effective method of evaluating the efficacy of a classifier is by comparing its performance against a designated benchmark. This comparison can be accomplished through the computation of a skill score. Typically, this score is expressed as the difference between the score value and that of a standard forecast, divided by the difference between a perfect score and the standard forecast. In the case of solar flare prediction, the development of such a skill score is logical, considering that the number of non-flaring regions is considerably greater than that of flaring ones. To assess the performance of various classifiers on the SWAN-SF dataset for flare prediction, we relied on the use of forecast verification metrics, with a focus on the True Skill Statistic (TSS) and Heidke Skill Score (HSS2).

TSS and HSS2 are calculated based on the model's confusion matrix, which depicts the frequencies of predicted and actual values. The term "TN" represents True Negatives, reflecting the accurate classification of negative examples. Correspondingly, "TP" signifies True Positives, indicating the correct classification of positive examples. On the other hand, "FP" denotes False Positives, representing the number of actual negative examples mistakenly classified as positive. Finally, "FN" represents False Negatives, indicating the number of actual positive examples erroneously classified as negative. An example of a confusion matrix for binary classification is shown in Figure 5.1.

	ACTUAL POSITIVE	ACTUAL NEGATIVE
PREDICTED POSITIVE	TRUE POSITIVE	FALSE POSITIVE
PREDICTED NEGATIVE	FALSE NEGATIVE	TRUE NEGATIVE

Fig. 5.1: Confusion matrix for binary classification

[30] utilize a definition of the Heidke Skill Score (HSS) proposed by the Space Weather Prediction Center, denoted as HSS2, which quantifies the enhancement of the forecast compared to a random forecast. HSS2 is calculated using the following formula:

$$HSS_2 = \frac{TP + TN - E}{P + N - E} \tag{5.1}$$

Where E represents the anticipated number of correct predictions that can be corresponded to chance:

$$E = \frac{(TP + FP) * (TP + FN) + (FP + TN) * (FN + TN)}{P + N} \tag{5.2}$$

Alternatively, HSS2 can be derived from the true positive (TP), true negative (TN), false negative (FN), and false positive (FP) classification outcomes, in addition to the total number of positive (P) and negative (N) instances:

$$HSS_2 = \frac{2 * [(TP * TN) - (FN * FP)]}{P * (FN + TN) + (TP + FP) * N} \tag{5.3}$$

While HSS2 may be influenced by the class-imbalance ratio of the testing set, TSS has been recommended by [31] as a more suitable metric in these cases, as it is known to be unbiased with respect to the class-imbalance ratio and is considered to be more equitable. The TSS is defined as-

$$TSS = \frac{TP * TN - FP * FN}{P * N} = \frac{TP}{TP + FN} - \frac{FP}{FP + TN} \tag{5.4}$$

also known as the Hansen-Kuipers skill score or Peirce skill score ([32], measures the difference between the recall and false alarm rate, and ranges from -1 to 1, with a score of 1 indicating a perfect forecast, a score of 0 representing a random or constant forecast, and a score of -1 indicating a forecast that is always incorrect. The TSS is considered a desirable metric for comparing the performance of various classifiers for solar flare forecasting, as it takes into account both false negatives and false positives in a balanced manner, and is not affected by the imbalance of the testing set.

A potential limitation of the True Skill Statistic (TSS) is that it considers false positive (FP) and false negative (FN) predictions as having equal weight, even though the consequences of these misclassifications can vary. In the context of forecasting solar flares, the cost of a false negative (not predicting a flare that occurs) can be higher than the cost of a false positive (predicting a flare that does not occur), such as in the scenario of a satellite that needs to be rotated to protect against an increase in energetic particles. The costs of false positives and false negatives are not symmetrical. The TSS is insensitive to the imbalance ratio of the testing set, whereas the Heidke Skill Score (HSS2) can be influenced by this ratio and converge to zero as the ratio increases.

5.0.2 Comparing different classes of classifiers

We evaluated the effectiveness of several time series classifiers, including LSTM, SVM, MRSEQL, and CIF, based on their TSS and HSS2 scores. By comparing the performance of these models, our analysis reveals that MINIROCKET achieves the highest accuracy and

outperforms the other classifiers in binary classification and all-class classification. This study highlights the potential of MINIROCKET as a powerful tool for flare classification. In the following sections, we provide a brief overview of each classifier and later compare the results obtained.

Long Short-Term Memory (LSTM)

We utilized Long Short-Term Memory (LSTM) networks in this research to learn representations of Multivariate Time Series (MVTS) instances without relying on hand-engineered statistical characteristics. The LSTM network was trained by sequentially feeding magnetic field parameter vectors into LSTM cells. Cell weights were optimized using gradient descent and backpropagation. The model effectively identified underlying patterns in the data and produced reliable predictions for flare occurrences through automated feature learning. LSTM networks excel in processing and categorizing time-series data due to their ability to capture order dependence and long-term dependencies that regular RNNs cannot. Additionally, deep LSTM networks, created by stacking multiple LSTM layers, can learn even more complex patterns in sequential data. The usage of LSTM networks in this study showcases their usefulness in learning time series data representations and their potential for various domains.

Support Vector Machine (SVM)

The support vector machine (SVM) classifier aims to identify a hyperplane in N-dimensional space that can accurately classify input points. The search for an ideal hyperplane involves finding a plane with the greatest margin, which represents the maximum distance between data instances of different classes. This margin is crucial as it enables effective generalization and improves prediction accuracy. Hyperplanes act as decision boundaries, separating data points, and their size is determined by the number of features in the data. Support vectors, which are data instances closest to the hyperplane, greatly influence

its orientation and placement. They play a critical role in optimizing the classifier's margin. By utilizing these support vectors, the SVM classifier finds the best hyperplane and achieves high prediction accuracy.

In the case of class imbalance in the flare dataset, the optimal hyperplane that intersects the decision boundary is pushed further toward the domain of the minority class. This adjustment aims to minimize the overall number of incorrect classifications, leading to an increase in true negatives (i.e., accurate classification of CBF-class flares) and a decrease in true positives (i.e., accurate classification of XM-class flares). In a class-imbalanced setting, models tend to exhibit a bias towards the majority class, which becomes concerning as the focus of flare-forecasting research lies on minority instances rather than the majority. The SVM classifier has gained popularity due to its ability to efficiently learn nonlinear decision surfaces, facilitated by support vectors and transformation functions (kernels). Various kernels can be utilized to enhance the transformation of data into new feature spaces, allowing for a more accurate separation of instances. Kernels require the specification of one or more variables in advance, similar to any other function.

Canonical Interval Forest (CIF)

The time series forest (TSF) classifier, known for its high performance, quick training, and prediction, is commonly regarded as a powerful interval method. However, it has fallen behind in recent advancements in alternative techniques. TSF initially summarized intervals using only three basic summary statistics. In recent developments, the 'catch22' feature set ([33]) was engineered as a concise and useful collection of 22-time series features to facilitate extensive time series analysis. Building upon these advancements, the Canonical Interval Forest (CIF) classifier, proposed by [34], combines the capabilities of both TSF and catch22. The CIF classifier aims to enhance performance and accuracy in time series analysis by leveraging the unique strengths of both techniques.

