Solar Flare Recurrence Prediction & Visual Recognition

by

Mangaliso Moses Mngomezulu (219030015)

Supervisor: Dr. Mandlenkosi V. Gwetu

Co-Supervisor: Dr. Jean V. Fonou-Dombeu

A dissertation submitted in fulfillment of the requirement for the degree of

Master of Science in Computer Science

School of Mathematics Statistics and Computer Science

University Of KwaZulu-Natal

Pietermaritzburg 3201, South Africa

December 2023

Declaration - Authorship

I, **Mangaliso M. Mngomezulu**, declare that;

1. The research reported in this thesis, except where otherwise indicated, is my original work.
2. This thesis has not been submitted for any degree or examination at any other university.
3. This thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
   (a) Their words have been re-written but the general information attributed to them has been referenced,
   (b) Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the thesis and in the references sections.

Signe[redacted].....Date.....27 March 2024.....

Mangaliso M. Mngomezulu

Declaration - Supervisor & Co-Supervisor

As the candidate's supervisor, I have approved this thesis for submission.

Signed ....[redacted]..... Date .....28 March 2024.....

Dr. Mandlenkosi V. Gwetu

As the candidate's co-supervisor, I have approved this thesis for submission.

Signed ....[redacted].... Date .....27/03/2024.....

Dr. Jean V. Fonou-Dombeu

Declaration - Publications & Awards

Details of contribution to publications and research awards that form part and/or include research presented in this dissertation are as follows.

1. **Publication 1** M. Mngomezulu, M. Gwetu, and J. V. Fonou-Dombeu, "Solar flare forecasting using individual and ensemble rnn models," in International Conference on *Innovative Techniques and Applications of Artificial Intelligence*. Springer, 2023, pp. 307–320. DOI: https://doi.org/10.1007/978-3-031-47994-6_29)
2. **Award 1** RNN Ensemble Calibration for Solar Flare Forecasting (Awarded 3^{rd} place at the UKZN's Postgraduate Research & Innovation Symposium (PRIS) 2023. Link: https://drive.google.com/file/d/219030015Award (including Masters & Doctoral candidates)
3. **Award 2** Ensemble CNNs For Solar Event Classification (Awarded 1^{st} place research project prize at the University Of KwaZulu-Natal's Award Ceremony in 2023. The name of the award is the Remi Adewumi Memorial Award)

Signed:..[redacted].....

Mangaliso M. Mngomezulu

Dedication

Dedicated to all who have been supportive before and during this journey. Thank you!

Abstract

Solar flares are intense outbursts of radiation observable in the photosphere. The radiation flux is measured in W/m^2. Solar flares can kill astronauts, disrupt electrical power grids, and interrupt satellite-dependent technologies. They threaten human survival and the efficiency of technology. The reliability of solar flare prediction models is often undermined by the stochastic nature of solar flare occurrence as shown in previous studies. The Geostationary Operational Environmental Satellite (GOES) system classifies solar flares based on their radiation flux. This study investigated how Recurrent Neural Network (RNN) models compare to their ensembles when predicting flares that emit at least $10^{-6}W/m^2$ of radiation flux, known as $\geq$C class flares. A Long-Short Term Memory (LSTM) and Simple RNN homogeneous ensemble achieved a similar performance with a tied True Skill Statistic (TSS) score of $70 \pm 1.5\%$. Calibration curves showed that ensembles are more reliable. The balanced accuracies of the Simple RNN Ensemble and LSTM are both 85% with f1-scores of 79% and 77% respectively. Furthermore, this study proposed a framework that shows how objective function reparameterization can be used to improve binary ($\geq$C or $<$C) categorical predictions of ensemble RNNs under uncertainty in the stochastic solar flare environment. The best-calibrated ensemble (Heterogenous Stacking Ensemble) had a TSS score of $65 \pm 1\%$, balanced accuracy of 83%, and an f1-score of 81%. Almost perfect calibration usually comes with a trade-off on some metrics, e.g., the TSS as seen in other studies. It is a fact that solar flares erupt from magnetically active regions. Visual manifestations of solar flares can be categorized into various morphological classes which are linked with the underlying magnetic field. Moreover, this study demonstrated how ensemble Convolutional Neural Networks (CNNs) can be used to improve visual recognition of solar flares observed at wavelength 1,600$\overset{\circ}{A}$. Base learner diversity was used to improve the likelihood of ensemble success. Classification improved from 94% to $99.99 \pm 0.01\%$ when compared to the only preceding study that used CNNs. Overall, this study demonstrated how base learners can be set up to improve ensemble performance in the context of solar flare predictions.

Acknowledgements

I would like to thank my supervisor Dr. M. V. Gwetu and co-supervisor Dr. J.V Fonou-Dombeu respectively for their enormous support and guidance during the project. I have learned and grown a lot under their excellent supervision. I am grateful to the School of Computer Science at the University of KwaZulu-Natal for the resources and environment that made this work possible. I would like to thank my casual mentors Mr. Andile M. Gumede (MSc Computer Science & Senior Software Engineer) and Mr. Ntokozo S. Khuzwayo (Mathematics PhD Candidate & Junior Lecturer) for sharing their wisdom with me.

Contents

List of Figures

List of Figures

List of Tables

List of Tables

List of Acronyms

SDO	Solar Dynamics Observatory
HMI	Helioseismic and Magnetic Imager
SHARP	Space weather HMI Active Region Patches
GOES	Geo-stationary Operational Environment Satellites
AIA	Atmospheric Imaging Assembly
AR	Active Region
RNN	Recurrent Neural Network
LSTM	Long-Short Term Memory
HSE	Homogenous Stacking Ensemble
HtrSE	Heterogenous Stacking Ensemble
GRU	Gated Recurrent Unit
CNN	Convolutional Neural Network
GS	Gumbel-Softmax
TSS	True Skill Statistic
MPI	Message Passing Interface
W/m^2	Watts per meter squared

Preface

The research discussed in this thesis was carried out in the College of Agriculture, Engineering, and Science of the University of Kwa-Zulu Natal, Pietermaritzburg, from January 2023 until December 2023 by Mangaliso Mngomezulu under the supervision of Dr. Mandlenkosi V. Gwetu and co-supervision by Dr. Jean V. Fonou-Dombeu.

As the candidate's supervisor, I, Mandlenkosi Gwetu, agree to the submission of this thesis.

Signed:MGwetu..... Date:28 March 2024.....

As the candidate's co-supervisor, I, Jean V. Fonou-Dombeu, agree to the submission of this thesis.

Signed:..... Date:27/03/2024.....

I, Mangaliso Mngomezulu, hereby declare that all the material incorporated in this thesis is my own original work, except where acknowledgment is made by name or in the form of a reference. The work contained herein has not been submitted in any form for any degree or diploma to any other institution.

Signed:..... Date:27 March 2024.....

The University of KwaZulu-Natal, March 27, 2024

Chapter 1: Introduction

1 Motivation

Solar flares are rapid electromagnetic radiation energy outbursts that take place on the solar surface [1, 2]. Extreme Ultraviolet (EUV) radiation, gamma rays, and X-rays are produced in the process. In the entire solar system, solar flares are distinguished by being associated with the highest energy release and being one of the major influences in space weather [3]. They can release at most about 10^{32} erg (a unit of energy measure equivalent to 10^{-7} Joules, hence 10^{32}erg $\approx 10^{25}$ Joules) in a duration of milliseconds to hours spanning a geographical area of hundreds of kilometers [4]. Solar flares release electromagnetic radiation (largely in the form of X-rays) when they take place (measured in Watts per square meter, W/m^2) [5]. A measure of the energy is used by the Geostationary Operational Environmental Satellite(GOES) classification system to classify the solar flares based on the intensity. The common classes are A, B, C, M, and X [6]. While some flares are of less significance e.g., A & B class solar flares, some are cataclysmic and more eruptive e.g., solar flare classes C, M, & X. An example of a medium-intensity flare can be seen in Figure 1.1. Such images are taken by the Solar Dynamics Observatory (SDO) rocket, see Figure 1.2. Some technologies used on Earth are dependent on technology instruments in space. The occurrence of solar flares can disrupt those technologies, e.g., aeronautical navigation, global positioning systems (GPS), and power supply grids [7]. Being able to precisely forecast solar flares can help with minimizing their impact. One of the main challenges associated with solar flare forecasting is that the occurrence of solar flares is a stochastic process [8]. Lethal solar events e.g., Coronal Mass Ejections (CMEs) are triggered by strong solar flares [9]. CMEs threaten human extinction on Earth. Solar flares also cause ionospheric heating which can lead to inaccurate information from satellites due to disrupted wave propagation [2]. Based on approximations, a major flare can result in a loss of US $\$163,000,000,000.00$ ($\approx R3,056,908,534,521.05$ in South African Rand (ZAR) [10]) in North America alone [11]. Global financial estimates make solar flares a significant area of attention.

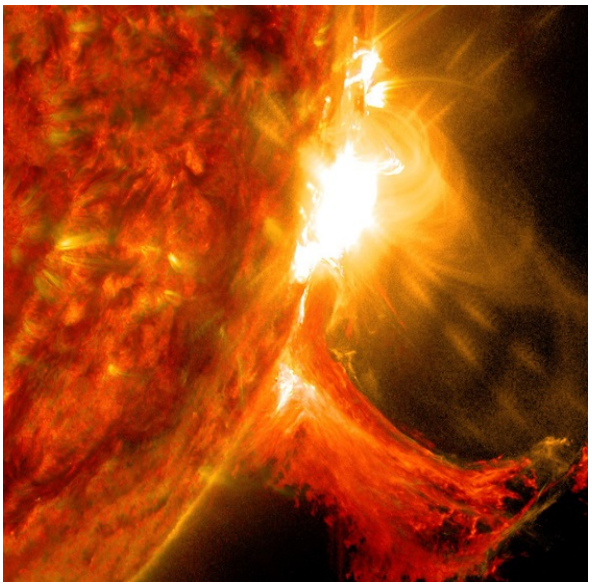

Fig. 1.1: A Solar Flare In Action [12]

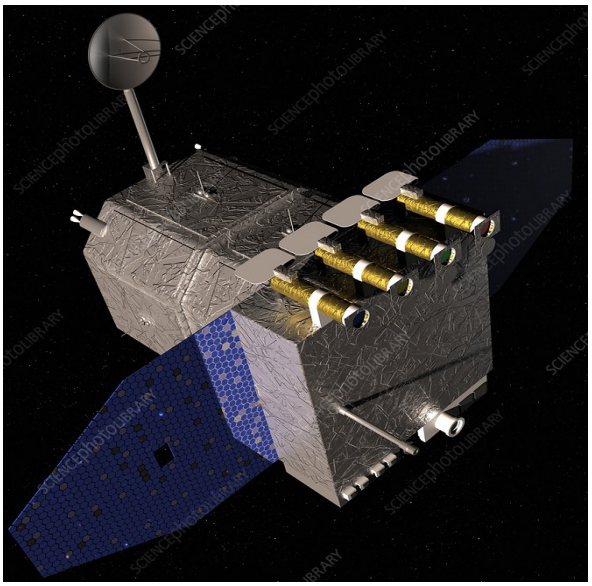

Fig. 1.2: A Photograph of the SDO Rocket [13]

2 Problem Statement

The factors or observations that pre-determine if a solar flare will occur or not and if it does occur how intense its radioactive energy will be are not known [2, 7, 14–16]. The sun has slightly darker regions where magnetic field activity is intense [4]. Those regions are called sunspots and normally form in clusters called active regions (ARs). ARs are known to precede co-located solar flare eruptions [5], but the presence of an AR does not predetermine a forthcoming solar flare. The occurrence of solar flares is mainly stochastic [6]. Most present data is based on ARs [5, 7, 11, 14, 17]. Existing solutions based on Recurrent Neural Networks (RNNs) that attempted to predict the recurrence of $\geq$C solar flares have shown poor calibration [7]. This is partly due to the stochastic nature of solar flare occurrence. The various morphological expressions of solar flare ribbons are closely linked with the magnetic field whose data is used for solar flare recurrence prediction [18]. Proper classification of flare ribbons can allow solar physicists to link the solar flare ribbon geometrical properties with the solar magnetic field structures to produce enriched solar flare data. Perfect or near perfect classification has not been achieved yet [18] when visually classifying images taken at wavelength 1,600$\mathring{A}$. Based on these facts this study attempts to answer the following questions:

1. How can the reliability of the RNN models predicting the recurrence of $\geq$C class solar flares be improved under uncertainty?

