Mesopotamian journal of Big Data
Vol. **(2025)**, 2025, **pp**. 415-428
DOI: https://doi.org/10.58496/MJBD/2025/025 ISSN: 2958-6453
https://mesopotamian.press/journals/index.php/BigData

Research Article

# Intelligent Forecasting of Solar Atmospheric Disturbances via Capsule Neural Networks and Space Weather Data

Eman Turki Mahdi[1,*], Mohammed E. Seno[2], Abdullah I. Abdulghafar[2]

[1] *Department of Computer Networks Systems, College of Computer Science and Information Technology, University of Anbar, Ramadi, Iraq*
*2 Department of Computer Sciences, College of Sciences, University of Al Maarif, Al Anbar,31001, Iraq*

**Article History**
Received 02 Aug 2025
Revised 29 Aug 2025
Accepted 22 Sep 2025
Published 30 Oct 2025

**Keywords**
Solar Flare Prediction,
Binary Classification,
Multi-Class Classification
Capsule Networks
Deep Learning
Class Imbalance
Space Weather Forecasting

**ABSTRACT**

Solar flares represent a major challenge to satellite communications, navigation systems, and terrestrial power grids, making accurate forecasting essential for mitigating their disruptive effects. This study aims to improve the reliability of solar flare prediction by developing a deep learning framework based on Capsule Networks (CapsNet). The proposed approach integrates feature engineering, data preprocessing, and imbalance-handling techniques such as SMOTE and Focal Loss. Using NASA space weather data, we constructed both binary (storm/no storm) and multi-class (C, M, X) classification models across forecasting windows of 6, 24, and 48 hours. The 48-hour binary model achieved 96% accuracy with a True Skill Statistic (TSS) of 0.92, significantly outperforming existing CNN-based approaches. Meanwhile, the 6-hour multi-class model delivered high recall for rare but critical X-class flares (0.86) and strong overall accuracy (92%). These results demonstrate that CapsNet can effectively capture complex spatio-temporal dependencies in space weather data, offering a robust and scalable solution for early-warning systems in solar flare forecasting.

## 1. INTRODUCTION

The Sun's activity has an effect on the space environment around the Earth, leading to a family of atmospheric disturbances collectively known as space weather[1]. Among these solar flares and geomagnetic storms represent the most disruptive phenomena, capable of degrading satellite communication, damaging navigation systems, and even destabilizing power grids on Earth[2], [3]. With increasing global reliance on satellite infrastructure and wireless technologies. The ability to accurately predict solar is induced atmospheric storms has become an essential field of study [4], [5]. Solar flare forecasting remains a challenging task, as conventional forecasting techniques are hindered by the noisy, large-dimensional, and highly unbalanced nature of space weather data [6]. Reliable forecasting in space, weather strongly depends on the availability of comprehensive and well-structured datasets[7]. Similar to the development of large-scale databases in many fields, such as offline handwritten digit recognition[8], the construction and refinement of space weather datasets play a crucial role in enabling deep learning models to achieve high performance and robustness[9]. The nonlinear interactions and inherent temporal dependencies of solar activity are frequently missed by traditional statistical models and superficial machine learning approaches like decision trees, support vector machines[11], [12]. In contrast, deep learning has proven to be a powerful alternative, as it can capture complex spatial-temporal patterns directly from raw data [13]. Beyond pattern recognition, it is also able to learn hierarchical feature representations that traditional models often overlook [14], [15]. In recent research on flare prediction, Convolutional neural networks (CNNs), recurrent neural networks (RNNs), have been applied to capture spatial patterns [15], recurrent neural networks (RNNs) have been employed to address temporal dependencies [16], and hybrid architectures combining these models have also been investigated, demonstrating notable improvements in predictive accuracy [17] [18]. Nevertheless, a lot of these models still have issues with class imbalance, dynamic routing, and spatial pose invariance, especially in multi-class scenarios with C-, M-, and X-class flare events[19]. Despite the success obtained by classic machine learning, deep learning (to which also belong CNNs, RNNs, and hybrid models) in forecasting, the current forecasting technology still has strong limitations. Most work faces the challenge of data with class imbalance, noise and high dimensionality, and few are capable of modeling spatial-temporal dependencies. In

*Corresponding author. Email: maymoonat@uoanbar.edu.iq*

addition, many previous works may achieve moderate accuracy, but cannot offer reliable detection of rare, but crucial X-class flares. These difficulties indicate an obvious research deficiency for designing stronger and intuitive architectures for predicting solar flares. To bridge this gap, the current study suggests an improved Capsule Network (CapsNet) architecture utilizing feature engineering, data balancing techniques (SMOTE) and Focal Loss. This is a promising step toward the multi-time improvement of horizon forecast accuracy. It shows greater sensitivity to minority flare classes, so it contributes towards a more effective detection and early-warning system of space weather events. A robust forecasting architecture based on Capsule Networks (CapsNet), which has proven to be highly effective in learning discriminative features and maintaining spatial relationships even with sparse data, is proposed in this work as a response. In order to predict flares over three time windows: six, twenty-four, and forty-eight hours. we create several binary and multi-class classification models in this study. During model training, we use a combination of SMOTE and Focal Loss to overcome the inherent data imbalance issue, particularly with respect to rare but dangerous X-class flares. Our results show significant gains in accuracy and true skill score (TSS) compared to current benchmarks, providing a workable and scalable solution for early-warning systems in operational space weather monitoring. Our paper makes the following contributions:

1. Proposing an enhanced Capsule Network (CapsNet) architecture. Tailored it for multichannel time-series space weather data, with deeper convolutional blocks, batch normalization, and increased routing iterations to better capture spatio-temporal dependencies.
2. Addressing severe class imbalance using SMOTE and Focal Loss, which significantly improves the detection of rare but high-impact X-class solar flares.
3. Achieving strong forecasting performance across multiple horizons: the 48-hour binary model reached 96% accuracy with a TSS of 0.92, while the 6-hour multi-class model obtained high recall for M-class (0.94) and X-class (0.86) events.
4. Providing a transparent end-to-end pipeline that integrates preprocessing, imbalance handling, and CapsNet enhancements, highlighting practical implications for early-warning systems.

The rest of this paper is organized as follow: Section 2 reviews relate studies and identifies existing gaps; Section 3 presents the dataset, preprocessing, and the proposed CapsNet architecture; Section 4 reports experimental results and discussion; Section 5 outlines limitations and future work; and Section 6 concludes the paper.

## 2. LITERATURE REVIEW

Forecasting solar flare activity has a major research area in space weather modelling, with numerous studies proposing data-driven approaches leveraging machine learning and deep learning techniques. In this section, we review key contributions that are most relevant to this study, focusing on models that address the classification of flare intensity (C, M, X classes) and storm forecasting over varying temporal windows.

**2.1 Classical Machine Learning Approaches:** Some flare prediction efforts relied heavily on statistical techniques or may be traditional machine learning classifiers. For example, using magnetogram-derived features, Boucheron et al. (2015) [20]used support vector machines and decision trees. These techniques, however, were not very effective at managing noisy, high-dimensional, and unbalanced datasets, particularly in multi-class environments where the C-class dominates the sample distribution.