Multiple Representations SEQuence Learner (MRSEQL)

MRSEQL, proposed by [35], is a robust univariate time series classifier that trains on features derived from multiple symbolic representations of time series. These representations include Symbol Aggregation Approximation (SAX) and Symbol Fourier Approximation (SFA), which are used with linear classification models (logistic regression). Mr-SEQL utilizes SEQL ([36]) to extract features based on three key ideas. Firstly, instead of relying on a single fixed representation, Mr-SEQL combines multiple symbolic representations obtained from various parameters, such as multiple SAX representations. Secondly, it incorporates numerous domain representations in time (such as SAX) and frequency (such as SFA), making it resilient across a wide range of problems. Finally, Mr-SEQL extends a symbolic sequence classifier (SEQL) to effectively explore the significant symbolic-words space, employing an efficient greedy feature selection technique to find optimal features for each representation. Mr-SEQL is a highly effective time series classifier with important qualities that make it well-suited for a wide range of applications.

5.0.3 Binary Classification

In the preliminary experiments, we executed a transformation of the original data labels into binary labels to simplify the classification process. The positive class, denoted as flaring, encompasses M and X class flares, while the negative class, referred to as non-flaring, encompasses F, B, and C class flares.

To evaluate the effectiveness of the classification models, we trained five different models, namely MINIROCKET, CLF, MR-SEQL, LSTM, and SVM, and compared their performances in terms of TSS and HSS2 scores. We present the results of our experiments in the line plots displayed in Figure 5.2 and 5.3. These plots highlight the obtained scores for TSS and HSS2, respectively.

Our analysis has demonstrated that the MINIROCKET classifier outperformed other

Fig. 5.2: TSS score comparison of 5 different models

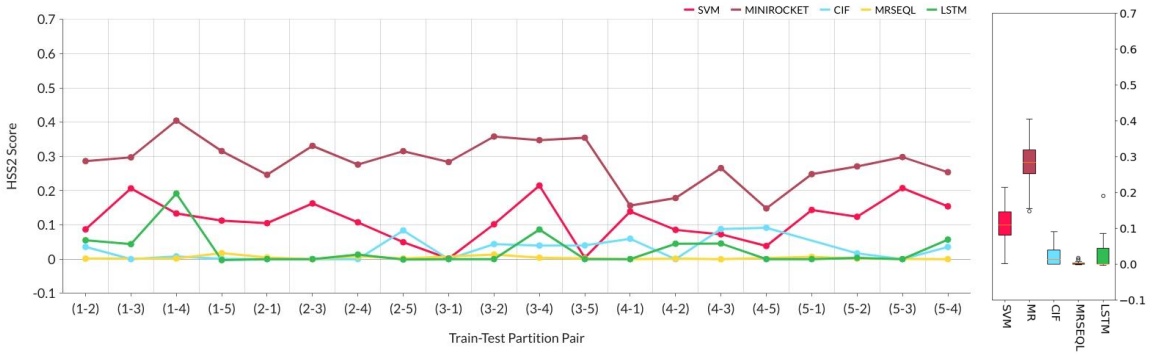

Fig. 5.3: HSS2 score comparison of 5 different models

classifiers on the SWAN-SF dataset, with an average improvement of 19.4% and 23.9% in the TSS and HSS2 scores, respectively.

5.0.4 Multi-class: All Class Classification

At this stage, we are engaged in the task of classifying the five distinct categories, namely F, B, C, M, and X. The experimental settings remain constant, wherein the training and testing are performed on 20 unique partition pairs, and the performance of the selected models, MINIROCKET, and SVM, are compared based on the TSS and HSS2 scores. Refer to the line plots displayed in Figure 5.4 and 5.5 for TSS and HSS2 score comparison.

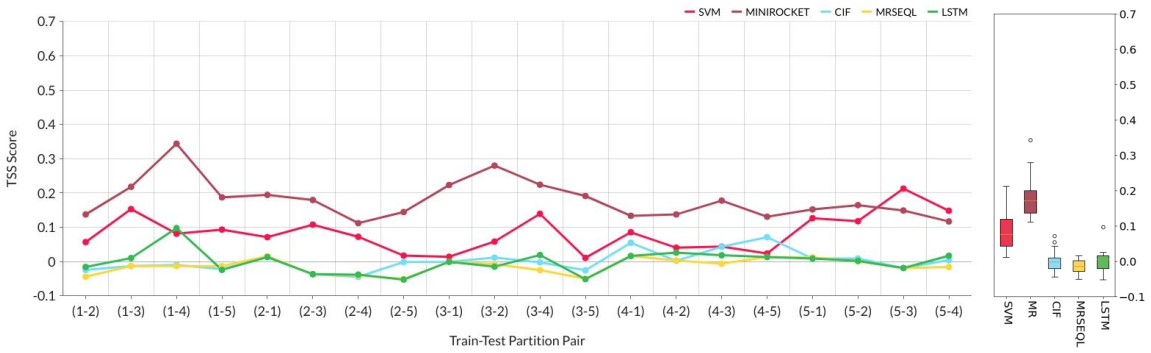

Fig. 5.4: TSS score comparison for all class classification

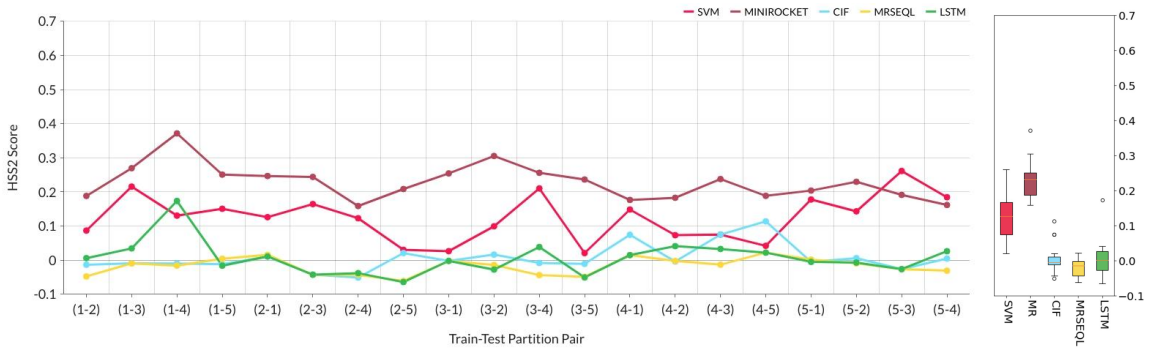

Fig. 5.5: HSS2 score comparison for all class classification

Our analysis of the all-class classification showed that the MINIROCKET classifier outperformed the other classifiers with a 9.61% higher TSS score and 10.36% higher HSS2 score.

5.0.5 Analysis with the exclusion of B and C class flares

In this phase of the experiment, B- and C-class flares would be excluded. This decision was made based on the research conducted by [31], which suggested that the inclusion of C-class flares may have a negative impact on performance metrics. In our analysis, we observed an improvement in the TSS score for all models after the removal of B and C-class flares. This underscores the significance of this exclusion in achieving optimal model performance.