2. How can denoising line of sight magnetograms and cropping regions of interest compare to raw magnetograms when they are both utilized for $\geq$C solar flare recurrence prediction using the

Convolutional Long-Short Term Memory model?

3. How can solar flare visual recognition be improved for images taken at observations of wavelength 1,600$\overset{\circ}{A}$ when using Convolutional Neural Networks?

3 Research Aim and Objectives

The aim of this study is to investigate Deep Learning solutions for predicting Solar Flare recurrence and its visual recognition. Secondary objectives include:

1. To explore the influence of using ensembles in improving the certainty of RNN models predicting the recurrence of $\geq$C class solar flares.
2. To investigate approaches related to modification or reparameterization of the RNN models' objective function for improved binary categorical ($\geq C$ or$< C$) solar flare predictions under uncertainty.
3. To investigate the impact of denoising and region of interest cropping on line of sight magnetograms then utilise the resulting pre-preprocessed magnetograms for $\geq C$ solar flare recurrence prediction using a Convolutional Long-Short Term Memory (ConvLSTM) model.
4. To explore the effectiveness of individual and ensemble Convolutional Neural Networks (CNNs) in visually recognizing solar flare ribbons observed at wavelength 1,600$\overset{\circ}{A}$

4 Research Hypotheses

It is mandatory for a comprehensive research study to constitute a research hypothesis [19]. The hypothesis must be linked with the research question [20]. This study aims to address more than one question. Based on these facts, this dissertation presents the following research hypotheses:

4.1 On Recurrence Prediction of $\geq$C Solar Flares

$\mathbf{H_0^1}$ The use of Ensemble RNNs when predicting $\geq$ C solar flares shows no improvement in contrast to the best calibrated RNN base learner

$\mathbf{H_1^1}$ The use of Ensemble RNNs when predicting $\geq$ C solar flares shows improved calibration in contrast to the best calibrated RNN base learner

4.2 On The Effectiveness Of Reparameterization Tricks

$\mathbf{H_0^2}$ Injecting constrained randomness in the binary predictions ($\geq$C and $<$C) made by RNN base learners does not increase generalization and or calibration of the ensemble RNN

$\mathbf{H_1^2}$ Injecting constrained randomness in the binary predictions ($\geq$C and $<$C) made by RNN base learners increases generalization and hence calibration of the ensemble RNN

4.3 On Line of Sight Magnetogram Preprocessing

$\mathbf{H_0^3}$ Denoising line of sight magnetograms by means of thresholding then closing and opening (morphological operations) and then cropping a smaller region of interest (where there is the largest darkish or whitish spot) does not increase the predictability of $\geq$C class solar flares when using the Convolutional Long-Short Term Memory model.

$\mathbf{H_1^3}$ Denoising line of sight magnetograms by means of thresholding then closing and opening (morphological operations) and then cropping a smaller region of interest (where there is the largest darkish or whitish spot) increases predictability of $\geq$C class solar flares when using the Convolutional Long-Short Term Memory model.

4.4 On Visual Recognition At Wavelength 1,600$\mathring{A}$

$\mathbf{H_0^4}$ Ensemble CNNs are not more effective in contrast to their best base learner in visually recognizing solar flares at wavelength 1,600$\mathring{A}$

$\mathbf{H_1^4}$ Ensemble CNNs are more effective in contrast to their best base learner in visually recognizing solar flares at wavelength 1,600$\mathring{A}$

5 Contributions of the Dissertation

This study shows how the use of ensembles can improve the calibration of RNN-based models when predicting $\geq$C class flares. The study further shows how the reparameterization of the binary cross-entropy objective function can be used to improve ensemble calibration. This is done by using the Gumbel-Softmax on the final layer ($\geq$C or $<$C neural network nodes) of the ensemble's base learners. A softmax function is used in the ensemble model's meta-leaner. The overall strategy is a setup where a meta-model learns how best to combine and generalize from stochastic base learner predictions. The results show that the calibration of the ensemble is among those at the forefront in the state-of-the-art, under the scope of predicting $\geq$C class solar flares with RNNs. This study also validates the use of raw magnetograms in $\geq C$ solar flare recurrence prediction by showing that

denoising techniques (thresholding and various combinations of morphological erosion and dilation) and cropping a region of interest (ROI) reduce the predictive power of a ConvLSTM model. In short, noise and size reduction techniques are linked with a decline discriminating ability of ConvLSTM between sequences of magnetograms that lead to $\geq C$ or $< C$ solar flares. The study employs measurable Convolutional Neural Network (CNN) base learner diversity techniques that improve results obtained by [18], from (94%) accuracy to 100% on the same dataset. The study by [18] was the first to use CNNs on the Atmospheric Imagery Assembly (AIA) data at observations of wavelength 1,600$\overset{\circ}{A}$. Overall, this study as a whole shows how careful and justified design of base learner configurations can positively impact an ensemble model's performance.

6 Assumptions and Limitations

This study is limited to only predicting the recurrence of solar flares that emit radioactive energy of at least $10^{-6}\ W/m^2$ (commonly known as the $\geq C$ based on the GOES solar flare classification system). The choice of focusing on $\geq C$ class solar flares was motivated by the less precise predictions and poor calibration shown in existing works that used RNNs e.g., [7]. The prediction addresses the question of whether a given set of sequential solar flare features will produce a $\geq C$ or not (<C). This dissertation used existing datasets and did not seek to identify new sources of data. This study is also limited to only using RNNs and their ensembles in all ≥C class solar flare recurrence predictions. Only CNNs and their ensembles are used in all visual recognition methods. The only visual classes involved in visual recognition are two-ribbon, limb, compact, and the background class for solar images with no flaring activities. The visual recognition was done only using data from the SDO's Atmospheric Imaging Assembly (AIA) tool on observations at wavelength 1,600$\overset{\circ}{A}$ only. This study is more focused on how the setup of base learners relates to their ensemble's performance than it is focused on experimenting with various ensemble methods. Although this study improves the methods of visual recognition and solar flare recurrence, it did not seek to combine visual recognition and solar flare recurrence prediction to improve predictions.

7 Structure and Scope of the Dissertation

The dissertation is organized into chapters as follows:

1. **Chapter 1** defines the central theme & context of the research.
2. **Chapter 2** reviews previous literature on solar flare recurrence prediction and visual recognition of $\geq C$ class solar flares.

3. **Chapter 3** provides details of the RNN-based and CNN-based individual & ensemble methods implemented for solar flare recurrence prediction and visual recognition respectively.

4. **Chapter 4** is focused on the discussion of the results and efficiency and limitations of ideas behind the methodology.

5. **Chapter 5** provides conclusive insights on the collective findings of this research and highlights related areas that can be addressed in future research.

Chapter 2: Overview of Solar Flare Recurrence Prediction & Visual Recognition

1 Introduction

This chapter is focused on an overview of solar flare recurrence prediction and visual recognition. First, the historical trends in solar flare recurrence prediction and some background on solar flare visual recognition are provided. Thereafter, a review of previous works that are relevant to the scope of this research is undertaken as well as an overview of the main techniques that have been applied effectively in solar flare recurrence prediction and visual recognition. Next, a summary of some key concepts on the use of ensemble methods and the reparameterization of an objective function are discussed. This is followed by the description of most used data sources and the common nature of the data for solar flare recurrence prediction and visual recognition. The chapter ends by highlighting the key points discussed and provides a foundation for the next chapter on the methodology used in this study.

2 Background and History

The comprehension of the solar flare occurrence process has challenged solar physicists for at least 100 years [21]. Significant research endeavors have been made into solar flare prediction. Modeling the relationship between solar flaring activity and the photospheric magnetic field is the common approach [22]. Previous modeling approaches include decision tree approaches [23], Bayesian analysis [24, 25] radial basis function [26], support vector machines (SVM) and k-Nearest Neighbors (kNN) [27], regression modeling [28], etc. Most of the initial approaches considered only morphological features of a magnetogram at the current time [23, 29]. The results of the forecasts were inherently unsatisfactory. It was later demonstrated that considering past data together with the current features helps models do better [15]. In addition to sequential modeling of the data,

ensembles methods have also shown improvement in the forecasts; for example, the authors in [30–33] emphasized the effectiveness of ensemble methods in solar flare forecasting. A common conclusion is that besides the numerous choices for ensemble methods, simple voting ensembles tend to be more efficient. The evolution of effective approaches has lately been more inclined towards sequence modeling approaches. Let x_t denote a sample observed at time t. To train models, usually m successive data samples are used, $m \in Z$. The sequence can be defined as x_{t-m+1},x_{t-m+2},...,x_{t-1},x_t. Then x_t is used as the label of the sequence [7]. The label is commonly associated with the GOES classes for solar flares. Flare classes along with their X-ray peak flux range can be enumerated as follows: $X- \geq 10^{-4}$, $M- \geq 10^{-5}$ up to $< 10^{-4}$, $C- \geq 10^{-6}$ up to $< 10^{-5}$, $B- \geq 10^{-7}$ up to $< 10^{-6}$, and $A- < 10^{-7}$ [34]. Alternatively, researchers may choose to use the numerical X-ray peak flux of a solar flare, but this approach is not very common. Recurrent Neural Networks have become the norm in the field [1, 2, 5, 7, 35]. Overall, the evolution of data attributes and the use of RNN models has had a significant impact on the improvement of solar flare recurrence prediction. Besides all the efforts, predicting solar flares remains an active area of research due to their complex patterns [5]. The complexity is mainly driven by the stochastic nature of their occurrence [6, 36].

One of the major concerns about solar flare prediction is the reliability of the models used [7, 21] on $\geq$C class solar flares. In consideration of the randomness involved in solar flare recurrence patterns, the need to develop well-calibrated models is a necessity. A probabilistic classifier achieves calibration excellence when, within the subset of test instances assigned to a predicted probability vector p, the empirical class distribution closely aligns with the theoretically expected distribution encapsulated by p [37]. Figure 2.4 shows an example of a calibration diagram, where theoretical perfect calibration should be on the diagonal line (0,0) to (1,1). Due to the complex patterns in solar flare occurrence, models are prone to being under-confident or overconfident (as a result of over-fitting, especially in small datasets) [37]. It is therefore of paramount importance to explore strategies for producing well-calibrated models.

Most of the common metrics do not take into consideration the actual probabilities from the model's prediction. Most of them consider true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN), e.g., accuracy, true skill statistic (TSS), f1-score, etc. [7]. The issue with TP, TN, FN, and FP is that they do not indicate the categorical probability distribution in each test instance's prediction [38]. This makes it difficult to assess the model's certainty when making predictions. Calibration curves address that shortcoming.

3 Data Availability

The National Aeronautics and Space Administration (NASA) initiated a space weather-dedicated mission in 2010 [39]. The Solar Dynamics Observatory (SDO) (see Figure 2.1) was the main tool used. The SDO's instruments are namely Atmospheric Imaging Assembly (AIA) [40], the Extreme Ultraviolet Variability Experiment (EVE), and the Helioseismic and Magnetic Imager (HMI) [39]. The HMI and AIA made it possible to record magnetographs and ribbon imaging simultaneously [22], which is an advantage. This is an advantage because it allows visual recognition and recurrence prediction to be linked. This data is publicly available and is about 19 petabytes in size [41]. Most studies [5, 7, 34, 42] forecast solar flares using features they obtain from Space-weather Helioseismic Magnetic Imager (HMI) Active Region Patches commonly abbreviated as SHARP. The SHARP data contains features derived based on equations that describe the underlying physics behind the occurrence of solar flares. Some of the features describe the Lorentz force, the magnetic field, and other features. Some of the features quantify essential features of the magnetograms associated with the solar flares. The use and combination of various sources of data [11] have resulted in improvements in most of the studies. Figure 2.2 shows a magnetogram of a full solar disk. The black and white regions show magnetically active regions (AR) where sunspots are clustered [5]. The magnetograms are the most common source of data for solar flare research.