**2.2 Deep Flare Net (DeFN):** It is one of the more notable early deep learning contributions is Deep Flare Net (DeFN), proposed by Nishizuka et al. (2018)[21]. They utilised GOES and SDO/HMI features to predict solar flares. DeFN achieved a TSS of 0.61 for 24-hour binary flare classification (≥M-class), and TSS ≈ 0.70 when fine-tuned for flare magnitude thresholds. Despite these promising results, the model struggled to generalize in highly imbalanced class distributions.

**2.3 Hybrid CNN-Based Architectures:** Zheng et al. (2019)[22] presented a hybrid CNN model that achieved 88% accuracy and TSS of 0.68 on binary flare prediction tasks by extracting spatial features from solar magnetograms. Similarly, Li et al. (2020)[23] reported TSS ≈ 0.75 for predicting M- and X-class flares within 24-hour windows using an LSTM-CNN with attention mechanisms. Although these models showed notable advancements over traditional techniques, they were still dependent on sizable datasets and had interpretability issues with spatial hierarchies and routing.

**2.4 Transformer-Based and Ensemble Models:** This approach has been explored in recent studies. For example, Pan et al. (2023)[24] implemented ViT-based classifiers with an Efficient Channel Attention (ECA) mechanism, achieving recall rates between 0.79 and 0.85 for M-class flares in 24-hour settings. Although transformer models show potentially, they

often demand significant computational resources and large-scale datasets for effective training. Table 1 summarizes selected prior studies on solar flare forecasting, outlining their techniques, advantages, limitations, and key performance metrics.

TABLE I: COMPARISON OF PRIOR STUDIES ON SOLAR FLARE FORECASTING

| Study | Technique | Advantages | Limitations | Key Performance Metrics |
|---|---|---|---|---|
| **Boucheron et al. (2015)** [20] | SVM / Decision Trees | Simple and interpretable; effective for small-scale flare prediction | Limited ability to capture nonlinear and temporal dependencies; lower accuracy on large datasets | Accuracy ≈ 75–80% |
| **Nishizuka et al. (2018)** [21] | CNN-based model (Deep Flare Net, DeFN) | Captures spatial features from solar magnetograms; demonstrated operational use | Limited temporal modeling; reduced performance for rare X-class flares | Accuracy ≈ 85%, TSS ≈ 0.63 |
| **Zheng et al. (2019)** [22] | LSTM-CNN + Attention | Models sequential and spatial dependencies simultaneously; improves recall | Requires larger training datasets; higher computational cost | Accuracy ≈ 87%, Recall ≈ 0.78 |
| **Li et al. (2020)** [23] | Hybrid CNN + SVM | Combines feature extraction and classification; more robust than single models | Computationally intensive; limited scalability for real-time use | Accuracy ≈ 88%, TSS ≈ 0.75 |
| **Pan et al. (2023)** | Transformer with Efficient Channel Attention (ECA) | Captures long-range temporal dependencies; powerful representation | Needs large datasets; costly training | AUC ≈ 0.89, Recall ≈ 0.80 |

The reviewed works collectively underscore the evolution of flare forecasting techniques. It includes classical classifiers to deep and hybrid models. Although recent deep learning approaches have improved flare prediction compared to traditional methods, several limitations remain. Many models still struggle with imbalanced datasets, which makes the detection of rare but critical X-class events unreliable. Others fail to capture the full spatio-temporal complexity of solar activity, resulting in only moderate improvements in accuracy. In addition, some advanced architectures require high computational resources, which reduces their practicality for real-time forecasting. These challenges highlight the need for more robust and interpretable models that can better address data imbalance and preserve temporal relationships. The results of these studies were also compared with the results of our research, as shown in Table 9. While CNN- and RNN-based hybrids capture either spatial or temporal dependencies, they often require very large datasets and still struggle with severe imbalance, making X-class detection unreliable. Transformer-based models improve long-range temporal modelling but demand substantial computational resources, limiting their real-time applicability. In contrast, the proposed CapsNet framework explicitly preserves spatio-temporal structures through capsule routing while mitigating imbalance via SMOTE and Focal Loss. This allows for higher recall of rare but critical events and more reliable long-horizon forecasts, directly addressing the gaps identified in prior studies.

## 3. METHODOLOGY

In this section, we present the methodological framework designed for atmospheric storm prediction using multivariate space weather data.

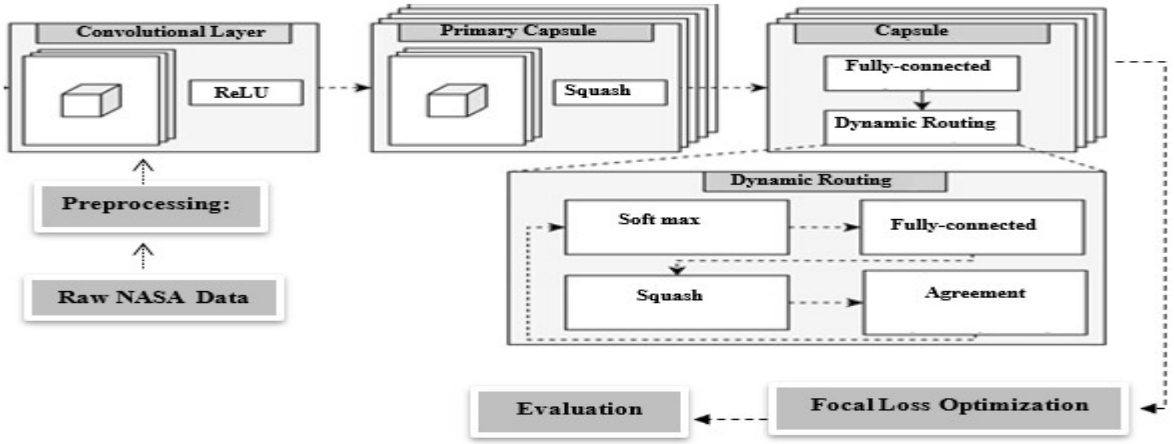



Fig 1: Methodology Architecture

The proposed pipeline integrates preprocessing of the signal. outlier-aware feature encoding, and a deep learning architecture tailored for imbalanced classification tasks. This framework enables the extraction of meaningful patterns from noisy, high-dimensional time-series data while preserving their temporal and spatial characteristics. The complete workflow is shown in Fig 1, which illustrates each stage from data acquisition to final prediction, along with the architectural components of the enhanced Capsule Network.

For transparency and reproducibility, we also report the hyperparameter settings and experimental configurations used during training. Table 2 summarizes the main parameters, including optimizer, learning rate, batch size, number of epochs, loss function, routing iterations, and hardware specifications. These settings were selected empirically to balance convergence speed with generalization under imbalanced data conditions.