Following the exclusion of B and C-class flares, the experiment was further divided

into two categories: binary class classification (Refer Figure 5.6 and Figure 5.7) and all-class classification (Refer Figure 5.8 and Figure 5.9).

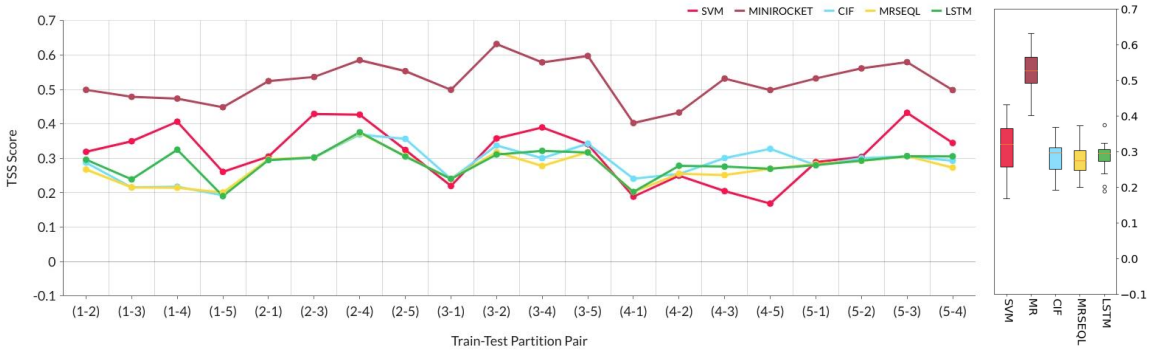

Fig. 5.6: TSS score comparison of binary class classification after removing B and C class flares

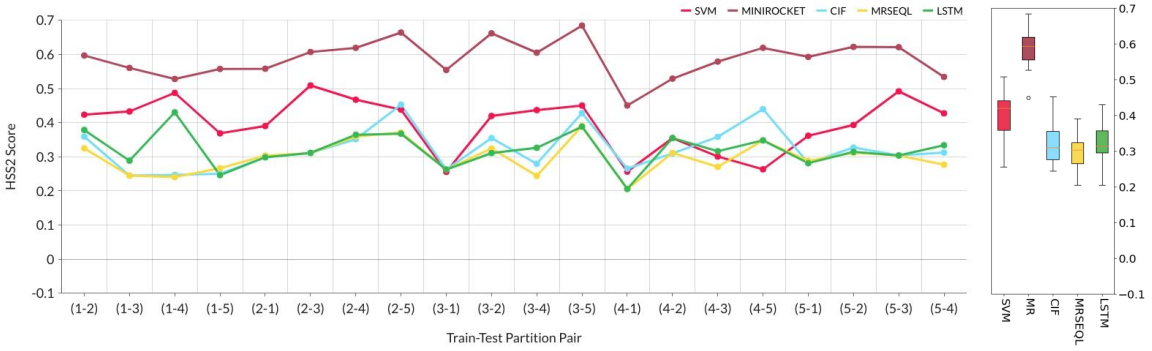

Fig. 5.7: HSS2 score comparison of binary class classification after removing B and C class flares

After removing the B and C class flares, our analysis for binary classification showed that MINIROCKET achieved a remarkable 30.06% increase in the TSS score and a 30.55% increase in the HSS2 score compared to other classifiers.

After removing the B and C class flares, we also analyzed the all-class classifications and observed that MINIROCKET again outperformed other classifiers by 20.13% in terms

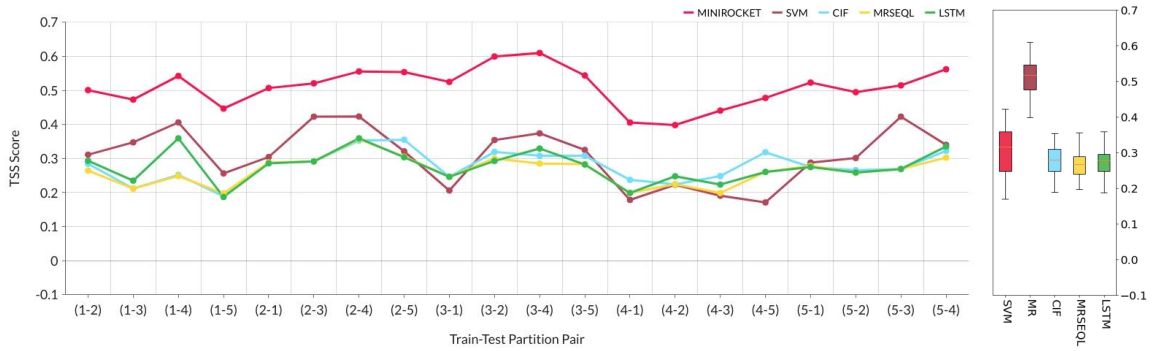

Fig. 5.8: TSS score comparison of all class classification after removing B and C class flares

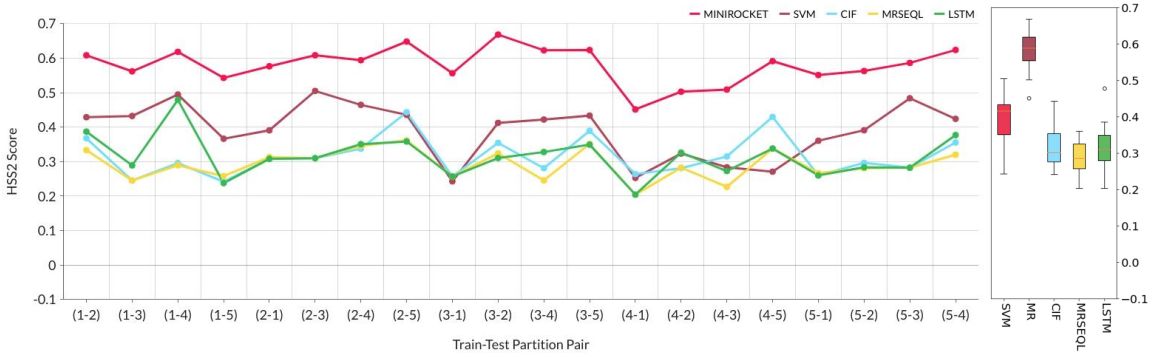

Fig. 5.9: HSS2 score comparison of all class classification after removing B and C class flares

of TSS score and 18.94% in terms of HSS2 score.

5.0.6 Transformer Model Experiment

In this section, we present our experimental findings, where we compare the performance of our model with six other MVTS-based flare prediction baselines using a benchmark dataset. We implemented our Attention/Transformer-based MVTS classifier using TensorFlow and the reduced dataset.

Dataset Description

For our experiments, we utilized the benchmark dataset for MVTS-based solar flare prediction published by Angryk et al. [22]. This dataset consists of multiple MVTS instances, with each instance comprising 25-time series of active region magnetic field parameters (a

comprehensive list can be found in Table 1). The time series instances are recorded at 12-minute intervals, spanning a total duration of 12 hours (60-time steps). The dataset is characterized by having 60 observation points (T) and 25 parameters (N). Our experimental dataset consists of 1,540 MVTS instances, which are evenly distributed across four flare classes: X, M, BC, and Q. Here, "Q" represents flare-quiet events, and "BC" represents a mixture of B and C class events.