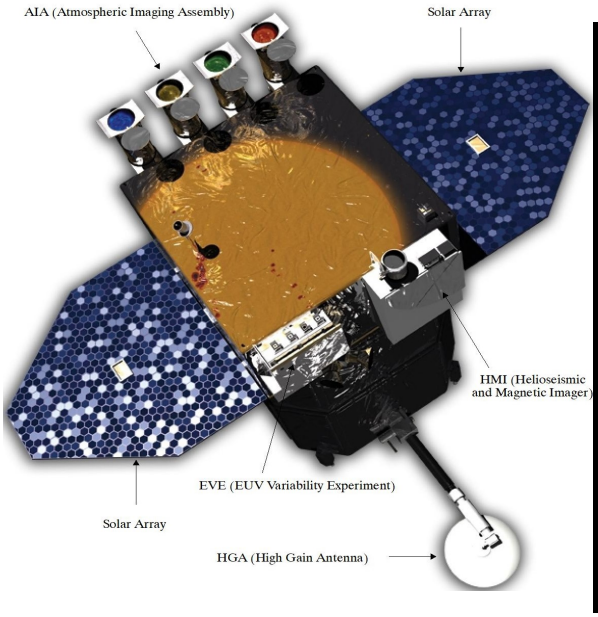

Fig. 2.1: The Solar Dynamics Observatory & Instruments, credits [12]

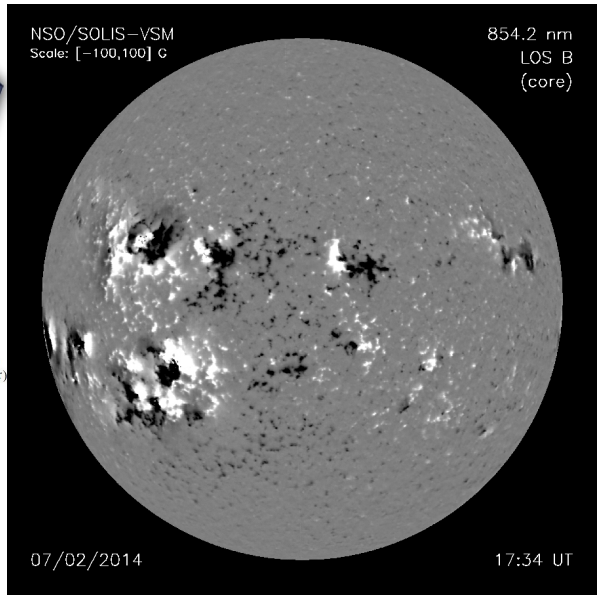

Fig. 2.2: A sample Magnetogram [43]

4 Literature Review

Solar flare research has a rich literature with a lot of unsolved problems. This review prioritizes studies that have used recurrent neural network models, a specialized multi-layer perceptron model, convolutional neural networks, and ensemble methods for solar flare prediction. A particular interest is given to the studies that focused on solving the problem of improved generalization and or calibration where $\geq$C solar flares were involved. The review first discusses issues related to recurrence then visual recognition follows.

The problem of calibration can be partially minimized by using random sampling methods on the data because if the solar flare environment is stochastic, then the data should reveal that. This however is not always a practically feasible solution. Most studies [7, 17, 21, 38] used and or commented on the chronological split of solar flare data. The data is commonly associated with active regions (ARs) where there are unique active region (AR) numbers for identifying ARs such as 9790, 9779,9794, etc. [44]. The active region numbers do not hold specific meaning except to serve as unique standard identifiers of the geospatial different solar regions. It is common for each AR to have multiple distinct instances in the dataset. The instances may differ in the time of capture and the X-ray peak flux at that time. The aforementioned existing studies claim that using random sampling makes the problem too easy because of overlap in training, validation, and test instances which may be from the same active region. To have strictly unseen instances in the test set, a chronological split with no overlapping AR instances is the common approach. For example, [21] and [38] use data from 2010-2014 for training and 2015 for testing. [7] uses data from 2010-2015 for training, 2014 for validation, and 2015-2018 for testing. Similarly, the data in [2] spans 2010-2018 and follows a chronological split. This study followed a chronological split but later used a reparameterization trick to simulate randomness which actually solves the calibration problem faced in [7] where LSTM calibration was undertaken when predicting $\geq$C class solar flares.

A study [7] used a Long-Short Term Memory (LSTM) and Random Forest (RF) models for predicting the recurrence of solar flares. The classes used were $\geq M.5$, $\geq M$, and $\geq C$ (as per the GOES solar flare classification system). Feature ranking reduced an initial set of 40 parameters to 22 and 14, respectively, with improvement in the predictions. This showed how significantly noisy data can affect a model's performance. As similarly observed in [17], C class flares were associated with a higher dependency on historical information. The results obtained were better than those from statistical methods [45, 46], SVM [17, 28, 46, 47] and Deep Flare Net [21]. The superiority is mainly credited to the dimensionality reduction in data, the architecture of the LSTM, and the attention mechanism

used along with the LSTM. This means that room for improvement is in both the data and architecture of models. The study yielded a True Skill Statistic score of 0.612±0.009 using the LSTM and 0.552 ±0.003 using the RF model. Besides the slightly better performance of the LSTM, the RF model appeared to be better calibrated. The reliability/calibration diagram (RD) for the LSTM can be seen in Figure 2.4. As per the definition of the calibration, the LSTM is not well calibrated as its curve is on average significantly displaced from the diagonal (0,0) to (1,1). This means that the model was not well aware of the uncertainties associated with its predictions, in this case, it shows to be under-confident. The RF is an ensemble model and it demonstrated that there is a possibility of an ensemble to be better calibrated than an individual model such as the LSTM in [7]. Hence in this study, the proposed framework uses an ensemble model. Since the RF is mainly a homogeneous ensemble, it may not have enough diversity in its base learners for better ensemble success. It is a requirement according to [48]. Motivated by this fact, this study used a heterogeneous stacking ensemble of Recurrent Neural Networks (RNNs) in the proposed solution. The adjustments in the two design choices indeed showed that the calibration of the model significantly improved, using the same dataset as that used in [7].

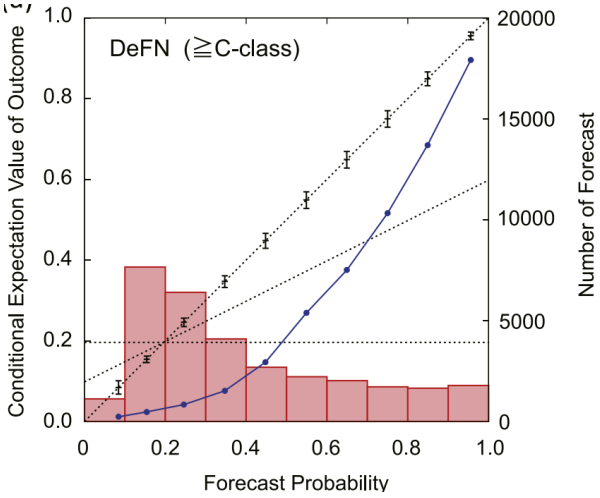

Fig. 2.3: DeFN (≥C) Calibration [38]

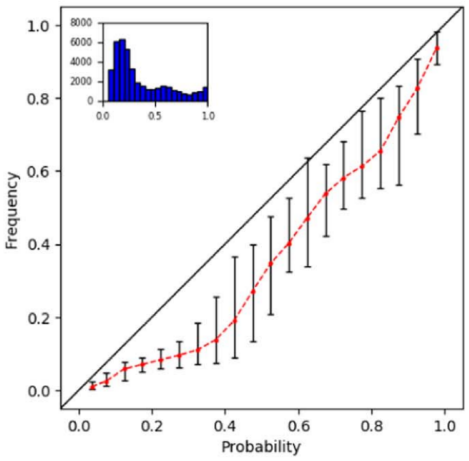

Fig. 2.4: LSTM (≥C) Calibration [7]

In 2018, [21] introduced a model named Deep Flare Net (DeFN) dedicated to solar flare prediction. The DeFN is a multi-layer neural network with no RNN layers. For ≥C solar flares, it yielded a similar TSS as [7], which is 0.63. It differed in that it used data from 2010 to 2015, while [7] used data from 2010-2018. The common issue for [7] and [21] was the calibration and they both used chronological splits in the data. The calibration of DeFN [21] is shown in Figure 2.3 [38], where the authors were improving DeFN to produce DeFN-R which is a more reliable version of DeFN. DeFN-R was optimized for probabilistic forecasting by maximizing the Brier Skill Score (BSS) metric amongst other modifications and it showed significant near-perfect calibration after improvement. In the critics, [38] outlines that DeFN (the poor calibrated model) was more focused on trying to make

deterministic predictions. From this analysis, it can be argued that if a model is trained to be closer to behaving like a deterministic model, then it is likely to be unreliable (poorly calibrated). This is based on the domain knowledge that the solar flare environment is a stochastic environment [4]. Instead of taking the route by [38], it is worthwhile exploring how a model deliberately trained to make non-deterministic predictions would behave. This can be achieved by adding noise in the binary categorical predictions. In solar flare prediction, there is no known study that does this. An adaptable method is the work of [49], which is the addition of Gumbel distributed noise in categorical predictions and then using a Gumbel-Softmax for the final probabilities. Therefore, this study used the Gumbel distributed noise to get better calibration.

RNNs have also been used in [5] to forecast solar flares within the next 24 hours. The study used the Helioseismic and Magnetic Imager (HMI) SHARP, with 20 parameters of interest. The LSTM, GRU, and Simple RNN models were used for binary and multi-class classification of solar flares. Binary class predictions outperformed the multi-class predictions. The advantage of binary class prediction is the specialized model's ability to optimize its weights better. The minimum performance of the RNN models was 60% on the accuracy metric. The GRU was better than the Simple RNN and the LSTM model. Part of the reason could be that the Simple RNN and the LSTM were too simple (fewer gates than the GRU) or too complicated (more gates than the GRU), respectively in their architecture and hence the deviations in performance. This hypothesized cause of difference, draws attention to the investigation of how the model architectures influence the differences in performances, when working with various data sets where there is a variety in the size of data and noise. Data were down-sampled to achieve the smallest balanced dataset. While that seems to be a good and fair way to train the models, the results show that models struggled to classify the negative classes well. Using an imbalanced dataset with the right metrics would possibly alleviate this issue, as was done in [7].

The first study for visual recognition of solar flare ribbons (captured at wavelength 1,600$\mathring{A}$) using CNNs was conducted in [18]. The study focused on classifying solar flare ribbons into one of four classes namely limb flares, compact ribbon flares, two-ribbon flares, and a control class named quiet sun for a solar disk region with no solar flare going on. The data used was obtained from the SDO's Atmospheric Imaging Assembly (AIA), limited to only $\geq$C class solar flares. The results showed a maximum of 94% classification accuracy. The subjective visual complexity of the observations was relatively low for observations at wavelength 1,600$\mathring{A}$. It can be argued that the architecture of the CNN model used by the researchers was eitheri too small, too large, or poorly optimized in training. Another study [50] demonstrated how some CNN architectures can perform in a more complex dataset where the task was to visually recognize various solar events. In consideration of those facts, this dissertation

shows that the use of sufficient architectures can allow for improved performances. Also, it shows that more complex model architectures can lower the performance. The study in [50] recommended the use of ensembles for better success of CNNs. Simply combining models to form an ensemble is not a reliable method for getting the best out of an ensemble architecture. An intensive review [48] discussed the various factors to be considered when setting up the ensemble. One of the key things is ensuring diversity in the base learners of the ensemble. This dissertation sets up diverse base learners whose diversity is mathematically quantified and produces an ensemble that improves the classification results of the wavelength 1,600$\overset{\circ}{A}$ observations to 100%. While individual model architecture selection can improve performance, where one model is not efficient an ensemble model can be a more effective solution.

5 Related Techniques and Applications

5.1 Recurrent Neural Networks (RNNs)

Recurrent Neural Networks (RNNs) have become more common in predicting solar flares [1, 2, 5, 7, 14]. The most common version of the RNNs is the Long Short-Term Memory (LSTM). The LSTM in most cases is preferred for its ability to handle long-term dependencies in contrast to the standard RNNs. The $\geq$C class flares have been identified to be more predictable based on historical data [7]. Although RNNs are a promising method in general, they have the shortcoming of poor calibration when forecasting $\geq C$ class solar flares. Addressing this matter may involve mimicking some of the characteristics of the alternative models that show better calibration, which are ensembles (not RNN based) as in [7]. Some RNNs can use sequential images as input, they are called Convolutional RNNs. Magnetograms can be treated as images which makes Convolutional RNNs applicable in magnetogram sequence-related predictions.