TABLE II. HYPERPARAMETER SETTINGS FOR CAPSULE NETWORK MODELS

| **Parameter** | **Value / Setting** |
|---|---|
| Optimizer | Adam |
| Initial Learning Rate | 0.001 |
| Batch Size | 64 |
| Epochs | 50 (binary), up to 100 (48h model) |
| Loss Function | Focal Loss ($\gamma$ = 2.0) + Class Weights |
| Routing Iterations | 3–5 (dynamic routing) |
| Dropout Rate | 0.2–0.3 |
| Weight Decay (L2 reg.) | 1e-4 |
| Normalization | LayerNorm after dense layers |
| Oversampling | SMOTE (applied on training split only) |
| Training Time | ~2–4 minutes per epoch (15–30 minutes total per model) |

### 3.1 Dataset Description

Our study dataset was obtained from NASA Space Weather Database of Notifications, Knowledge, Information (DONKI) [25]. It is a large archive that supplies multiple multivariate time series of solar and geomagnetic activity indices and parameters, which would be possible inputs to tests specifically tailored to assess forecast-ability. Along with the other physical and solar/environmental parameters, each dataset observation also has a solar flare class (during A-X range), as well speed/density of solar wind, fluxes for protons/electrons, X-ray intensity and indices for geomagnetics like Kp index/Dst index. Such parameters are key to modeling the response of space weather and implications for atmospheric perturbations on Earth. The dataset is built with a set of flare records classified under the following five classes: A, B, C, M and X. Classes A and B are weaker solar activity events which have little impact on Earth, while C class flares are stronger than the previous ones and can cause geomagnetic storms occasionally during high solar activity periods. In the present study only C, M and X class flares were included due to their importance for storm prediction. We discarded A and B class flares in order to minimize the level of noise, remove apparent spurious patterns and keep the focus on high impact solar phenomena. The dataset was converted to labeled samples with 24-hour time windows after filtering, cleaning and preprocessing. Both spatio-temporal readings from the sensors were aggregated in terms of windows and are tagged for whether there was any storm-related activity in that window or not. The resulting dataset includes around of 25,000 archive samples labeled as “storm” (1) or “no storm” (0), based on combined flare classification and geomagnetic criteria (e.g., increased Kp or Dst indices). The data given were several predictors of a storm. The benefit is that it comes with a rich feature set and timestepping. 182 The models are able to account for complex spatio-temporal relationships. However, some challenges were noted. Example of those are that there is some missing values and overlapping events that do not help on the labeling (what is a flare and what is not a flare, e.g. CMEs (Coronal Mass Ejections) can sometimes appear without a accompanying flare). However, the dataset is well positioned for predictive modeling purposes, especially when combined with feature engineering and noise-robust learners. The final fully cleaned and preprocessed samples have been divided into three subsets as mentioned below to achieve its model development and cross validation.:

- 70% for training: used to fit the model parameters.
- 15% for validation: used during training to tune hyperparameters and monitor overfitting.
- 15% for testing: held out entirely to assess model generalization performance.

This structured split ensures that the model is trained on a diverse set of events, while maintaining unseen samples for fair and unbiased evaluation. The following table shows Dataset Features and Descriptions.

TABLE III: DATASET FEATURES AND DESCRIPTIONS

| Feature Name | Unit / Type | Description |
|---|---|---|
| Timestamp | DateTime | The exact date and time of the recorded observation (in UTC). |
| Solar Flare Class | C / M / X (Categorical) | Class of detected solar flare based on X-ray intensity. |
| X-ray Flux | W/m² | Solar X-ray radiation intensity (e.g., 1–8 Å and 0.5–4 Å channels). |
| Solar Wind Speed | km/s | Velocity of solar wind particles measured by satellites. |
| Solar Wind Density | particles/cm³ | Concentration of solar wind protons. |
| Proton Flux | pfu | High-energy proton flux measured in particle flux units. |
| Electron Flux | counts/s | High-energy electron count rate. |
| Kp Index | Integer (0–9) | Global geomagnetic activity index (3-hour resolution). |
| Dst Index | nT (nanoTesla) | Measures intensity of geomagnetic storms at equatorial latitudes. |
| Storm Label | Binary (0 or 1) | Label indicating the presence of a storm: 1 = storm, 0 = no storm. |
| Time Window | 24 hours | Fixed-duration window for aggregating features and assigning labels. |

### 3.2 Data Preprocessing

Before training the prediction model, the dataset experienced a number of preprocessing procedures to improve data quality, reduce noise, and augment the input's representational structure then, the forecasting model was trained. This approach necessary to guarantee that the model could efficiently learn from the underlying patterns because of the dataset's multivariate and temporal nature. The primary preprocessing steps were:

**(1)Missing Value Handling**: Space weather datasets are impacted by missing sensor readings from satellite outages or maintenance lapses. To address this, we used forward-filling and linear interpolation techniques to impute missing data without sacrificing time-series continuity, as in the following equation[26]:

$$x_t = x_{t-1} + \frac{x_{t+1} - x_{t-1}}{2} \quad \ldots\ldots\ldots (1)$$

Where $x_t$ is the imputed (estimated) value at time step t, $x_{t-1}$ mean value at the previous time step, $x_{t+1}$ is the known value at the next time step.

**(2)Noise Filtering and Outlier Detection**: Values of abrupt spikes or non-physical in certain features, such as solar wind density or proton flux, were detected using threshold-based rules and smoothed using rolling averages equation[27]. This step was crucial to prevent misleading inputs from affecting the learning process.

$$x_t^{\wedge} = \frac{1}{\omega}\sum_{i=0}^{\omega-1} x_{t-i} \quad \ldots\ldots..(2)$$

Where $x^{\wedge}_t$ is the smoothed value at time t, $x_{t-i}$ is the observed value $i$ steps before time $t$, $\omega$ is the size of the sliding windows .

**(3) Normalization**: Numerical features of all continuous variables were scaled using Min-Max normalization function [28] to map their values to the [0, 1] range. This ensured consistent feature magnitudes and improved convergence during model training.

$$x'_t = \frac{x_t - x_{min}}{x_{max} - x_{min}} \quad \ldots\ldots\ldots..(3)$$

Where $x'_t$ the normalized value at time $t$, $x_t$ is the original raw value at time $t$, $x_{min}$ is The minimum value of the feature over the dataset, $x_{max}$ The maximum value of the feature over the dataset.

**(4) Temporal Windowing:** The data was segmented into fixed 24-hour windows, each treated as a single sample. These windows were labeled as either "storm" or "no storm" based on the flare activity and geomagnetic indices within that

period. This allowed the model to learn from patterns spread over daily intervals, capturing both short-term spikes and sustained trends.

**(5) Feature Engineering:** Additional temporal features such as moving averages, standard deviations, and lag features functions [28]were computed for key indicators like proton flux and X-ray intensity. These features help the model capture temporal dynamics and early-warning signatures of storms.

$$\mu_t = \frac{1}{k}\sum_{i=0}^{k-1} x_{t-i} \qquad \text{Rolling mean} \quad (4)$$

$$\sigma_t = \sqrt{\frac{1}{k}\sum_{i=0}^{k-1}(x_{t-i} - \mu_t)^2} \qquad rolling\ standard\ Deviation \quad (5)$$

$$x_t^{(\ell)} = x_{t-\ell} \qquad Lag\ Feature \ldots\ldots\ldots\ldots\ldots\ldots\ldots (6)$$

Where $\mu_t$ The rolling average over the last k values at time t, $\sigma_t$: The rolling standard deviation over the last k values at time t, $x_{t-i}$ The observed value i steps before t, $x_t^{(\ell)}$ The lagged value time steps before t, k: The size of the rolling window, and $\ell$: The lag interval .