Baseline Models

We evaluated our model with six other baselines.

1. **Flattened vector method (FLT):** This is a naive method, where each 60 $\times$ 25 MVTS instance is flattened into a 1, 500- dimensional vector.

2. **Vector of last timestamp (LTV):** This method was introduced by Bobra et al [20], where vector magnetogram data (feature space of all magnetic field parameters) were used for classification. Since the last timestamp of the MVTS is temporally nearest to the flaring event, we sampled the vector of the last timestamp (25 dimensional) to train the classifier.

3. **Time series summarization-based MVTS representation (TS-SUM):** This method, proposed by Hamdi et al. [21] summarizes each time series of length T by eight statistical features: mean, standard deviation, skewness, and kurtosis of the original time series, and the first-order derivative of the time series. As a result, we get an 8 $\times$ 25-dimensional vector space, which is used for training the downstream classifier.

4. **Long-short term memory (LSTM):** This LSTM-based approach was proposed by Muzaheed et. al. [37]. Each MVTS instance was considered as a T -length sequence of $x^{<t>} \in^N$ timestamp vectors. After sequentially feeding the LSTM model with each timestamp vector, the last hidden representation was considered as the MVTS

representation. In our experiments, we set the number of cell state and hidden state dimensions to 128, the number of training epochs to 500, and the learning rate in stochastic gradient descent to 0.01.

5. **Recurrent Neural Network (RNN):** As the fifth baseline, we replace LSTM cells of the model of [37] with standard RNN cells. We use the number of RNN hidden dimensions as 128, the number of training epochs as 1,000, and the learning rate in stochastic gradient descent as 0.01.

6. **Random Convolutional Kernel Transform (ROCKET):** ROCKET was shown as the best-performing algorithm in the MVTS classification benchmarking study by Ruiz et al [38], which included 26 MVTS datasets of the UEA archive [39]. ROCKET uses a large number of random convolution kernels along with a linear classifier, where each kernel is applied to each univariate time series instance. In line with the experimental setting of Ruiz et al. [38], we set the number of kernels in ROCKET to 10,000.

The first three baselines involve embedding followed by classification methods. We use a logistic regression classifier with L2 regularization for classification. In all the baseline experiments, we split the dataset into train and test sets using the stratified holdout method, with two-thirds of the data used for training and validation, and one-third for testing.

Multiclass classification performance

Table 5.1 presents the classification performances of the Transformer-based MVTS classifier compared to several baseline methods. To provide a comprehensive evaluation, we report accuracy, precision, recall, and F1 scores for each class. The experiments were conducted using five different train/test sets, which were sampled using stratified holdout, and we report the mean and standard deviation of the results.

Table 5.1: Multiclass classification performance of the proposed method with the baselines

Measures	FLT	LTV	TS-SUM	RNN	LSTM	ROCKET	Transformer
Accuracy	0.26 ± 0.012	0.32 ± 0.02	0.61 ± 0.091	0.43 ± 0.025	0.63 ± 0.03	0.74 ± 0.02	**0.83 ± 0.026**
Precision (X)	0.23 ± 0.024	0.34 ± 0.041	0.71 ± 0.054	0.53 ± 0.031	0.76 ± 0.028	0.92 ± 0.03	**0.95 ± 0.023**
Recall (X)	0.26 ± 0.053	0.39 ± 0.043	0.77 ± 0.024	0.63 ± 0.028	0.95 ± 0.023	0.98 ± 0.01	**0.98 ± 0.008**
F1 (X)	0.24 ± 0.032	0.36 ± 0.04	0.74 ± 0.034	0.58 ± 0.019	0.84 ± 0.014	0.95 ± 0.02	**0.97 ± 0.013**
Precision (M)	0.25 ± 0.012	0.32 ± 0.033	0.52 ± 0.031	0.41 ± 0.014	0.59 ± 0.018	0.66 ± 0.04	**0.82 ± 0.051**
Recall (M)	0.26 ± 0.023	0.33 ± 0.061	0.55 ± 0.022	0.40 ± 0.03	0.54 ± 0.014	0.7 ± 0.03	**0.85 ± 0.067**
F1 (M)	0.26 ± 0.026	0.33 ± 0.042	0.53 ± 0.023	0.41 ± 0.029	0.57 ± 0.02	0.68 ± 0.02	**0.83 ± 0.026**
Precision (BC)	0.23 ± 0.044	0.26 ± 0.024	0.45 ± 0.033	0.28 ± 0.031	0.50 ± 0.013	0.58 ± 0.02	**0.71 ± 0.055**
Recall (BC)	0.24 ± 0.053	0.21 ± 0.02	0.47 ± 0.014	0.26 ± 0.021	0.41 ± 0.023	0.57 ± 0.05	**0.70 ± 0.066**
F1 (BC)	0.24 ± 0.041	0.23 ± 0.024	0.46 ± 0.041	0.27 ± 0.031	0.45 ± 0.031	0.57 ± 0.03	**0.70 ± 0.053**
Precision (Q)	0.32 ± 0.034	0.34 ± 0.044	0.58 ± 0.045	0.48 ± 0.024	0.60 ± 0.024	0.81 ± 0.04	**0.85 ± 0.056**
Recall (Q)	0.25 ± 0.042	0.36 ± 0.071	0.66 ± 0.034	0.41 ± 0.042	0.68 ± 0.023	0.72 ± 0.03	**0.78 ± 0.048**
F1 (Q)	0.28 ± 0.014	0.35 ± 0.013	0.62 ± 0.043	0.45 ± 0.032	0.64 ± 0.024	0.77 ±0.03	**0.81 ± 0.033**

The results demonstrate that the Transformer-based MVTS classifier outperforms all other baselines across all performance measures. When considering the overall evaluation, ROCKET achieves the second-best performance, followed by the LSTM model in third place. Notably, the Transformer-based MVTS classifier achieves an accuracy that is 20% higher than that of the LSTM model. This highlights the significance of learning MVTS representations in both spatial and temporal domains, rather than solely relying on the temporal domain.

Among the shallow ML models, TS-SUM performs better than the FLT and LTV models. Overall, the exceptional performances of TS-SUM, RNN, LSTM, ROCKET, and our Transformer-based MVTS classifier emphasize the importance of time series representations in understanding solar events.