5.2 Convolutional Neural Networks (CNNs)

CNNs have demonstrated success in medical imaging. Medical images are of a similar nature to those of solar events [51], as demonstrated in [52–54]. As described in [50], a CNN's convolutional layers are specialized to image grid-like inputs. A kernel (matrix) is utilized by the convolutional layers to identify local features from an image [55]. A pooling layer follows and reduces the spatial size of the image. As the CNN depth progresses, a fully connected layer detects high-level features partially depending on previously extracted simpler features [55]. The activation functions in neurons help in deciding if a neuron will be useful for some computations. Back-propagation helps with weight adjustment which is aimed at loss minimizing as the CNN model learns. The ReLU activation

function is faster to compute than the $tanh$ activation function [56, 57] hence it is preferred. Drop-out is a technique that helps decrease the number of neurons to use in the computations and minimize the issue of over-fitting [48]. In cases of unsatisfactory results from individual CNN models, [50] recommended ensemble CNNs.

5.3 Ensemble Methods

Individual models are not always efficient when classifying $\geq C$ solar flare ribbons [18]. Ensembles can be applied in such cases [48]. Ensemble learning is all about combining models with the objective of having a superior combination [48]. [58] conducted a rigorous experiment of multiple datasets and models to demonstrate ensemble superiority over individual models. The reasons behind ensemble success are explainable [59]. The hypothesis search space for an ensemble is usually larger than that of the individual models. The best hypothesis to be learned by the base learners may not be in a search space of any individual model but in that of the combination of the base learners. An ensemble is more likely to find a global optima in contrast to its base learners. To define an ensemble, assume that there is a dataset with n instances and p features. The associated mathematical definition of the data is given in Equation 2.1 [48].

$$D = \{(x_i, y_i)\}(|D| = n, x_i \in R^p, y_i \in R) \tag{2.1}$$

the ensemble model ρ uses a function E that aggregates the K base learners $\{f_1, f_2, ..., f_k\}$ to make a single prediction $\hat{y}$ (the ensemble prediction). D is the dataset such that $|D| = n$. x_i and y_i are the instance and label respectively.

$$\hat{y} = \rho(x_i) = E(f_1, f_2, ..., f_k) \tag{2.2}$$

where $\hat{y} \in R$ or $\hat{y} \in Z$ for regression or classification cases respectively. For good performances in an ensemble, base learners must be diverse and have at least a 0.5 probability of making the right prediction. Whilst there are many types of ensembles, a simple majority voting ensemble is powerful [34]. According to [48], if $V = \{v_1, v_2, ..., v_n\}$ is a set of n voters, and the probability $p(v_i)$ that each voter chooses the right option in given options is above 0.5, then combining these voters yields better accuracy. It is therefore critical to ensure that base learners each yield at least 50% success on the data. In this dissertation, the precondition is met by all base learners used. The diversity of the base learners is highly regarded as the most decisive factor of an ensemble's success [48, 60]. There is no clear definition of diversity [60]. The strengths of base learners on distinct parts of the data help the ensemble perform well [61]. Ensembles use the philosophy of "crowds are wiser than

one", not all crowds are wiser though [48]. The ensemble will more likely be successful if base learners' decisions are independent to avoid biases and it must be possible to apply some base learner decision consolidation techniques [48]. The combination of each base learner's search spaces helps the ensemble have an even bigger solution search space. In this dissertation, base learner diversity is quantified using the Q-statistic [60] method which is explained in detail later in the methodology section.

5.4 Reparameterization Methods

Stochastic neural networks are normally used to learn stochastic processes [49]. Based on the common use of hyper-parameters that regulate over-fitting, e.g., dropout and regularization, RNNs are mostly used as stochastic models. The solar flare occurrence process is a stochastic one [6]. The issue of major concern is the calibration of the RNN models involved in solar flare recurrence prediction. Including randomness in the predictions of RNNs can possibly help the RNNs learn the stochastic nature of solar flare occurrence. An existing method for doing this is the application of Gumbel distributed noise in the categorical predictions before applying the softmax function (which is then referred to as Gumbel-Softmax) as proposed in [49] on a general paper that had nothing to do with RNNs or solar flares. The concern is how much randomness to inject into the predictions. This is addressed by controlling a parameter called tau (τ) that determines the degree of Gumbel-distributed noise in the categorical predictions. So in this dissertation's methodology, the application of the Gumbel-distributed noise is used in RNN base learners of an ensemble to leverage the advantage of randomness in predictions to finally produce a well-calibrated model. It should be noted that there is no known study at the time of this dissertation that applies this technique.

6 Conclusion

RNNs have demonstrated significant improvement in the recurrence prediction of solar flares in comparison to conventional methods and other types of machine learning models. The need for improvement in calibration when predicting $\geq C$ class flares is emphasized by the state-of-the-art results, which show poor calibration. Ensemble models (not RNN-based) have demonstrated potential for improvement in the calibration of RNNs based on how they are calibrated under the same training and test sets [7]. In general, ensemble success is highly dependent on the diversity of the base learners [48]. To induce diversity in base learners while enabling better learning of the stochastic nature of solar flare occurrence, reparameterization methods can be combined with ensembles to create more reliable models that comprehend the uncertainty associated with their

predictions.

Solar flare visual recognition for images captured at wavelength 1,600$\mathring{A}$ using CNNs has not been explored in depth as recently pioneered in [18]. While the data was not very complex in terms of subjective separability of the classes (provided the individual is well informed), the results presented in [18] obtained a maximum of 94% accuracy. The areas of improvement that need exploration are individual model architecture and ensembles with diverse base learners. The next chapter details how ensemble methods are set up for solar flare prediction and how reparameterization techniques can be applied to improve the ensemble's performance in terms of reliability (calibration). The next chapter also includes a detailed description of the setup of CNNs and ensemble CNNs with diverse base learners.

Chapter 3: Methodology

1 Introduction

This chapter provides details on the methods used in the experiments and justifies experimental design choices and the identified areas of improvement. Firstly, the experimental setup of RNN-based ensembles that predict the recurrence of ≥C solar flares is presented. The first objective of the experiments is to investigate the performance of individual models (LSTM, GRU, and SimpleRNN) in contrast to their ensembles (heterogeneous stacking, homogeneous stacking, soft-voting, and hard-voting). Furthermore, the setup of the experimental design for exploring how reparametrization of the objective function can be used in the individual RNN and RNN-based ensemble models as a means for improving calibration is described. In addition to that, a magnetogram preprocessing pipeline is used to investigate the effectiveness of various preprocessing methods. A Convolutional LSTM model is used to evaluate the various approaches. Lastly, details on the experimental design of CNNs and CNN-based ensembles used for visual recognition of ≥C solar flare ribbons captured at wavelength 1,600$\mathring{A}$ are provided. This chapter is also structured based on the four experiments that were separately conducted in this study. Each experiment is assigned a subtitle in this chapter. The first two experiments use the same dataset, whereas the last two experiments each use different datasets.

2 Ensemble RNNs For ≥ C Solar Flare Recurrence Prediction

2.1 Dataset

The data used was adapted from a solar flare forecasting study [7] - the benchmark study. Although the data originally had instances with 40 data points, this study used the features that the benchmark study in [7] concluded were the top ranking features. This was done for comparable performances, mostly in calibration. The features combine Space-weather HMI Active Region Patches (SHARP) data points and historical flaring data points. The entire dataset spans the years 2010 May -2018 May, inclusive.

The GOES solar flare classification scheme was used for converting X-ray peak fluxes(real number values) to classes (categorical). The X-ray peak fluxes were provided by the National Centers for Environmental Information (NCEI). After partitioning, the training set had 66311 and 18266 negative and positive samples (spanning 2010-2013), respectively. For validation, 19418 and 7055 negative and positive instances (spanning 2014) were used, respectively. Lastly, 35957 and 8732 negative and positive samples were partitioned for testing, respectively (spanning 2015-2018). The data was split based on years of observation, in the original dataset [7]. In this study, the same partitions were used. Using data from some range of years for training, validation, and testing can improve the performance of the model in contrast to randomly stratified samples with a similar class proportionality. This is due to the solar cycle dependence nature of the solar flare data [2]. To preserve the seasonality in the data and ensure equally proportional class distribution, some samples in each partition were dropped. Finally, in each partition, the ratio of positive to negative samples was 1: 2.75, e.g. 18266 $\geq C$ positive samples and 50274 negative samples for (2010-2013) . This study forecasts the occurrence of a solar flare within the next 24 hours. The main models used are the GRU, Simple RNN, and LSTM. All models were implemented with an attention mechanism as it helps with learning the sequential data [62]. There is no difference in the attention mechanism used in [2] and the one used in this study. Since RNNs use data that is in the form of sequences, the sequence length used was 10, which is the same as in [7] The time gap between instances was 60 minutes. The final feature vector size per instance was 14 (same as in [7]). The similarities in some aspects were preserved for a fair comparison with the work in [7].

2.2 Ensemble Models

Ensemble models were implemented by combining the LSTM, GRU, and Simple RNN. The choice of 3 models helps the majority voting ensemble to always have a majority preference in class choice. The ensembles used are a hard voting ensemble (uses majority voting), a soft voting ensemble, and stacking ensembles. For the stacking ensembles, there is a heterogeneous stacking ensemble of the three models, then homogeneous stacking ensembles, one for each model, each with three base learners trained on 25% of the training set. Each base learner was trained on a distinct stratified sample from the training set to improve base learner diversity. The base learners were all trained on data that spans (2010-2013). In this study, the loss functions, optimizers, data, and model structure were the main deliberate means of making the models diverse. Figure 3.1 shows an example of the setup for the ensemble models. The meta-learner uses the outputs of the base learner outputs as inputs to learn the best combination. Due to limited positive instances in the data, models share validation data. For comparable results, the models are all tested on the same testing set for

reasonable comparison. Figure 3.1 shows how the three base learners (LSTM, GRU, and Simple RNN) are set up and related to the meta-model. Each of those base learners is a complete model that can make predictions. The meta learner has a dense layer of 500 then 50 neurons that sequentially precede the final prediction layer which has two neurons one for ≥C's probability and the other for <C's probability.