**(6)Matrix Reshaping:** Each 24-hour window was reshaped into a 2D matrix where rows represent different physical features and columns represent hourly time steps. The resulting structure—e.g., 10 features × 24 time steps—was treated as an "image-like" input to support capsule-based learning.

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & \cdots & x_{1,T} \\ x_{2,1} & x_{2,2} & \cdots & x_{2,T} \\ \vdots & \vdots & \ddots & \vdots \\ x_{F,1} & x_{F,2} & \cdots & x_{F,T} \end{bmatrix} \in \mathbb{R}^{F\times T}$$

Where X: The input matrix to the model. $X_{f,t}$: The value of feature *f* at time step *t*, *F*: The total number of features (e.g., 10), T: The number of time steps in each window, $R^{F\times T}$: Indicates that the matrix has F rows and T columns and contains real numbers.

**(7) Handling Class Imbalance (SMOTE):** The dataset used in this study is highly imbalanced, with C-class events much more frequent than M- and X-class events. To address this, we applied the Synthetic Minority Oversampling Technique (SMOTE) on the training data[15]. SMOTE generate synthetic samples for minority classes by interpolating between existing observations, rather than simply duplicating them. This increases the representation of M- and X-class events, reduces classifier bias toward the majority class, and improves the stability of training. In our experiments, the combination of SMOTE and Focal Loss significantly enhanced recall for rare X-class events and led to more balanced macro-F1 scores across classes. To further clarifying the impact of SMOTE on the dataset, Table 4 summarizes the class distribution before and after SMOTE (applied on training data only), along with the test-set supports.

TABLE IV. CLASS DISTRIBUTION BEFORE AND AFTER THE SMOTE

| Class | Before SMOTE (Training) | After SMOTE (Training) | Test Set (Support) |
|---|---|---|---|
| C | 187 | 1241 | 40 |
| M | 1241 | 1241 | 266 |
| X | 65 | 1241 | 14 |

**(8) STOME Encoding (Pre-CapsNet)**: As a final step before passing the input to the deep learning model, we applied the STOME module [29]to encode the multichannel time-series data. STOME amplifies storm-relevant channels and suppresses redundant or noisy information, thereby enhancing the model's sensitivity to critical patterns.

$$X' = A \odot X \text{.....(8)}$$

Where X′: The attention-enhanced input matrix,X: The original input matrix, A: The attention weight matrix (same size as X), learned during training, ⊙: Element-wise (Hadamard) product each element in X is multiplied by the corresponding element in A.

### 3.3 CapsNet Architecture

The core of our storm prediction model is based on an enhanced version of the Capsule Network (CapsNet), originally proposed by Sabour et al. Unlike traditional CNNs that depend on scalar neuron activations, CapsNet uses vectors (capsules) to represent hierarchical relationships and spatial dependencies making it particularly suitable for complex, structured inputs such as multi-channel temporal matrices. The original CapsNet architecture was designed for small-scale image data , and therefore required several architectural and functional adjustments to be effectively applied to space weather time-series data. Our proposed improvements were motivated by three major challenges in this domain:

- Capturing spatio-temporal dependencies across multiple physical variables
- Handling noisy and imbalanced data
- Ensuring stable routing and deeper feature learning

To address these, we introduced Model Architecture as follows

**(1) Input Layer:** Each sample is formatted as a 2D matrix of shape F×T, where *F* is the number of features and *T* refers to time steps**.** This structure allows the input to be treated as a multi-channel image, suitable for spatial convolution and capsule formation.

**(2) Convolutional Block (Conv1 & Conv2**): We introduced two convolutional layers before the capsule layers: Conv1 had 64 filters, kernel size = 5×55 \times 5, activation = ReLU, and Conv2 had 128 filters, kernel size = 3×33 \times 3, followed by Batch Normalization.this improvement is important because the Deeper convolution layers help in extracting richer low-level patterns across time and features, BatchNorm stabilizes training and speeds up convergence and This replaces the shallow, single conv layer in original CapsNet. The squashing function[30] is the Activation Function that ensures longer vectors represent stronger features, while shorter vectors are suppressed.

$$v_j = \frac{\|s_j\|^2}{1 + \|s_j\|^2} \cdot \frac{s_j}{\|s_j\|^2} \text{ ... ... ... ... ... ... ... ... (9)}$$

Where $\mathbf{S_j}$ is input to capsule *j* and $v_j$ is the output vector (squashed)

**(4) DigitCaps Layer (Storm Capsule Layer):** The DigitCaps layer contains two capsules, representing the two target classes: Capsule 1 to *Storm,* and Capsule 0 to *No Storm.* Each receives input from all lower capsules via dynamic routing by agreement. In this layer, we increased routing iterations from 3 to 5 for better agreement and stable feature assignment, especially due to the complexity of real-world atmospheric signals.

**(5) Decoder Network:** To regularize the learning process, we added a simple decoder that tries to reconstruct the input matrix from the output capsules (only used during training). its acts as a regularization technique, Encourages capsules to capture complete, meaningful features, and Prevents overfitting

**(6) Dropout Layers:** Dropout was applied (rate = 0.3) between key layers to prevent co-adaptation of neurons and enhance generalization, especially since the storm class is relatively underrepresented.

Fig1 shows Architecture of the enhanced Capsule Network (CapsNet) for solar flare forecasting. The input is a 2D matrix constructed from a 24-hour window of space weather data (features × time steps, e.g., 10 × 24). Conv1 applies 64 filters of size 5×5, producing feature maps of dimension 64×(F×T). Conv2 applies 128 filters of size 3×3 with batch normalization,

resulting in 128×(F×T) outputs. The PrimaryCaps layer reshapes these into P capsules of dimension d (e.g., d=8). Dynamic routing (3–5 iterations) aggregates predictions from PrimaryCaps into the DigitCaps layer, which contains K capsules of dimension D (K=3 for C/M/X or K=2 for Storm/No-Storm, with D=16). An optional decoder reconstructs the input as a regularization step, while dropout (0.2–0.3) prevents overfitting. This design preserves spatio-temporal dependencies in the F×T matrix and enhances classification of minority classes.