Binary classification performance

In the context of data-driven flare prediction, binary classification plays a significant role in distinguishing major flaring events from minor flaring events or flare quiet events. In this experiment, we focus on classifying X and M class MVTS instances as flaring events, while considering all other instances (Q and BC) as non-flaring events. The figure depicts the mean binary classification performances of all models over five different train/test sam-

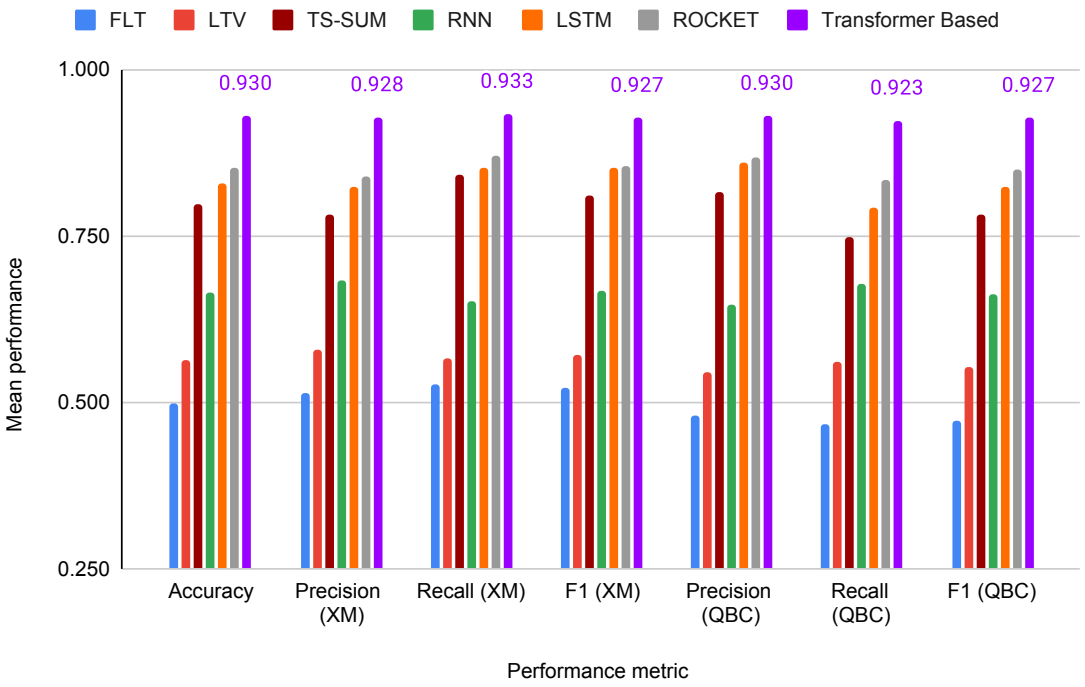

Fig. 5.10: Binary classification performance of all baselines.

ples. Evaluation metrics such as accuracy, precision, recall, and F1 scores are used for both the flaring and non-flaring classes.

The results demonstrate that the Transformer-based MVTS model outperforms all other baseline models, and achieves an average improvement of approximately 8% compared to the second-best performing ROCKET algorithm across all performance metrics. These findings highlight the superior performance of our model in binary classification and multi-class classification. This consistency reinforces the efficacy and reliability of our Transformer-based model in accurately predicting flaring events.

Classification varying training set size

To investigate the adaptability of our model to larger training datasets, we conducted experiments by varying the size of the training set. The training set size was adjusted from

10% to 90% of the total dataset size, while the remaining instances were used for testing. Stratified train/test sampling was performed, and the classification performance of the classifiers was evaluated five times using distinct samples of training and test sets.

In Figure 5.11 5.12, we present the mean accuracy values and mean F1 (X class) values obtained from five runs. Across all training set sizes, our Transformer-based MVTS classifier consistently outperformed the other baselines. Notably, the Transformer-based MVTS model achieved a classification accuracy of 75% using only 20% of the training data, surpassing the performance of the third-best performing LSTM model, which required 90% of the training data to achieve a similarly high level of performance. We observed consistent improvement patterns in deep learning and kernel-based methods, including our Transformer-based model, ROCKET, LSTM, and RNN.

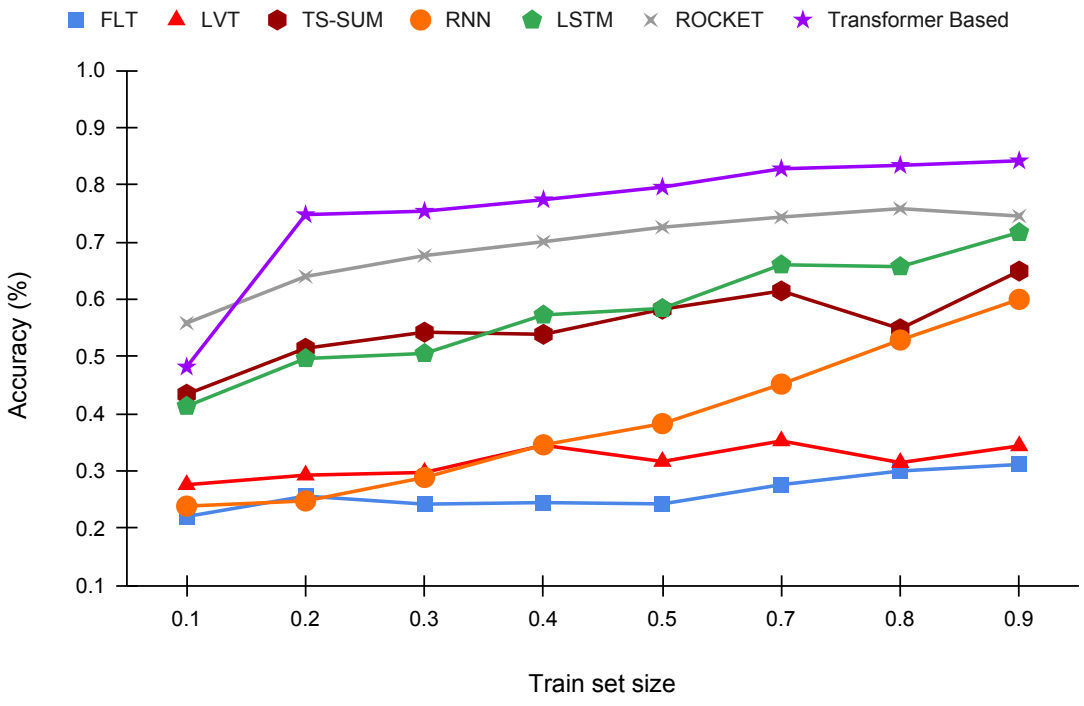

Fig. 5.11: Multi-class classification accuracy with increasing training data.