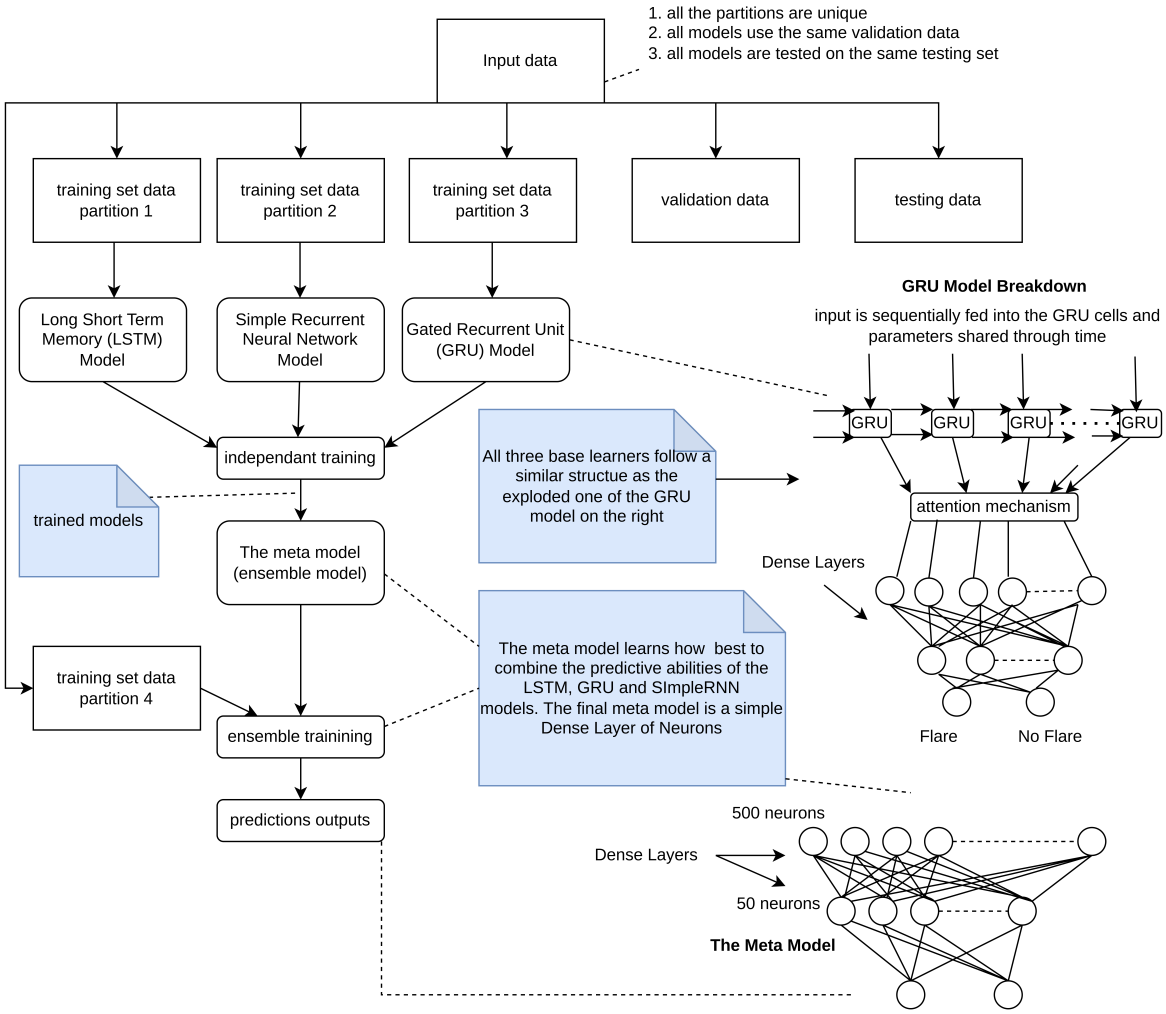

Fig. 3.1: A Setup of the Heterogeneous Stacking Ensemble (HtrSE)

2.3 The Simple Recurrent Neural Network (Simple RNN)

[63] mathematically define the standard recurrent cell as:

$$\begin{aligned} h_t &= \sigma(W_h h_{t-1} + W_x x_t + b), \\ y_t &= h_t, \end{aligned}$$

where x_t, h_t, and y_t at time t of the cell, are the input, recurrent information, and output, respectively. W_x, W_h, b are the input weight, recurrent information weight, and bias, respectively. This architecture

has demonstrated success in some studies [64].

2.4 The Long Short-Term Memory (LSTM)

The LSTM extends the Simple RNN architecture to improve long-term dependency learning [63]. The authors in [65] came up with the idea of the LSTM in Figure 3.2. Figure 3.2 shows how the LSTM is applied in the context of this study. Since the data used is sequential, the LSTM is able to learn the long-term and short-term solar flare historical information and shares this information through the cell gates namely C_t and h_t (see upper and lower right of Figure 3.2 respectively). Still in Figure 3.2, X_t (see bottom left) shows where each instance of a solar flare is taken as input. The instances/elements of a solar flare sequence of events are each processed as time series data, one instance at a time in order of appearance in the sequence.

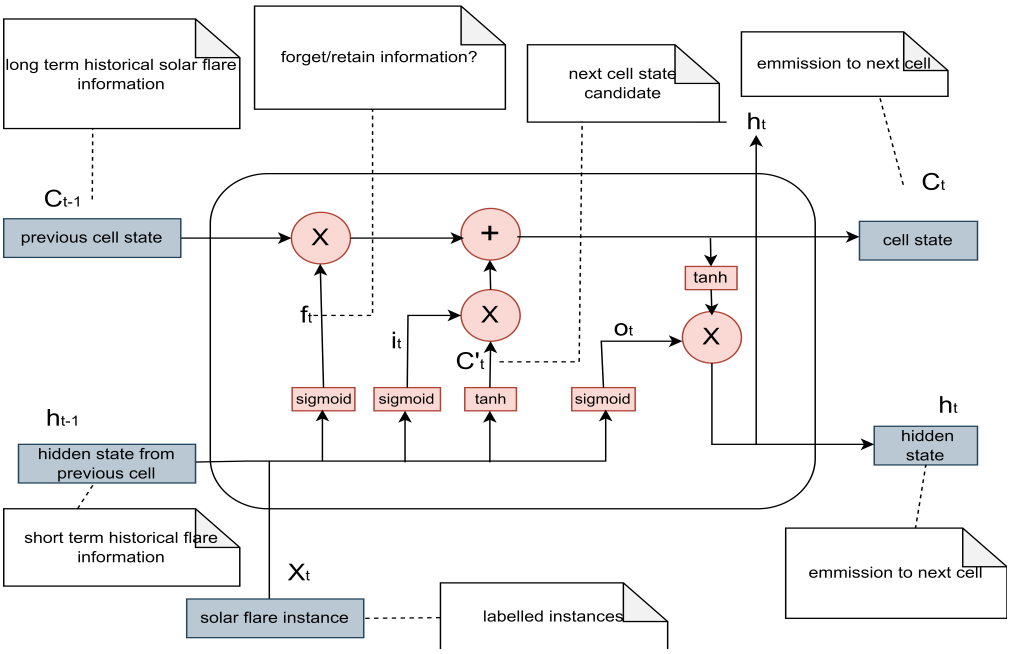

Fig. 3.2: LSTM Diagrammatic Overview As Used In this Experiment

The LSTM has been modified numerous times since its initial proposal. The most commonly used

variation is the LSTM with a forget gate [66], and can be mathematically expressed as:

$$\begin{aligned} f_t &= \sigma(W_{fh}h_{t-1} + W_{fx}x_t + b_f), \\ i_t &= \sigma(W_{ih}h_{t-1} + W_{ix}x_t + b_i), \\ \tilde{c_t} &= tanh(W_{\tilde{c}h}h_{t-1} + W_{\tilde{c}x}x_t + b_{\tilde{c}}), \\ c_t &= f_t \cdot c_{t-1} + i_t \cdot \tilde{c}_t, \\ o_t &= \sigma(W_{oh}h_{t-1} + W_{ox}x_t + b_o), \\ h_t &= o_t \cdot tanh(c_t), \end{aligned}$$

where the Ws are weights and the bs are biases. Both the weights and biases are learned during the training. x_t denotes the input. h_{t-1} is the resulting vector from time step $t-1$. $\tilde{c_t}$ is the candidate cell state. h_t is the resulting vector at time step t, which is dependent on the new cell state c_t. i_t regulates the amount of new information going into the cell. The key feature of this model is the ability to learn what to forget [63], through f_t, the forget gate. $f_t \in (0, 1)$ and $f_t \in \mathbf{R}$ with 0 for forget and 1 for retain. LSTMs perform better than the standard RNN in practice [63]. The price for better performance is computational load due to extra operations. As a means to balance the trade-off between computational load and performance, [67] introduced the Gated Recurrent Unit (GRU).

2.5 The Gated Recurrent Unit (GRU)

The GRU combines the forget f_t and input i_t gates as an update gate making it have one less gate compared to the LSTM. Sacrificing that one gate from the LSTM to form the GRU comes with limitations of the GRU in contrast to the LSTM [68, 69]. Both the LSTM and GRU are better than the standard recurrent cell [67]. The set of mathematical equations defining the GRU can be outlined as follows:

$$\begin{aligned} r_t &= \sigma(W_{rh}h_{t-1} + W_{rx}x_t + b_r), \\ z_t &= \sigma(W_{zh}h_{t-1} + W_{zx}x_t + b_z), \\ \tilde{h_t} &= tanh(W_{\tilde{h}h}(r_t \cdot h_{t-1}) + W_{\tilde{h}x} + b_z), \\ h_t &= (1 - z_t) \cdot h_{t-1} + z_t \cdot \tilde{h_t}. \end{aligned}$$

In the GRU, the reset gate (r_t) and update gate (z_t) facilitate the forgetting/retaining and updating of information respectively. For example, z_t regulates the degree with which a new solar flare instance

Table 3.1: Configurations for LSTM, GRU, and Simple RNN Models

Model	Loss Function	Optimizer	DLA
LSTM	KLD	SGD	Relu
GRU	BCE	Adam	LeakyReLU
Simple RNN	BCE	Adamax	Relu

updates the information needed to make the final ≥C or <C possible events. Table 3.1 shows the details for each model's configurations. The variation of the configurations are means for inducing or increasing diversity in the performance of the RNNs. Diverse base learners are more likely to form a superior ensemble [48]. Some of the models make use of the following methods/techniques: Dense Layers Activation (DLA) function, Binary Cross Entropy (BCE), Kullback Leibler Divergence (KLD) and Stochastic Gradient Descent (SGD). The activation function used in output layers is the softmax for all models. The dropout rate is 0.5. This helps with combating over-fitting. The LSTM, GRU, and Simple RNN models have common configurations. Each sequence of length 10 where each element contains 14 features goes into a layer with 10 RNN (LSTM, GRU, or SimpleRNN) cells. The hidden states are then passed to the attention mechanism [7]. The outputs of the attention mechanism are then passed to a dense layer of a neural network. All the models have 3 dense layers. In all models, the second dense layer has 500 neurons and 2 for the output layer. The input (first) dense layers of the Simple RNN and LSTM have 200 neurons whereas the GRU first dense layer has 300 neurons. The difference was influenced by an experiment-based observation which was that 300 neurons lead to a slightly better performance for the GRU.

2.6 Performance Metrics

The key metrics used in this study are TSS, BACC, f1-score, precision, and recall. The receiver operating characteristic curve, confusion matrices, and reliability diagrams are also used as means to increase the interoperability of the RNN models. The Q-statistic helps with quantifying the diversity of two RNN models. Since diversity increases the likelihood of ensemble success [48], it is critical to measure the base learner's Q-statistic to give a stronger intuition of the likelihood of the ensemble's success with the given base learners. Due to class imbalance, BACC is used since it gives a better impression of the model's performance when taking into consideration the class distribution. The accuracy metric is not reliable in this context of imbalanced data e.g., if a model gets 90% accuracy by correctly predicting 100% of negative solar flare samples which make 90% of the solar flare test data, where the remaining 10% are positive samples. Another metric namely the Hiedke Skill Score

(HSS2) was included for comparison with other works. The mathematical definitions of some of the metrics are provided in [7] [16].

The TSS is used because the data is imbalanced. The class imbalance does not affect the TSS by the majority or minority class weight in the data [17]. The possible values of the TSS range from -1 to 1. A score of 1 means perfect prediction, whereas -1 means all predictions are wrong. Random guessing scores a 0. Basically, the TSS gives an idea of how better than random the model predicts. The ROC curve is also used in this study along with precision-recall curves. The ROC is normally presented as a line graph where the Area Under the Curve (AUC) shows the model's degree of separability amongst classes and the best possible value is 1 [70].

3 Reparameterization for ≥C Solar Flare Prediction Ensemble Calibration

3.1 Experimental Overview

This experiment implements a framework that is based on the use of heterogeneous ensemble RNN models. The base learners of the heterogeneous ensemble are configured with a Gumbel-Softmax (GS) function [49]. The main functionality of the GS function is to inject randomness into the learning process which leads to a well-calibrated meta-model. The Gumbel-Softmax has a parameter named tau(τ) which regulates the degree of relaxation of the model's predicted probabilities. Another feature of the framework is the use of a Simulated Annealing (SA) algorithm to optimize the effective range of values of τ that allows the overall ensemble to have an improved calibration in comparison to an alternative where the base learners do not use the GS. It is critical to note that this study does not propose a globally efficient optimal τ value nor suggests the SA algorithm as the best for the task. This study proposes the approach for subjectively optimizing the τ parameter for improved ensemble calibration and uses a practical example which the remainder of the study is mainly focused on. The optimization of τ is said to be subjective since the chosen τ value is not a single (as returned by the Simulated Annealing algorithm) value but a range where the performance of the RNN model is stable and more consistent in multiple simulations.