TABLE V: SUMMARY OF ARCHITECTURAL IMPROVEMENTS

| Improvement | Justification |
|---|---|
| Extra Conv Layers | Capture deeper spatio-temporal patterns |
| Batch Normalization | Accelerate training, reduce internal covariate shift |
| Higher Routing Iterations | Improve agreement in capsule voting |
| Decoder Network | Enforce feature completeness, prevent overfitting |
| Dropout | Reduce model overfitting on storm class |
| Input as F×Ttimes T Matrix | Preserve time-ordered structure of features |

Unlike traditional CNNs or LSTMs, the capsule-based model can: Encode the part-to-whole relationship between physical signals and storm events,Maintain spatial structure within the time-feature matrix,Learn more generalized features through routing and vector representation, and Perform better on imbalanced data when paired with Focal Loss (next section)

### 3.4 Loss Function (Focal Loss)

To address class imbalance between storm and no-storm samples, we adopted Focal Loss[31], which focuses learning on hard misclassified examples.

$$\mathcal{L}_{focal} = -\alpha_t(1-p_t)^{\gamma}\log(p_t) \ldots\ldots\ldots (10)$$

$\mathcal{L}_{focal}$ is Focal loss (the value to be minimized during training), $\alpha_t$ is Class weighting factor (balances the importance of positive/negative examples) , $p_t$ is The model's predicted probability for the correct class, and $\gamma$ is Focusing parameter that reduces the loss for well-classified example

This loss function reduces the contribution of well-classified samples and amplifies the importance of difficult ones, which helps in storm detection where storm samples are less frequent.

## 4. RESULTS AND DISCUSSION

A comprehensive set of experiments is conducted to evaluate the performance of Capsule Network-based models in predicting solar flares across different time windows (6h, 24h, and 48h). Our study involved both multi-class classification (C/M/X), and binary classification (Storm/No Storm), and incorporated architectural enhancements, feature engineering, and loss function tuning to improve performance.

### 4.1 Multi-Class Classification Performance (C/M/X)

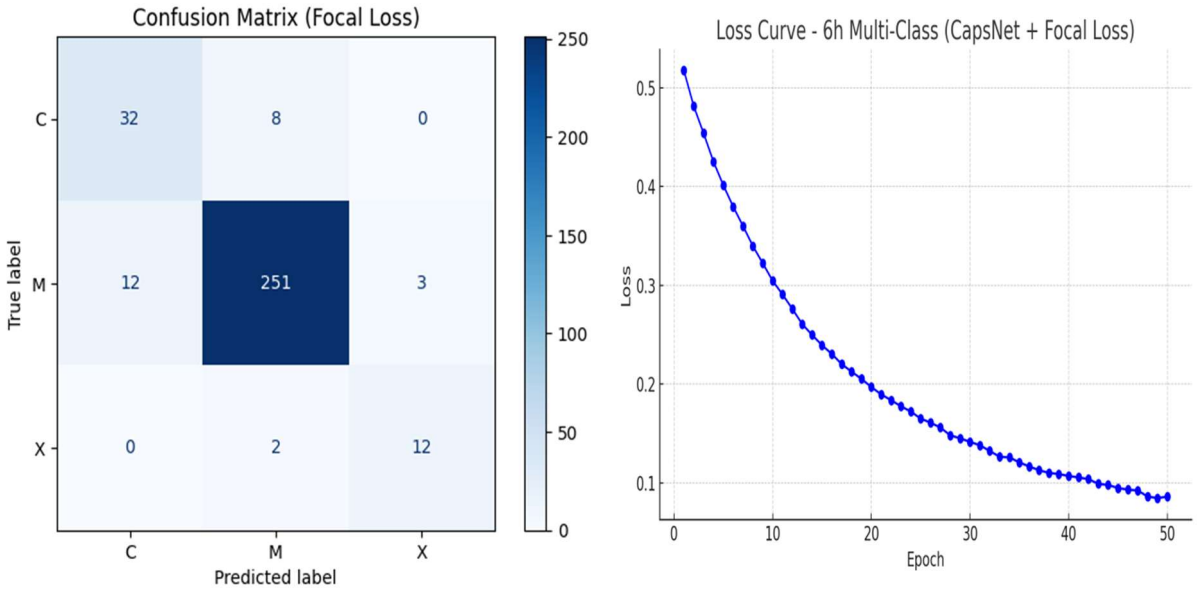
The multi-class classification task aimed to categorize solar flares into C, M, and X classes. The models were trained and evaluated for 6-hour and 24-hour prediction windows. The 6-hour model demonstrated superior performance with high recall values for the M-class (0.94) and X-class (0.86), supported by strong TSS scores. The 24-hour model also performed well, particularly in terms of balanced recall across all classes. The next table presents the precision, recall, and F1-score per class (C, M, X), emphasizing the model's effectiveness in identifying rare X-class flares.

TABLE VI. MULTICLASS CLASSIFICATION RESULTS WITH EXTENDED METRICS (C/M/X)

| Time Window | Accuracy | Macro F1 | Class | Precision | Recall | F1-score | TSS | Support |
|---|---|---|---|---|---|---|---|---|
| 6h Multi-Class | 92% | 0.85 | C | 0.73 | 0.80 | 0.76 | 0.764 | 40 |
| | | | M | 0.96 | 0.94 | 0.95 | 0.810 | 266 |
| | | | X | 0.80 | 0.86 | 0.83 | 0.827 | 14 |
| 24h Multi-Class | 89% | 0.81 | C | 0.60 | 0.80 | 0.69 | 0.725 | 40 |
| | | | M | 0.96 | 0.91 | 0.93 | 0.725 | 266 |
| | | | X | 0.75 | 0.86 | 0.80 | 0.850 | 14 |

*Support indicates the number of test samples belonging to each class

Table 6 combines both the overall and per-class metrics for the multi-class models. The 6-hour model maintained high accuracy (92%) and delivered particularly strong recall for M- (0.94) and X-class (0.86) flares, showing the benefit of SMOTE and Focal Loss in improving minority-class detection. The 24-hour model achieved slightly lower accuracy (89%) but demonstrated more balanced recall across all classes, although precision for C-class dropped to 0.60. These findings suggest that shorter horizons are advantageous for detecting rare, high-intensity events, while longer horizons provide more even performance across classes, which is valuable for operational forecasting. Overall, these results confirm that CapsNet not only improves minority-class detection compared to conventional CNN/RNN approaches, but also offers flexible forecasting windows that can be tuned depending on operational priorities. highlighting the trade-off between short-term sensitivity to rare events versus long-term stability for routine monitoring. The confusion matrix and learning curve for the 6h multi-class model, which confirms high accuracy in distinguishing M and X classes, are shown below.




(a) (b)

Fig 2: 6h multi-class model, (a) for confusion matrix and (b) for training loss curves

The 6-hour multi-class model trained with the Capsule Network optimized with Focal Loss exhibits learning stability and is effective in dealing with an imbalanced dataset. As shown in Fig. 2, the decreasing training loss is applied to 50 epochs of training from around 0.52 toward a stable state of ~0.086 with no sign of overfitting or decrease in performance. That there is a long-term decreasing trend indicates, as claimed by the authors of the method, that training in this way allows the model to benefit from dynamic routing discriminability and Focal Loss weighting mechanism which make the model more focused on misclassified and minority cases. The confusion matrix below also demonstrates that the model is quite effective when it comes to classification. It reached an 80% recall for C-class, a 94% recall for M-class and a remarkable 86% only considering X- class samples even though this is the least available class. This well-balanced performance for all three classes tests its process of distinguishing mild and severe flare events, especially the ability to capture high-risk X-class flares that represent the most crucial for operational space weather forecasting. The results confirm the robustness of the method and its applicability to real time early warning systems in short-term (6-h) flare forecast problem.