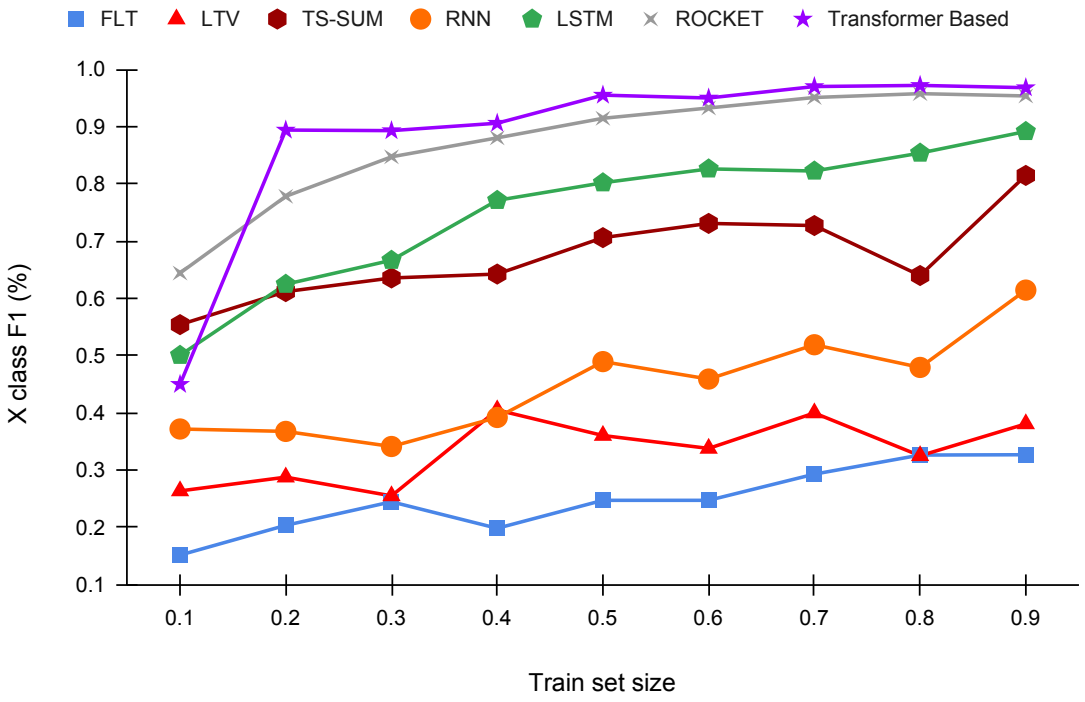

Fig. 5.12: F1 for X class with increasing training data

This observation suggests that with sufficiently large datasets, deep learning models have the potential to outperform traditional classifiers or embedding methods by a significant margin. These findings underscore the superiority of Transformer models when working with large datasets.

t-SNE Embedding performance

Visualizing high-dimensional data in 2D or 3D space using techniques like t-SNE is a well-established method for assessing the effectiveness of learned representations. To evaluate the quality of the learned MVTS representations, we present a visualization of the t-SNE transformed MVTS representations extracted from the final layer of the Transformer-based model. All instances are projected onto a t-SNE-reduced 2D space (see Figure 5.13). We employed a stratified holdout strategy for pre-training the model.

The resulting 2D projection demonstrates distinct clustering of the MVTS instances. The t-SNE scatter plot provides meaningful insights, as it allows us to easily distinguish patterns among the four classes. Flare-quiet events (Q) and minor flaring events (B and C) exhibit relatively similar characteristics. On the other hand, X and M class flares show significant dissimilarity from the other classes. Additionally, we observe that certain flare-quiet events share similarities with minor flaring events, while some minor flares display characteristics similar to M-class flares. The characteristics of X-class flares are distinct, with no observed similarity in instances from other classes.

By visualizing the t-SNE transformed representations, we gain valuable insights into the distinguishable patterns and similarities among the different classes of MVTS instances. This analysis allows for a deeper understanding of the learned representations and sheds light on the distinct features and characteristics of flaring events.

Ablation Study of the Transformer-base MVTS Classification Mode

To gain a better understanding of the contributions and effectiveness of the different layers in our model, we conducted several experiments to evaluate the significance of various aspects (refer to figure 5.14). Firstly, we assessed the importance of the self-attention mechanism by removing it from the model architecture and comparing the results. The removal of the attention mechanism led to a noticeable drop in accuracy, from 83% to 71%. This outcome highlights the significant role played by the Multi-Head Attention layer in capturing relevant patterns and relationships within the MVTS data.

Secondly, we examined the impact of layer normalization by removing the layer normalization layers from the model. This resulted in a decrease in accuracy from 83% to 77%. This finding underscores the importance of layer normalization in maintaining the model's performance and stability.

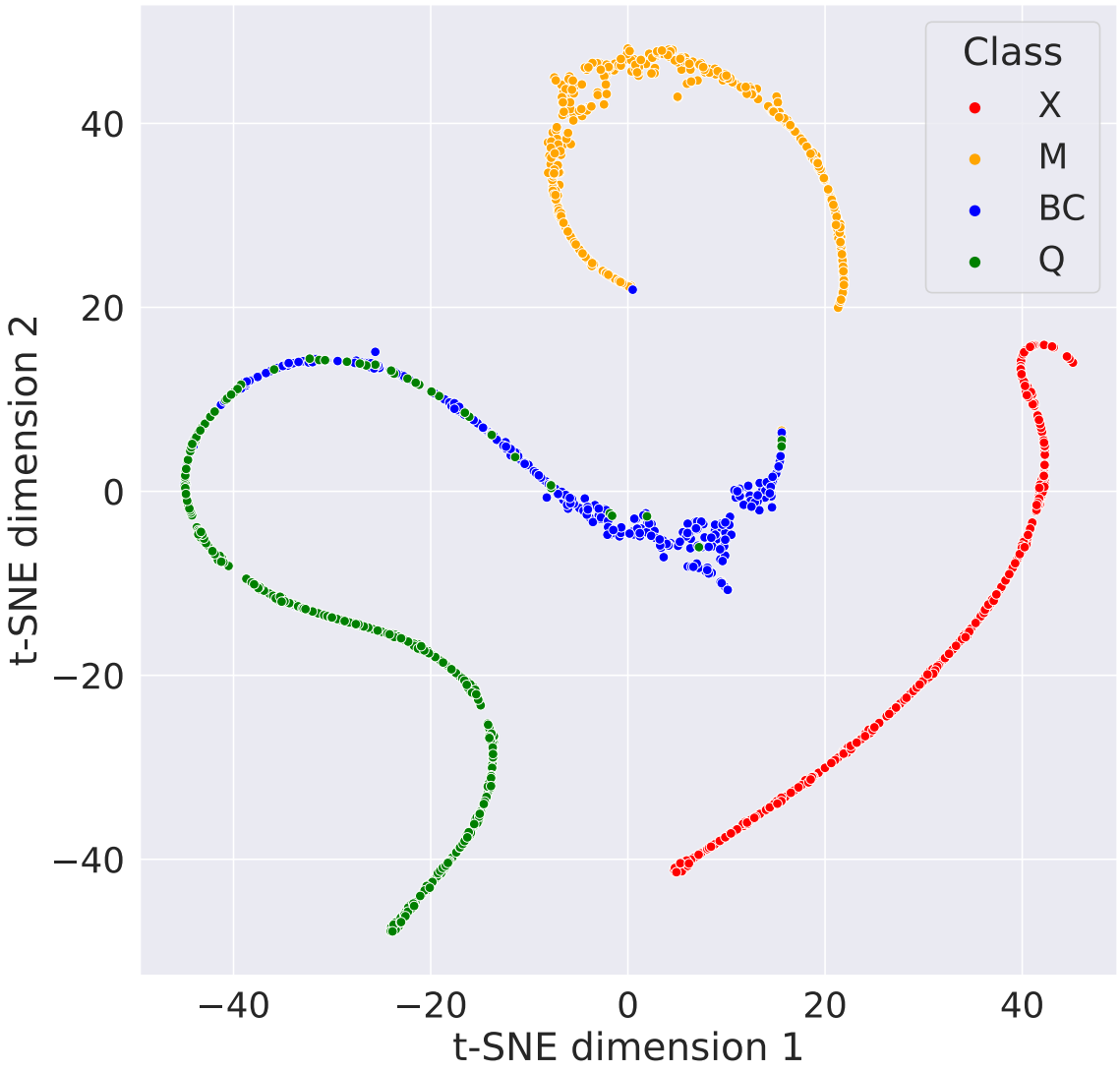

Fig. 5.13: t-SNE embedding of Transformer-based generated representations of all MVTS instances in the dataset