The GS is used in this study instead of the normal softmax. The GS allows the model to explore a wider search space by injecting randomness in its final prediction and hence a wider search space is used. The GS is defined as follows, given a set of logits $\mathbf{z} = (z_1, z_2, \ldots, z_k)$, where k is the number of classes and a temperature $\tau > 0$, we compute the Gumbel-Softmax probabilities $\mathbf{p} = (p_1, p_2, \ldots, p_k)$

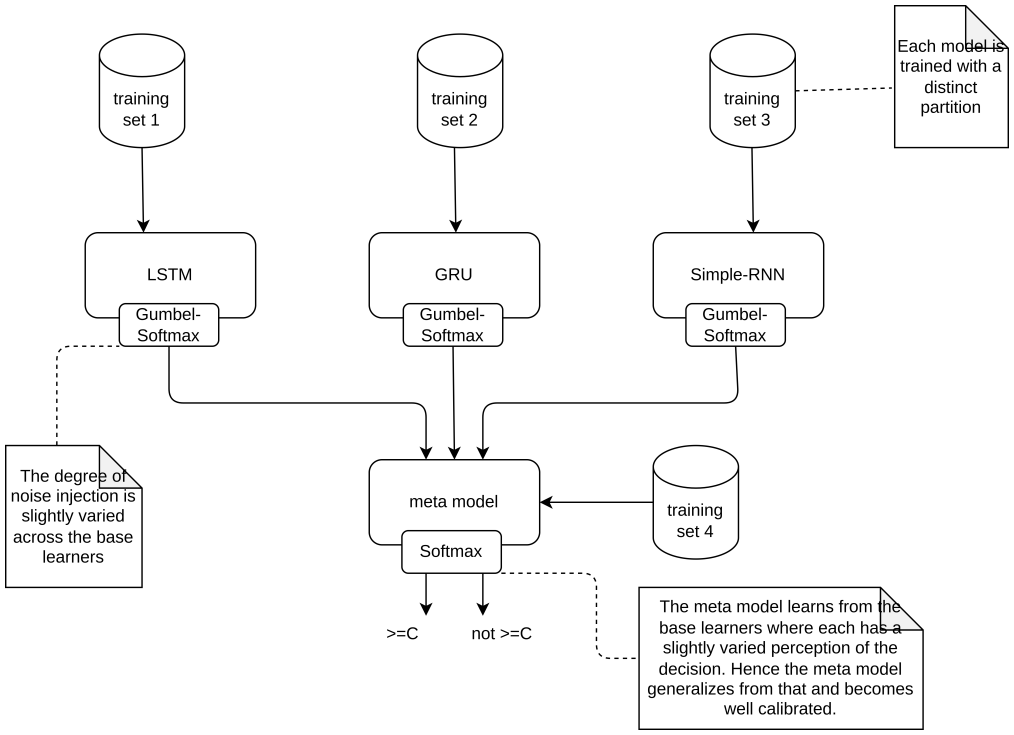

Fig. 3.3: The Stacking Ensemble Proposed Architecture

as in Equation 3.1 [49].

$$p_i = \frac{e^{((\log(z_i)+g_i)/\tau)}}{\sum_{j=1}^{k} e^{((\log(z_j)+g_j)/\tau)}} \tag{3.1}$$

where g_i are samples from the Gumbel distribution with location 0 and scale 1. $g_i = -log(-log(U_i))$, where U_i is a random variable constrained between 0 and 1.

The parameter τ from the GS definition in Equation 3.1 affects the exploration and exploitation of the search space in three ways. As $\tau \to 0$, $GS(p, \tau) \to p$. If $\tau \in (0, 1)$ then the model has a balance between exploration and exploitation. If $\tau > 1$ the model explores the search space more. The setup of the ensemble is provided in Figure 3.3. The LSTM, GRU, and Simple RNN in Figure 3.3 are complete models that are capable of making predictions of ≥C solar flares. Each base learner is configured with the Gumbel-Softmax which introduces randomness in the learning process of each model. The meta-model (see mid-bottom in Figure 3.3) does not use a Gumbel-Softmax as it aims to learn the randomness and not to deliberately introduce more. The meta-model makes the final predictions with better certainty in contrast to the base learners.

Simulated Annealing (SA) is one of the most used optimization techniques for finding optimal solutions [71]. In relevance to this study, the GS is more effective with optimal τ configuration. There are known ranges of τ that have known implications on the categorical distribution [49]. Based on the ranges of τ and their implications, an iterative approach must be investigated to find the best τ.

This makes the SA a candidate method for optimizing τ. The SA requires a function for computing a ranking metric for solutions, in this case, the prediction ability of the model with a certain configuration of τ. To harness the full power of the SA, more iterations and solution neighbor generation should be made. The downside of this approach is the computationally expensive operations. What then becomes significant is the consideration of the trade-off that is made between finding the absolutely optimal τ or an approximation that can work. This study used large iteration steps in the iterations in the SA and less neighbor solution generation for the mentioned reasons.
A fourth partition was used to test the LSTM while observing its behavior as τ increases from ≈ 0.05 to ≈ 3. The use of this fourth partition rules out any counterargument on the use of some of the test data for assessing a hyperparameter's optimal range. Taking this fourth partition from the test set would result in an unfair comparison of the resulting model in contrast to related studies. Hence, the fourth partition was taken from the validation set.

3.2 Framework for Solar Flare Forecasting

The aforementioned facts were used to design a framework for robust solar flare forecasting RNN in which the Simulated Annealing algorithm(SA) is used to optimize τ. The main objective of the SA, in this case, is to find τ that maximizes the True Skill Statistic of the RNN model on test data. The variables Sc, Sb, and So represent the current, best, and initial states in pseudo-code of the algorithm in Figure 3.4. The variables TSS_{Ec} TSS_{En}, τ_{best}, represent the current state, neighbor state, and best τ values respectively. So in this version of the SA, the energy of the solution is the TSS obtained when testing the model trained with the Gumbel-Softmax using the current τ.

The LSTM model used in the search for τ was configured as follows. The loss function used was the Gumbel-Softmax loss function. The optimizer was the Stochastic gradient descent algorithm. The epochs were set to 20. The performance metrics used in the experiment are the same as those described in Section 2.6. In the original paper [49] on the Gumbel-Softmax, one of the suggestions was that τ be internally (in the model) annealed during training. This study uses a fixed but carefully selected value of τ. A τ value fixed at a good value allows for faster hyper-parameter optimization. The alternative approach keeps changing the degree of relaxation (τ) of the noise injected in predictions during model training. While that may effectively search for the best possible τ value in some contexts, it requires a sufficiently large dataset to optimize τ. During training, it can be interesting to know how the model behaves under exploration vs exploitation. A fixed value of τ (if in a good range) makes it easier to observe that, faster. The disadvantage of the fixed value is that the researcher must have a strong intuition of what value is good. Luckily, the work by [49] specifies the ranges of τ and

```
state SA(T_min, T_max, delta_T, model=None):
    Sc,So,Sb=[None, 0] #[model,TSS]
    TSS_Ec, TSS_En,T_best=delta_E = 0

    for (T=T_min, T< T_max, T+=delta_T):_{TSS,BACC}

          model_c = lstm(nclass, n_features, series_len)
          TSS_Ec = train_evaluate_TSS(model_c, T)
          model_n = lstm(nclass, n_features, series_len)
          T_n = T + (0.5 * delta_T)
          TSS_En = train_evaluate_TSS(model_n, T_n )
          delta_E = TSS_Ec - TSS_En

          if(delta_E[0] > 0):
             Sc=Sb
          if(Sb[1] < Sc[1]):
             Sb=Sc
             #T_best=T

    return Sb, T_best
```

Fig. 3.4: SA Pseudo-code For Exploring τ vs TSS & BACC

their implications on the model's performance in general. This makes it easy to have a good guess. An internally annealed τ will keep changing during training and delay hyper-parameter optimization. This may cause the τ value to change before some of the hyper-parameters have adjusted best relative to it, making interpretations hard, i.e., it may not be clear if τ is the source of the model's failure or the parameters and hyper-parameters are not best optimized yet. Obviously, this will depend on the conditions under which the τ is configured to change during the training. The issues are preventable if the available data is large enough.

4 Line of Sight Magnetogram Preprocessing

This section covers an investigation of magnetogram preprocessing approaches and how they affect solar flare recurrence predictions. The parallel computing Message Passing Interface (MPI) API is used to reduce the pipeline's execution time based on available CPU cores. A Convolutional LSTM model was used to examine the efficiency of the pre-processing methods.

4.1 Dataset

The dataset used was adopted from [72] from the Institute for Data Science, Fachhochschule Nordwestschweiz (FHNW), Switzerland. The dataset providers credit the SDO satellite mission &

Joint Science Operations Center(JSOC) Stanford as original sources of the initial version of the data. Although the data contained multiple types of solar imagery, this study only used the magnetograms and their metadata (start time, end time, and X-ray peak flux(W/m^2)). The data originally contained 39 samples of X-class solar flares from 17 unique active regions. The M class had 500 samples from 111 active regions. The C class had 2936 samples from 407 unique active regions. The B class had 1210 samples from 106 unique active regions. Finally, the Quiet class (no ongoing solar flare) had 3651 samples from 450 unique active regions.

4.2 Preprocessing

Preprocessing the magnetograms was done to improve their representation of the visual features that are theorized to be important, e.g., the umbra and penumbra which form an active region where solar flares erupt [73]. Figure 3.5 shows the magnetogram pipeline that was used. The magnetograms are first obtained from the SDO dataset [72]. Each magnetogram initially has 250 $\times$ 250 pixels. First, the largest umbra or penumbra in the magnetogram is located and used as the central point for cropping out a region of interest (ROI) for all magnetograms containing active regions (AR). This is done to reduce the spatial pixel area in an attempt to improve computational efficiency under the assumption that the cropped region contains more useful data. The ROIs have dimensions of 150 $\times$ 150 pixels. A similar full version of the magnetogram is kept along with the ROI and they both undergo the following preprocessing steps. This is done to test the impact of using full magnetograms in contrast to the ROIs. Let p denote the pixel value. The pair of magnetograms are then thresholded such that the darkish ($0 \leq p < 120$) and whitish ($180 \leq p \leq 255$) regions become fully black and fully white respectively. The remainder of the pixels are set to be gray, at the pixel value of ($120 \leq p < 180$). The thresholds were set based on subjective analysis of the related histograms from multiple classes, e.g., see Figure 3.9. The resulting image has noisy pixels, from a visual perspective. What follows then is a closing and then opening operation. Figure 3.11 shows how the magnetogram is affected. The input image in Figure 3.11 has already been thresholded. The closing operation helps remove minor holes in the magnetogram. The opening operation was done on the closed image to further remove noise and preserve the large objects. The closing procedure is done through dilation and then erosion [74]. The opening procedure is done through erosion and then dilation. A structuring element $S_e \in Z^{7\times7}$ was used. $\forall b_{i,j} \in S_e \; b_{i,j} = 1$. The dimension of S_e was chosen during practical experimentation. Dilation is defined as follows: $(S_e \oplus M'_n)(x, y) = \max\{S_e(i, j) + M'_n(x - i, y - j)|(i, j) \in S_e\}$. Similarly, erosion is defined as $(S_e \ominus M'_n)(x, y) = \max\{S_e(i, j) + M'_n(x + i, y + j)|(i, j) \in S_e\}$. The data used spans a period of 3 years.

For formality, the active region data and observation timestamps are encoded in the following format,

$< AR\_yyyy - mm - ddThhminsms >$ which indicates the active region number, year, month, day, hours time, minutes, and milliseconds, respectively. The training set spans periods of 11386_2012-01-01T070600 (active region number: 11386, in the year 2012, 1^{st} of January, 070600) to 11823_2013-08-16T115000 and makes 80% of the data used. The validation set spans periods of 11823_2013-08-20T085100 to 11897_2013-11-18T110700 and makes 10% of the data used. The test spans periods of 11897_2013-11-18T143700 to 11955_2014-01-22T173500 and also makes 10% of the data used. The overlap in the validation and testing set is only 3 instances from the same active region with AR number 11897, which was ignored (the actual magnetogram sequences are not exactly the same). Instances (sequences) were set to have magnetograms from the same active region and day. If a day had less than 4 observations, it was dropped out of the data. If a day had more than 4 magnetograms that could form another full sequence, the sequence was used as a different instance from the same day as the first one.

Class distribution is important in the partitioning of a machine learning model's training, validation, and test data [75].The test set had $54\% < C$ instances, and $46\% \geq C$ instances. The percentage distributions were similar in the training set. The validation set had $58\% < C$ instances, and $42\% \geq C$ instances. To preserve more samples, stratified sampling was not applied as it would force some instances to be dropped. The aspect of class distribution was considered reasonable since the actual training and test sets had corresponding representations of <C & ≥C solar flares.