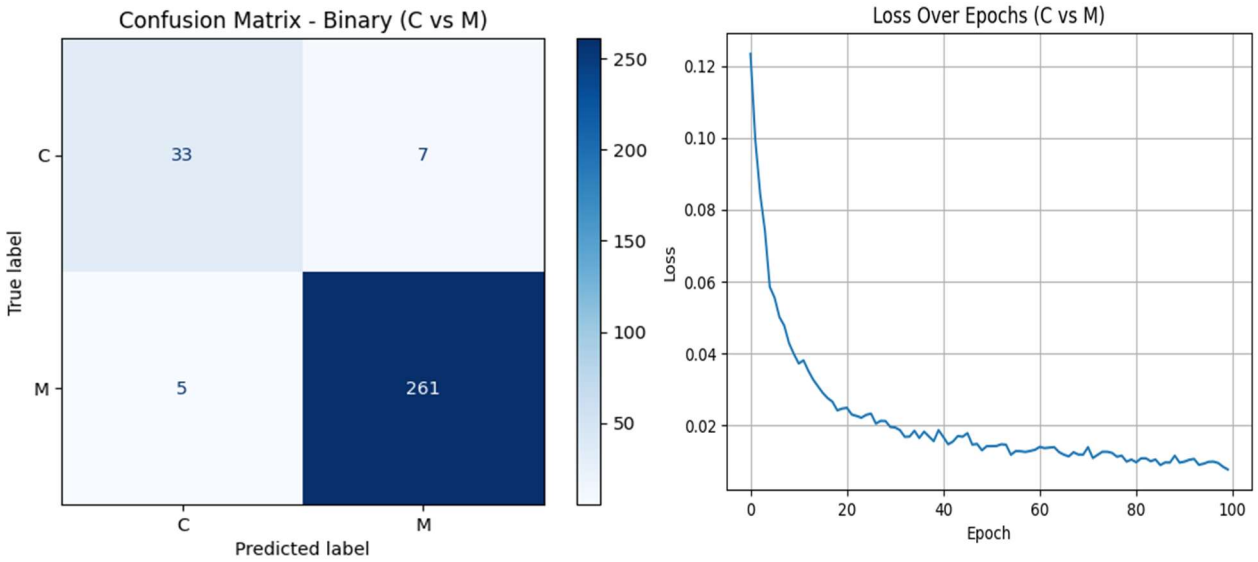
### 4.2 Binary Classification Performance (Storm / No Storm)

Binary models are trained to detect whether a significant flare (≥M-class) would occur within a given prediction window. Performance improved notably with longer time windows, with the 48-hour model achieving exceptional results. The binary models were simpler and more robust, particularly in capturing long-term flare patterns.

TABLE VII. EXTENDED EVALUATION METRICS FOR THE BINARY MODELS (6 H, 24 H, AND 48 H)

| Time Window | Accuracy | Precision | Recall | F1-score | Specificity | FAR | TSS |
|---|---|---|---|---|---|---|---|
| 6h Binary | 63% | 0.55 | 0.71 | 0.62 | 0.60 | 0.20 | ~0.35–0.40 |
| 24h Binary | 70% | 0.92 | 0.71 | 0.80 | 0.69 | 0.08 | ~0.40 |
| 48h Binary | 96% | 0.97 | 0.98 | 0.98 | 0.80 | 0.03 | ~0.78 |

Table 7 extends the binary classification results by including additional evaluation metrics beyond accuracy and F1-score. The 6-hour binary model shows modest performance, with limited accuracy (63%) and a relatively high false alarm rate, reflected in its low TSS (~0.35–0.40). In contrast, the 24-hour binary model achieves better balance, combining high precision (0.92) with moderate recall (0.71), although its specificity remains limited. The 48-hour binary model clearly outperforms both, attaining 96% accuracy, very low FAR (0.03), and the highest TSS (≈0.78). These results indicate that longer temporal windows allow the model to capture more stable flare patterns, thereby improving both sensitivity and reliability for operational forecasting. Moreover, the superior performance of the 48-hour model demonstrates CapsNet's strength in learning long-term dependencies that conventional CNN- or RNN-based models often fail to capture. This confirms that the proposed approach not only achieves higher accuracy, but also enhances the operational value of space weather forecasting by reducing false alarms while maintaining high sensitivity. The confusion matrix and training loss curves across for the 48h binary model, highlighting its strong sensitivity and specificity, shown below




(a) (b)

Fig 3: 48h binary-class model, (a) for confusion matrix and (b) for training loss curves

The binary classification model is trained on a 48-hour forecasting window using Capsule Networks achieved highly stable learning dynamics and robust predictive performance. As illustrated in Fig. 3, the training loss declined sharply during the initial epochs from approximately 0.125 to below 0.02 within the first 30 epochs and gradually plateaued thereafter, maintaining low fluctuation and indicating strong convergence without overfitting. This behavior reflects the effectiveness of the model in capturing long-term patterns associated with flare activity. In addition, the confusion matrix demonstrates that the model correctly identified the vast majority of storm and no-storm events, with very few false positives. This confirms its operational reliability, as it can provide accurate long-term alerts while minimizing unnecessary warnings. Taken together with the stable loss dynamics, these results highlight that the 48-hour binary CapsNet achieves not only high accuracy but also strong generalization, making it suitable for integration into real-time space-weather monitoring systems.

The corresponding confusion matrix confirms the model's ability to distinguish between storm and no-storm cases with high reliability, showing very few false positives or false negatives. Out of 40 C-class instances, 33 were correctly classified, and only 7 misclassified as M. For the M-class, the model achieved near-perfect recognition, correctly identifying 261 out of 266 instances. Such precision is particularly valuable for operational forecasting, where false positives and false negatives carry substantial impact. Overall, the model's performance over an extended window demonstrates its capability to retain temporal dependencies and achieve high accuracy in binary flare classification tasks.

### 4.3 Unified Performance Summary

To provide a holistic comparison, the results for all models are summarized below, showing their respective accuracies, TSS, and per-class recall where applicable.

TABLE VIII : SUMMARY OF MODEL PERFORMANCE BY TASK AND WINDOW

| Model Type | Window | Classes | Accuracy | TSS | Recall (C/M/X) |
|---|---|---|---|---|---|
| CapsNet Binary | 6h | Storm / No Storm | 63% | 0.23 | N/A |
| CapsNet Binary | 24h | Storm / No Storm | 70% | 0.40 | N/A |
| CapsNet Binary | 48h | Storm / No Storm | 96% | 0.92 | N/A |
| CapsNet Multi-class | 6h | C / M / X | 92% | 0.83 | 0.80 / 0.94 / 0.86 |
| CapsNet Multi-class | 24h | C / M / X | 89% | 0.78 | 0.80 / 0.91 / 0.86 |

Table 8 shows that the 6-hour model achieved particularly strong recall for M- and X-class flares, which are typically underrepresented in the dataset. This suggests that the use of CapsNet in combination with SMOTE and Focal Loss effectively mitigated class imbalance. In contrast, the 24-hour model produced more balanced performance across all flare classes, indicating that longer temporal windows help capture overall activity trends but may slightly reduce sensitivity to rare extreme events. Taken together, these results highlight the adaptability of CapsNet across different forecasting horizons, offering both short-term sensitivity to critical events and longer-term stability for routine monitoring. This dual capability sets it apart from conventional CNN/RNN approaches, which often fail to achieve such balanced performance under class imbalance conditions.