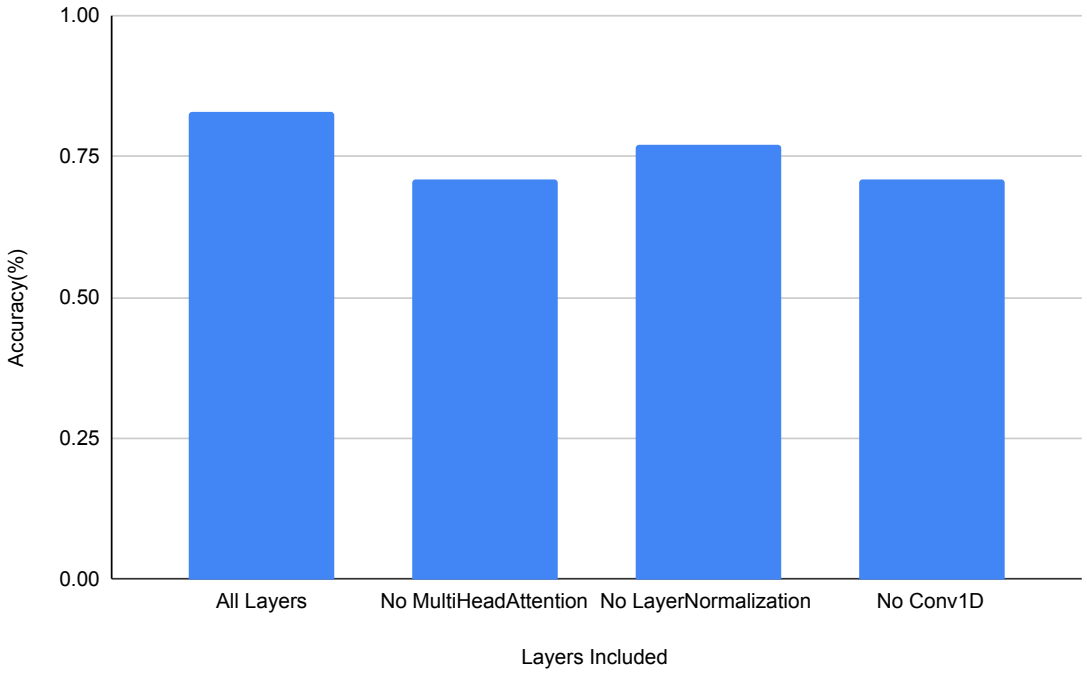

Fig. 5.14: Ablation Study: Revealing the Contributions of Model Components in MVTS Classification of Solar Flares.

Lastly, we investigated the effect of the 1D convolutional layers. When these layers were removed from the model, there was a significant drop in accuracy from 83% to 71%. This result demonstrates the crucial role played by the 1D convolutional layers in capturing important temporal features and contributing to the overall performance of the model.

Overall, the ablation study provided valuable insights into the contributions of different layers in our model. The significant decrease in accuracy upon removing the attention mechanism, layer normalization, and 1D convolutional layers highlights their importance in capturing relevant patterns, maintaining stability, and extracting essential temporal features. These findings underscore the effectiveness and significance of each layer in our model architecture.

CHAPTER 6

CONCLUSION

The field of astrophysics lacks a specific physical theory that comprehensively explains the mechanism behind the occurrence of solar flares, which limits the ability to forecast and classify them [40]. While various groups of physicists are actively researching to unveil a definitive theory for flare prediction, the likelihood of success remains uncertain. Given the rapid advancements in AI and machine learning, the most promising approach is to adopt a data-driven strategy using the Active Region parameters observed by the solar dynamics observatory. The goal is to develop a model that can establish an empirical relationship between AR parameters and flare occurrences.

In this work, we presented the application of novel MINIROCKET classifier, Transformers/Attention based classifier, and baselines using the SWAN-SF dataset along with the challenges posed by extreme class imbalance. We discussed various approaches to address these issues, including undersampling and oversampling techniques. To evaluate our approach, we adopted a total of 20 distinct partition pairs. One partition was used for training, while the remaining four were used for individual testing. For instance, we trained the model with partition 1 and tested it with partitions 2, 3, 4, and 5 separately.

Furthermore, we introduced the efficient application of MINIROCKET classifier, emphasizing minimal data manipulation and augmentation. Our model was compared with other classes of classifiers, such as LSTM, MRSEQL, SVM, and CIF. To assess performance, we employed the True Skill Statistic (TSS) score and Heidke Skill Score (HSS2), the most commonly used metrics for flare prediction in class imbalance data. Our research revealed that MINIROCKET outperformed other classifiers on the SWAN-SF dataset. Notably, across all experimental settings, MINIROCKET achieved an average improvement of 19.8%

in terms of the TSS score and 20.92% in terms of the HSS2 score.

Additionally, we introduced an end-to-end transformer-based flare prediction model that utilized the self-attention mechanism for multivariate time series (MVTS) classification. We validated our experiments using the reduced dataset and showcased the strengths of the Transformer model in capturing temporal relationships within MVTS instances, including higher-order inter-variable relationships and local and global temporal changes. Our model, integrating attention/transformer-based techniques, achieved an impressive accuracy of 83% in multi-class MVTS classification on the solar flare prediction dataset.

These results underscore the potential of our approach in enhancing the accuracy and reliability of solar physics and space weather forecasting. The effectiveness of the Transformer model in handling MVTS data complexities and leveraging self-attention mechanisms for crucial pattern detection is evident. Additionally, the remarkable performance demonstrated by MINIROCKET with minimal data preprocessing can significantly advance the vision of real-time solar flare classification. These contributions hold the promise for improving space weather forecasting.

REFERENCES

[1] A. Ahmadzadeh, B. Aydin, M. K. Georgoulis, D. J. Kempton, S. S. Mahajan, and R. A. Angryk, "How to train your flare prediction model: Revisiting robust sampling of rare events," *The Astrophysical Journal Supplement Series*, vol. 254, no. 2, p. 23, 2021.

[2] E. Larsen, "Predicting solar flares with remote sensing and machine learning," *CoRR*, vol. abs/2110.07658, 2021. [Online]. Available: https://arxiv.org/abs/2110.07658

[3] S. M. Abd Elrahman and A. Abraham, "A review of class imbalance problem," *Journal of Network and Innovative Computing*, vol. 1, no. 2013, pp. 332–340, 2013.

[4] 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, pp. 3549–3578, 2014.

[5] A. Dempster, D. F. Schmidt, and G. I. Webb, "Minirocket: A very fast (almost) deterministic transform for time series classification," in *Proceedings of the 27th ACM SIGKDD conference on knowledge discovery & data mining*, 2021, pp. 248–257.

[6] A. Dempster, F. Petitjean, and G. I. Webb, "Rocket: exceptionally fast and accurate time series classification using random convolutional kernels," *Data Mining and Knowledge Discovery*, vol. 34, no. 5, pp. 1454–1495, 2020.