Figure 3.6 shows some randomly selected magnetograms (their histograms in Figure 3.7) from the SDO Benchmark Dataset. The issue of similarity from a subjective visual perspective is evident. The magnetograms are labeled with various classes besides their similar appearances. The Q represents 'no solar flare' and C for a solar flare of at least $10^{-6}W/m^2$ up to the class upper bound radioactive energy.

Figure 3.8 shows a magnetogram that is at the edge of the solar disk. Authors [7, 17], usually avoid including these kinds of magnetograms in their data. The issue is that they are constructed from an angle that prevents the observer from having a clear layout of the spatial proximity of the umbra and the penumbra to mention the least. This affects the feature extraction. Besides their ill reputation on model performance, those magnetograms from the edge of the solar disk (with respect to the observer) are included since in reality they will be there and they affect the occurrence of events in the more central regions of the solar disk. The concept that describes how their occurrence may be related to those occurring at a central position (relative to the observer) is referred to as sympathetic flaring [76].

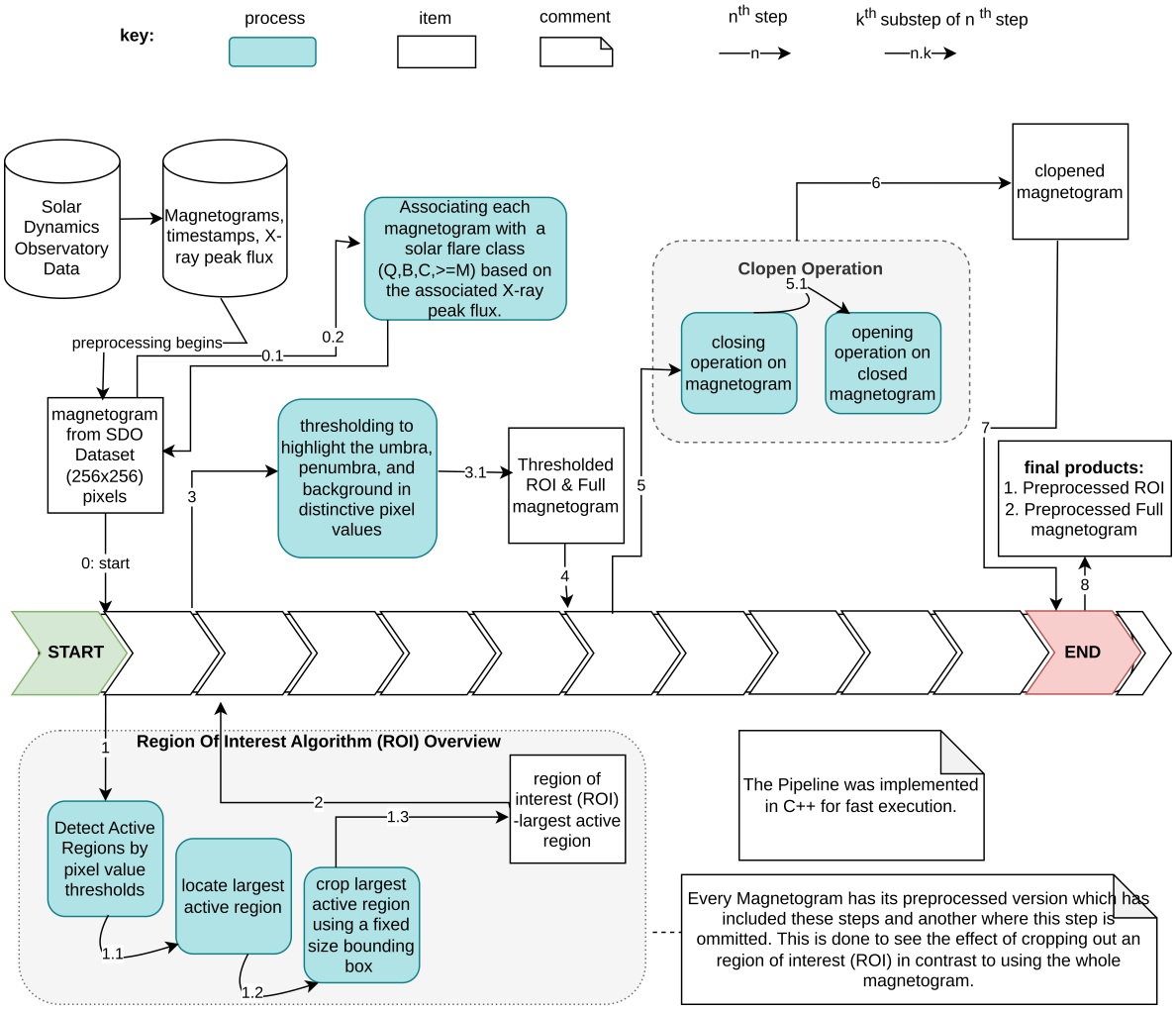

Fig. 3.5: Magnetogram Preprocessing Pipeline

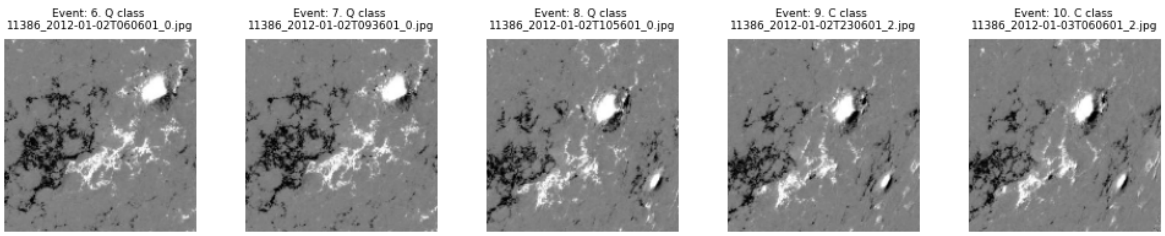

Fig. 3.6: Randomly Selected Sequence of Solar Flare Magnetograms

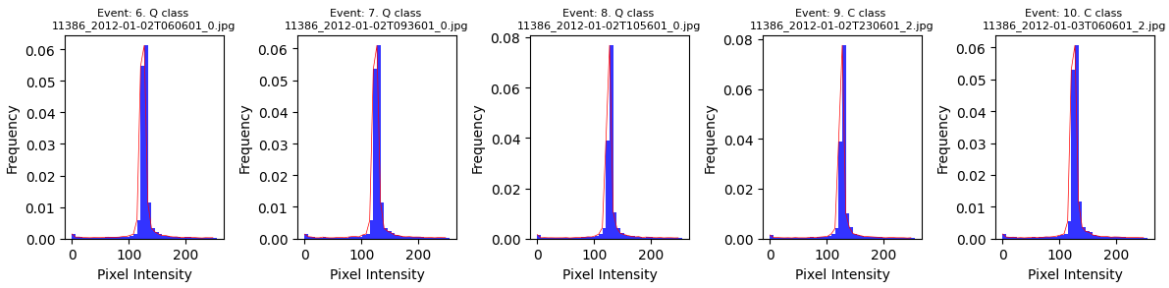

Fig. 3.7: Histograms Corresponding To Magnetograms From Figure3.6

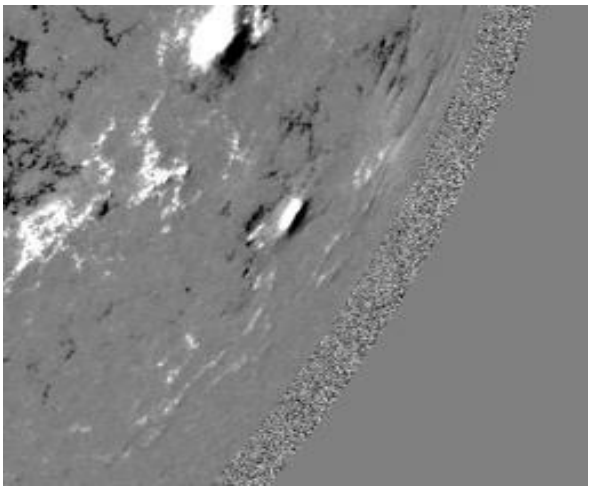

Fig. 3.8: Example of Elusive Magnetogram

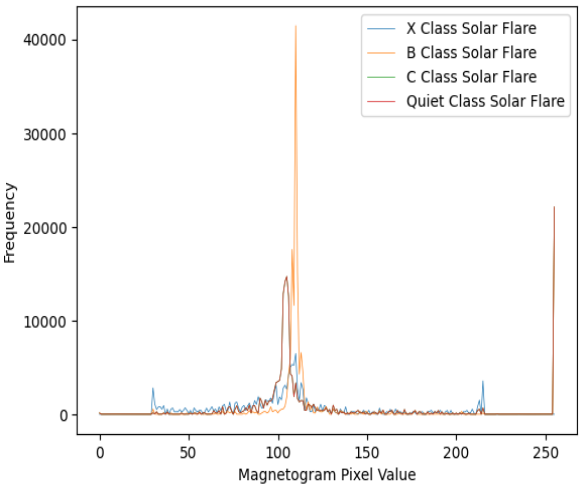

Fig. 3.9: Sample Magnetogram Histograms

4.3 Preprocessing - Message Passing Interface Setup

One of the issues when dealing with large datasets is finding solutions for efficient computation. This issue made it to the list of issues encountered in the experiments in this study. To handle this, the preprocessing of the magnetograms was implemented in a parallel computing setup. A Message Passing Interface (MPI) [77] implemented in Python was used to distribute the preprocessing tasks for each Central Processing Unit(CPU) core. The distributed computing script is adaptive in that in a different machine, it will use the maximum available CPU cores. Considering the slow execution speed of Python in contrast to C++, the actual preprocessing pipeline to run in each CPU core was implemented using C++. The C++ script was called with command-line arguments by each Python-implemented process. Using Python together with C++ was a design choice influenced by the ease of implementation of MPI in Python and leveraging the computational speed of C++ where it is needed. The details of the hardware device are outlined as follows. CPU(s): 4, Vendor ID: Genuine-Intel, Model name: Intel(R) Core(TM) i7-6600U CPU @ 2.60G, CPU family: 6, Model: 78, Thread(s) per core: 2, Core(s) per socket: 2, CPU max MHz: 3400.0000, CPU min MHz: 400.0000. Figure 3.10 shows an overview of the parallel computing setup. Overall, this approach saves computational time which is the common bottleneck when processing large datasets like the one used in this study. The advantage of this method is that it does so without needing extra hardware, the focus is to make the best use of available resources. In contrast to executing the pipeline program in one CPU core, this approach reduced the computational time by $\approx 60\%$. It should be noted that this is dependent on the number of CPU cores used and the size of the data partition being processed. It is worth noting however that in some cases, the use of alternative approaches is inevitable.

Figure 3.11 shows an example of how a magnetogram is affected by the pipeline from Figure 3.10.