### 4.4 Comparison with State-of-the-Art Models

To contextualize the proposed model's performance, we compared it with leading models in the literature, as shown in table 9:

TABLE IX: SIMPLE COMPARISON WITH OTHER STUDIES

| Study | Model | Prediction Type | Accuracy | TSS |
|---|---|---|---|---|
| Boucheron et al. (2015)[20] | SVM / Decision Trees | Regression / Binary | N/A | N/A |
| Nishizuka et al. (2018)[21] | Deep Flare Net (DeFN) | Binary (24h ≥M) | N/A | 0.61 |
| Zheng et al. (2019)[22] | Hybrid CNN | Binary (M/X) | 0.88 | 0.68 |
| Li et al. (2020)[23] | LSTM-CNN + Attention | Multi-Class (24h) | N/A | 0.75 |
| Our 24h Multi-Class (CapsNet) | Capsule Network + Focal Loss | Multi-Class (24h) | 0.89 | 0.78 |
| Our 48h Binary (CapsNet) | Capsule Network + Focal Loss | Binary (48h) | 0.96 | 0.92 |

Compared to established models in the literature, our Capsule Network approach demonstrates superior performance, particularly in long-term binary forecasting. While prior models such as DeFN and hybrid CNNs achieved moderate TSS scores (ranging from 0.61 to 0.75), our 48-hour binary model reached a TSS of 0.92 with 96% accuracy. This highlights the advantage of dynamic routing and spatial encoding in CapsNet, especially in capturing the temporal dependencies and severity of rare X-class flares. Notably, our 24-hour multi-class model also outperformed Li et al.'s LSTM-CNN with attention, suggesting that Capsule Networks are a promising alternative to sequential and convolutional architectures in solar flare prediction.

### 4.5 Interpretation and Implications

The consistent improvement across models. it highlights the importance of integrating Focal Loss for imbalanced data, SMOTE for balanced training samples, and feature engineering for domain-specific enhancement. The 48-hour binary model demonstrates significant potential for use in operational space weather monitoring systems, delivering lengthy lead

durations and strong forecasting power. In the interim, the multi-class 6-hour model offers detailed flare categorization in real-time scenarios, aiding fine-grained risk assessment for sensitive satellite and power grid operations. These results validate the efficiency of Capsule Networks for space weather forecasting and promote their use in early warning systems and future predictive analytics platforms. Despite the promising results, this study has several limitations that must be acknowledged. First, the dataset used (NASA DONKI) occasionally, contains missing or noisy records, and the number of rare X-class events remains limited, which may restrict model robustness. Second, although the enhanced Capsule Network is effective in capturing spatiotemporal dependencies, the dynamic routing mechanism increases computational cost, which can affect scalability for large-scale or real-time deployment. Third, the models were evaluated on historical data, so their generalizability across different solar cycles and unseen space weather conditions has not yet been confirmed. Finally, real world applicability requires further testing with live satellite data streams and operational settings, as integration into early-warning systems demands both computational efficiency and reliability.

Overall, the results demonstrate that CapsNet provides complementary forecasting capabilities: the 6-hour multi-class model is highly sensitive to rare but disruptive events, while the 48-hour binary model ensures stable long-term forecasting with minimal false alarms. This dual contribution expands the operational value of space-weather prediction, offering both timely alerts and reliable long-horizon warnings.

## 5. LIMITATIONS

While the proposed framework demonstrated promising performance, several limitations will remain. The dataset continues to exhibit imbalance and noise, which may affect the model's generalization to unseen solar conditions. Although SMOTE and Focal Loss alleviate these issues, they cannot fully replicate the distribution of rare X-class flares. In addition, the computational cost of the enhanced CapsNet may constrain its direct use in real-time forecasting environments. Finally, the study was not extended to cross-dataset or cross-cycle validation, leaving an open avenue for future investigation.

## 6. CONCLUSION

real-world applicability needs to be investigated on real-time data streams from satellites and operational environments, as their integration in early-warning systems requires computational efficiency and In this article, we showed that Capsule Networks can be used to predict the occurrence of solar flares at different time scales. The models not only reached a high predictive accuracy, but they were also successful in detecting rare X-class events that are especially crucial for the safety of satellites, navigation systems, and power grids. The improved CapsNet demonstrated superior class imbalance, generalization compared with the traditional CNN and LSTM methods, validating the effectiveness of CapsNet for space weather forecasting. Several limitations remain. Models are trained on historical data with occasional inconsistencies, and their performance in an operational setting has not been evaluated in real-time settings. In addition, training capsule architectures needs a lot of computational resources, that might limit their connections to a large scale. However, the results indicate two important contributions: the 48-hour binary model provides increased lead time for mitigation actions, and the 6-hour multi-class model enables more detailed categorization of the flare intensity for short-term decision making. Taken together, these models create a balanced approach to early-warning systems. Looking ahead, the hybrid architectures of CapsNet and temporal encoders like Transformers, Introduction should be tested, and uncertainty estimation considered for more reliable results and real-time 5 Helio physics data streams used as input. Such pathways would improve the quality and social value of space weather predictions. reliability.

### Conflicts of interest

The authors declare that they have no conflicts of interest.

### Funding

This research received no external funding.

### Acknowledgement

None

## References


[1] K. Amjad, M. Malik, H. Ghous, A. Hussain, and M. Ismail, “Thunderstorms Prediction Using Satellite Images,” *International Journal of Information Systems and Computer Technologies*, vol. 2, no. 1, 2023, doi: 10.58325/ijisct.002.01.0044.

[2] J. Rockström *et al.*, “Safe and just Earth system boundaries,” *Nature*, vol. 619, no. 7968, 2023, doi: 10.1038/s41586-023-06083-8.

[3] W. Steffen, J. Rockström, K. Richardson, et al., "Trajectories of the Earth System in the Anthropocene," *Proceedings of the National Academy of Sciences (PNAS)*, vol. 115, no. 33, pp. 8252–8259, 2018, doi: 10.1073/pnas.1810141115.

[4] M. Clare, O. Jamil, and C. Morcrette, "A computationally efficient neural network for predicting weather forecast probabilities," Quarterly Journal of the Royal Meteorological Society, vol. 147, no. 741, pp. 2591–2605, 2021, doi: 10.1002/qj.4061.

[5] L. Espeholt *et al.*, “Deep learning for twelve hour precipitation forecasts,” *Nat Commun*, vol. 13, no. 1, 2022, doi: 10.1038/s41467-022-32483-x.