[7] M. Middlehurst, J. Large, and A. Bagnall, "The canonical interval forest (cif) classifier for time series classification," in *2020 IEEE international conference on big data (big data)*. IEEE, 2020, pp. 188–195.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[9] O. Olabiyi, "Multi-turn dialogue response generation with autoregressive transformer models," *ArXiv*, vol. abs/1908.01841, 2019.

[10] P. S. McIntosh, "The classification of sunspot groups," *Solar Physics*, vol. 125, pp. 251–267, 1990.

[11] S. F. Boubrahimi, B. Aydin, D. Kempton, and R. Angryk, "Spatio-temporal interpolation methods for solar events metadata," in *2016 IEEE International Conference on Big Data (Big Data)*. IEEE, 2016, pp. 3149–3157.

[12] Y. Cui, R. Li, L. Zhang, Y. He, and H. Wang, "Correlation between solar flare productivity and photospheric magnetic field properties: 1. maximum horizontal gradient, length of neutral line, number of singular points," *Solar Physics*, vol. 237, pp. 45–59, 2006.

[13] J. Jing, H. Song, V. Abramenko, C. Tan, and H. Wang, "The statistical relationship between the photospheric magnetic parameters and the flare productivity of active regions," *The Astrophysical Journal*, vol. 644, no. 2, p. 1273, 2006.

[14] K. Leka and G. Barnes, "Photospheric magnetic field properties of flaring versus flare-quiet active regions. ii. discriminant analysis," *The Astrophysical Journal*, vol. 595, no. 2, p. 1296, 2003.

[15] D. Yu, X. Huang, H. Wang, and Y. Cui, "Short-term solar flare prediction using a sequential supervised learning method," *Solar Physics*, vol. 255, pp. 91–105, 2009.

[16] H. Song, C. Tan, J. Jing, H. Wang, V. Yurchyshyn, and V. Abramenko, "Statistical assessment of photospheric magnetic features in imminent solar flare predictions," *Solar Physics*, vol. 254, pp. 101–125, 2009.

[17] 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, pp. 157–175, 2013.

[18] A. Al-Ghraibah, L. Boucheron, and R. McAteer, "An automated classification approach to ranking photospheric proxies of magnetic energy build-up," *Astronomy & Astrophysics*, vol. 579, p. A64, 2015.

[19] N. Nishizuka, K. Sugiura, Y. Kubo, M. Den, S. Watari, and M. Ishii, "Solar flare prediction model with three machine-learning algorithms using ultraviolet brightening and vector magnetograms," *The Astrophysical Journal*, vol. 835, no. 2, p. 156, 2017.

[20] 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.

[21] S. M. Hamdi, D. Kempton, R. Ma, S. F. Boubrahimi, and R. A. Angryk, "A time series classification-based approach for solar flare prediction," in *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017, pp. 2543–2551.

[22] R. A. Angryk, P. C. Martens, B. Aydin, D. Kempton, S. S. Mahajan, S. Basodi, A. Ahmadzadeh, X. Cai, S. Filali Boubrahimi, S. M. Hamdi *et al.*, "Multivariate time series dataset for space weather data analytics," *Scientific data*, vol. 7, no. 1, p. 227, 2020.

[23] W. D. Pesnell, B. J. Thompson, and P. C. Chamberlin, *The Solar Dynamics Observatory (SDO)*. New York, NY: Springer US, 2012, pp. 3–15. [Online]. Available: https://doi.org/10.1007/978-1-4614-3673-7_2

[24] J. T. Hoeksema, Y. Liu, K. Hayashi, X. Sun, J. Schou, S. Couvidat, A. Norton, M. Bobra, R. Centeno, K. Leka *et al.*, "The helioseismic and magnetic imager (hmi) vector magnetic field pipeline: overview and performance," *Solar Physics*, vol. 289, pp. 3483–3530, 2014.

[25] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, "On the class imbalance problem," in *2008 Fourth international conference on natural computation*, vol. 4. IEEE, 2008, pp. 192–201.

[26] G. H. Fisher, D. J. Bercik, B. T. Welsch, and H. S. Hudson, "Global forces in eruptive solar flares: the lorentz force acting on the solar atmosphere and the solar interior," *Solar Physics*, vol. 277, no. 1, pp. 59–76, 2012.

[27] K. Leka and A. Skumanich, "On the value of'αar'from vector magnetograph data," *Solar Physics*, vol. 188, no. 1, pp. 3–19, 1999.

[28] J. Wang, Z. Shi, H. Wang, and Y. Lue, "Flares and the magnetic nonpotentiality," *The Astrophysical Journal*, vol. 456, p. 861, 1996.

[29] C. J. Schrijver, "A characteristic magnetic field pattern associated with all major solar flares and its use in flare forecasting," *The Astrophysical Journal*, vol. 655, no. 2, p. L117, 2007.

[30] J. P. 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.

[31] 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.

[32] F. Woodcock, "The evaluation of yes/no forecasts for scientific and administrative purposes," *Monthly Weather Review*, vol. 104, no. 10, pp. 1209–1214, 1976.

[33] C. H. Lubba, S. S. Sethi, P. Knaute, S. R. Schultz, B. D. Fulcher, and N. S. Jones, "catch22: Canonical time-series characteristics," *CoRR*, vol. abs/1901.10200, 2019. [Online]. Available: http://arxiv.org/abs/1901.10200

[34] M. Middlehurst, J. Large, and A. J. Bagnall, "The canonical interval forest (CIF) classifier for time series classification," *CoRR*, vol. abs/2008.09172, 2020. [Online]. Available: https://arxiv.org/abs/2008.09172

[35] T. L. Nguyen, S. Gsponer, I. Ilie, M. O'Reilly, and G. Ifrim, "Interpretable time series classification using linear models and multi-resolution multi-domain symbolic representations," *CoRR*, vol. abs/2006.01667, 2020. [Online]. Available: https://arxiv.org/abs/2006.01667

[36] T. L. Nguyen, S. Gsponer, and G. Ifrim, "Time series classification by sequence learning in all-subsequence space," in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, 2017, pp. 947–958.

[37] A. A. M. Muzaheed, S. M. Hamdi, and S. F. Boubrahimi, "Sequence model-based end-to-end solar flare classification from multivariate time series data," in *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2021, pp. 435–440.

[38] A. Pasos Ruiz, M. Flynn, J. Large, M. Middlehurst, and A. Bagnall, "The great multivariate time series classification bake off: a review and experimental evaluation of recent algorithmic advances," *Data Mining and Knowledge Discovery*, vol. 35, no. 2, pp. 401–449, 2021.

[39] A. J. Bagnall, H. A. Dau, J. Lines, M. Flynn, J. Large, A. Bostrom, P. Southam, and E. J. Keogh, "The uea multivariate time series classification archive, 2018," *arXiv preprint arXiv:1811.00075*, 2018.

[40] K. Kusano, T. Iju, Y. Bamba, and S. Inoue, "A physics-based method that can predict imminent large solar flares," *Science*, vol. 369, no. 6503, pp. 587–591, 2020.