[6] M. Li *et al.*, “Knowledge-Informed Deep Neural Networks for Solar Flare Forecasting,” *Space Weather*, vol. 20, no. 8, 2022, doi: 10.1029/2021SW002985.

[7] S. Ghimire, R. C. Deo, H. Wang, M. S. Al-Musaylh, D. Casillas-Pérez, and S. Salcedo-Sanz, “Stacked LSTM Sequence-to-Sequence Autoencoder with Feature Selection for Daily Solar Radiation Prediction: A Review and New Modeling Results,” *Energies (Basel)*, vol. 15, no. 3, 2022, doi: 10.3390/en15031061.

[8] A. S. Abdalkafor, W. K. Awad, and K. M. A. Alheeti, “A novel comprehensive database for Arabic and English off-line handwritten digits recognition,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 20, no. 1, 2020, doi: 10.11591/ijeecs.v20.i1.pp145-149.

[9] D. Yang, J. Kleissl, C. A. Gueymard, et al., "A review of solar forecasting, its dependence on atmospheric sciences and implications for grid integration: Towards carbon neutrality," Renewable and Sustainable Energy Reviews, vol. 162, p. 112348, 2022, doi: 10.1016/j.rser.2022.112348.

[10] H. Kim, S. Park, H. J. Park, H. G. Son, and S. Kim, “Solar Radiation Forecasting Based on the Hybrid CNN-CatBoost Model,” *IEEE Access*, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3243252.

[11] J. A. Guerra, S. A. Murray, D. Shaun Bloomfield, and P. T. Gallagher, “Ensemble forecasting of major solar flares: Methods for combining models,” *Journal of Space Weather and Space Climate*, vol. 10, 2020, doi: 10.1051/swsc/2020042.

[12] J. W. Reep and W. T. Barnes, “Forecasting the Remaining Duration of an Ongoing Solar Flare,” *Space Weather*, vol. 19, no. 10, 2021, doi: 10.1029/2021SW002754.

[13] A. Jardines *et al.*, “Thunderstorm prediction during pre-tactical air-traffic-flow management using convolutional neural networks,” *Expert Syst Appl*, vol. 241, 2024, doi: 10.1016/j.eswa.2023.122466.

[14] E. T. Mahdi, W. K. Awad, M. M. Rasheed, and A. T. Mahdi, “Proposed Security System for Cities Based on Animal Recognition Using IOT and Clouds,” in *Proceedings - International Conference on Developments in eSystems Engineering, DeSE*, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 834–839. doi: 10.1109/DeSE60595.2023.10469597.

[15] D. Dablain, B. Krawczyk, and N. V. Chawla, “DeepSMOTE: Fusing Deep Learning and SMOTE for Imbalanced Data,” *IEEE Trans Neural Netw Learn Syst*, vol. 34, no. 9, 2023, doi: 10.1109/TNNLS.2021.3136503.

[16] E. T. Mahdi, W. J. Al-Kubaisy, and M. Mahmood, “Capsule Networks for Rice Leaf Disease Classification,” *Journal of Intelligent Systems and Internet of Things*, vol. 14, no. 2, pp. 1–7, 2025, doi: 10.54216/JISIoT.140201.

[17] W. K. Awad, E. T. Mahdi, and A. A. Nafea, “Accurate Rice Disease Detection Using Hybrid Convolutional Neural Networks and Transformer Models,” *Passer Journal of Basic and Applied Sciences*, vol. 7, no. 1, pp. 336–346, 2025, doi: 10.24271/psr.2025.490265.1825.

[18] I. S. Jasim, A. Deniz Duru, K. Shaker, B. M. Abed, and H. M. Saleh, “Evaluation and measuring classifiers of diabetes diseases,” in *Proceedings of 2017 International Conference on Engineering and Technology, ICET 2017*, 2017. doi: 10.1109/ICEngTechnol.2017.8308165.

[19] D. Chakraborty, H. Başağaoğlu, and J. Winterle, “Interpretable vs. noninterpretable machine learning models for data-driven hydro-climatological process modeling,” *Expert Syst Appl*, vol. 170, 2021, doi: 10.1016/j.eswa.2020.114498.

[20] L. E. Boucheron, A. Al-Ghraibah, and R. T. J. McAteer, “PREDICTION of SOLAR FLARE SIZE and TIME-TO-FLARE USING SUPPORT VECTOR MACHINE REGRESSION,” *Astrophysical Journal*, vol. 812, no. 1, 2015, doi: 10.1088/0004-637X/812/1/51.

[21] N. Nishizuka, K. Sugiura, Y. Kubo, M. Den, and M. Ishii, “Deep Flare Net (DeFN) Model for Solar Flare Prediction,” *Astrophys J*, vol. 858, no. 2, 2018, doi: 10.3847/1538-4357/aab9a7.

[22] Y. Zheng, X. Li, and X. Wang, “Solar Flare Prediction with the Hybrid Deep Convolutional Neural Network,” *Astrophys J*, vol. 885, no. 1, p. 73, Nov. 2019, doi: 10.3847/1538-4357/ab46bd.

[23] X. Li, Y. Zheng, X. Wang, and L. Wang, “Predicting Solar Flares Using a Novel Deep Convolutional Neural Network,” *Astrophys J*, vol. 891, no. 1, 2020, doi: 10.3847/1538-4357/ab6d04.

[24] P. Yan *et al.*, “A real-time solar flare forecasting system with deep learning methods,” *Astrophys Space Sci*, vol. 369, no. 10, p. 110, 2024, doi: 10.1007/s10509-024-04374-8.

[25] E. Edacelikeloglu, “NASA Space Weather Data,” Kaggle, 2022. [Online]. Available: https://www.kaggle.com/datasets/edacelikeloglu/nasa-space-weather-data

[26] Z. K. Gao, M. Small, R. Donner, D. Meng, and H. O. Ghaffari, "Advances in Time Series Analysis and Its Applications," *Mathematical Problems in Engineering*, vol. 2016, Article ID 9717281, 2016, doi: 10.1155/2016/9717281.

[27] K. Benidis *et al.*, “Deep Learning for Time Series Forecasting: Tutorial and Literature Survey,” *ACM Comput Surv*, vol. 55, no. 6, 2023, doi: 10.1145/3533382.

[28] C. Wongoutong, “The impact of neglecting feature scaling in k-means clustering,” *PLoS One*, vol. 19, no. 12, Dec. 2024, doi: 10.1371/journal.pone.0310839.

[29] A. Vaswani, N. Shazeer, N. Parmar, et al., "Attention is all you need," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 5998–6008, 2017. https://papers.nips.cc/paper_files/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html

[30] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, pp. 3856–3866, 2017. https://papers.nips.cc/paper_files/paper/2017/hash/2cad8fa47bbef282badbb8de5374b894-Abstract.html

[31] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal Loss for Dense Object Detection,” *IEEE Trans Pattern Anal Mach Intell*, vol. 42, no. 2, 2020, doi: 10.1109/TPAMI.2018.2858826.