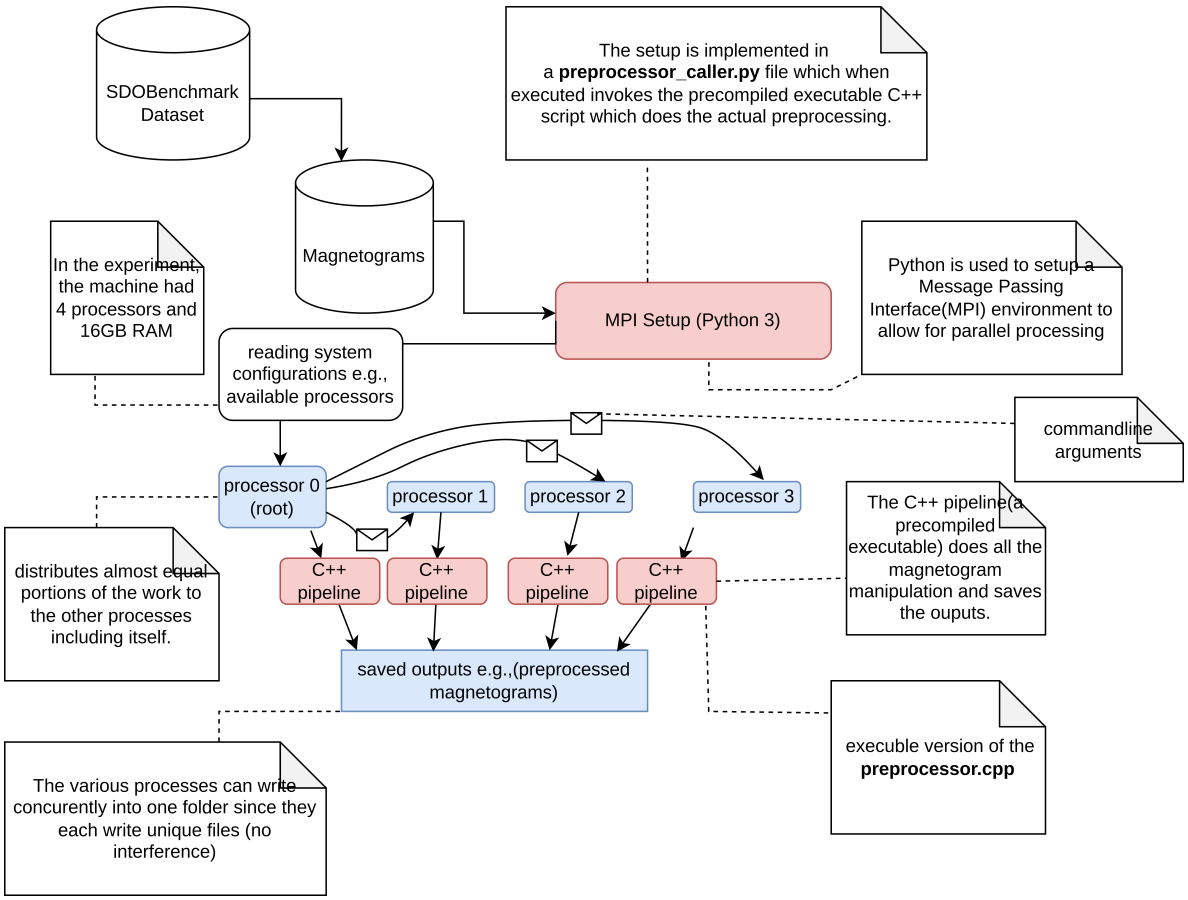

Fig. 3.10: MPI (Python) and Pipelines (C++) Parallel Processing Setup

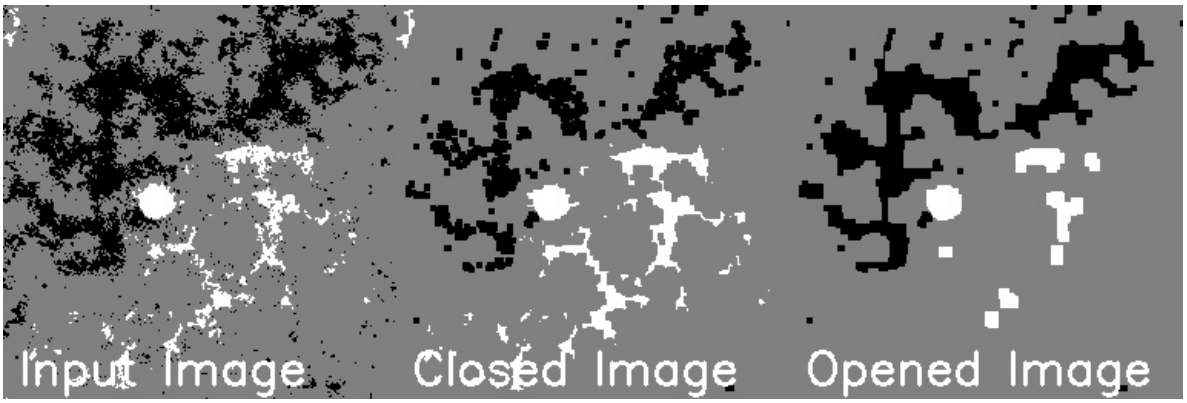

Fig. 3.11: Example Flow of the Clopen Operation On A Magnetogram

Figure 3.11 does not show the region of interest cropping effect. Referring to Figure 3.11, it is evident that the closing operation (see center of Figure 3.11) eliminates the regions with very small darkish and whitish pixel clusters on the thresholded magnetogram. The following opening operation (see right of Figure 3.11) further removes some of the remaining smaller regions and closes gaps. It is critical that this process does not eliminate the magnetic polarity inversion zones, e.g., the area of contact between the whitish and darkish regions (refer to the white circular component in the far left of Figure 3.11 and the black region in contact with it as an example). The areas of magnetic polarity inversion have been proven to be essential for the model's learning as stated in [78].

4.4 Preprocessing - Sequence Preparation

In preparation of the data into sequences, multiple factors were considered. The notation introduced in this part will be used in other sections as well. Let t_n denote the n^{th} observation of a solar flare at some time, hence t_n is part of a sequence of solar flare events. Each term t_n is defined as a four-tuple $< M_n, c_n^M, s_n^M, e_n^M >$ where M_n is the magnetogram feature vector such that $M_n \in R^{150\times150}$, c_n is the class of the solar flare derived from the X-ray peak flux (W/m^2) at the time, s_n^M is the start time, e_n^M is the end time of the solar flare, and $\forall n$, $e_n^M > s_n^M$. Observations contained in one sequence were from the same active region. The number of terms/observations per sequence was 3. It can be noted that the chosen sequence length is small compared to other studies, e.g. 10 from [7]. During the dataset exploration, it was discovered that longer sequences would result in a significant decrease in the number of resulting sequence instances. That would prevent logical/generalizable conclusions from being drawn from results that would come thereafter. The input-to-output mapping was set up to be a many-to-one. Basically, a sequence of 3 magnetograms is labeled with the GOES class of the next magnetogram (the one that would have been the fourth). The classes are binary, they are either $\geq C$ or $< C$. A chronological split was done to prevent active region instance overlaps into training

and testing sets [7, 21].

4.5 Convolutional Long-Short Term Memory

A Convolutional Long-Short Term Memory (ConvLSTM) model was implemented for comparison of the preprocessing methods. The input layer was a Convolutional LSTM layer that takes 2D input since the magnetograms are in grayscale. The kernel size used was a 3×3 kernel. The size was chosen based on experimentation with larger and smaller kernels but is not guaranteed to be the best possible configuration. The Convolutional LSTM layer returns a sequence which will then be passed to the next layer. The activation function used was the Rectified Linear Unit (ReLU) which helped with faster training. The Convolutional RNN layer is followed by a Max Pooling layer with a 2×2 window. A dropout layer with a 0.2 dropout rate follows. The dropout layer helps with combating over-fitting. The pattern of the similar configuration of the Convolutional RNN, followed by Max Pooling and Dropout repeats 3 times. Then a flattening layer follows which precedes the Dense final layer which gives categorical outputs. The difference in the Convolutional LSTM layers was the number of filters used. First, it was 8, then 16, then 24, and finally 32. This made feature extraction efficient in correspondence to the max pooling layers. The resulting model had 128098 trainable parameters. All the models were configured with Early Stopping to optimize training time, model parameters, and hyperparameters. The models were configured to stop training if the validation loss fails to decrease for 7 consecutive epochs. The best-learned weights were restored and used. The learning rate was also reduced by a 10% factor for every 5 consecutive epochs where the validation loss failed to decrease. The same model was trained and tested on 3 versions of the data. The first was the original magnetograms, the second was the clopened full magnetograms and the clopened magnetograms where a region of interest was cropped out and used.

5 Ensemble CNNs for ≥C Solar Flare Recognition At Wavelength 1,600$\mathring{A}$

5.1 Data Pre-processing

The data used was adopted from [18] stored as pickle objects.The data was first deserialized, went through image standardization, then shuffling, and finally batching in readiness to be fed to the CNNs. Image sizes of 250 x 250 pixels were used. There were 966 images in the dataset in total. Data was partitioned into 70%, 15%, and 15% for training, testing, and validation respectively. Each image has its class. Each class was equi-proportionally represented in the data partitions. Figure 3.12 shows the

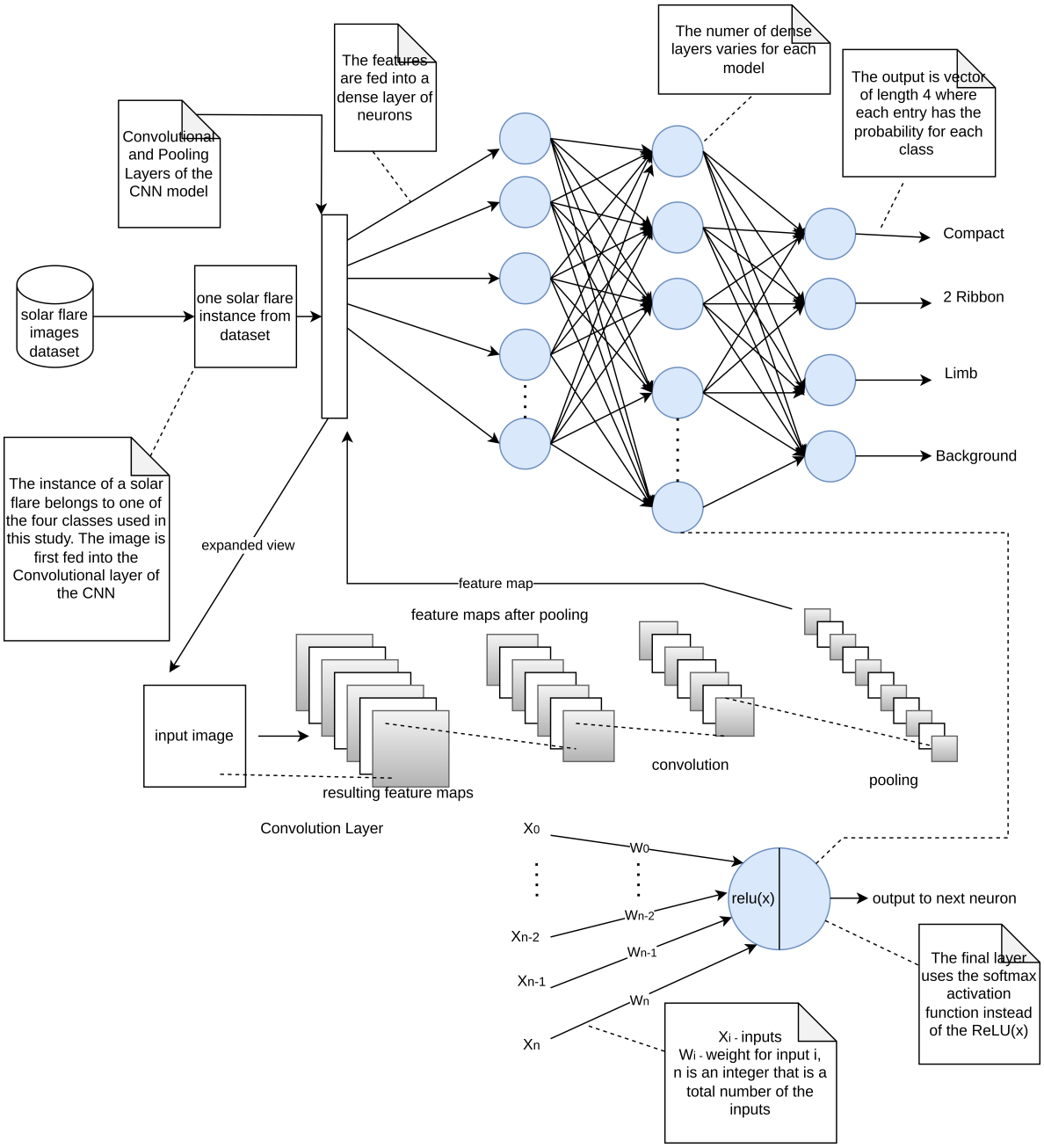

Fig. 3.12: Sample Generic CNN Diagram (inspired by [50], [79], [18])

diagram of the system implemented.

5.2 Similarities Between The Used Individual CNNs

All the individual CNNs use the softmax activation function in their final layer. The softmax is defined in Equation 3.2.

$$softmax(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}} \tag{3.2}$$

where x_i is the input vector, n is the number of classes and e is the standard Euler's number derived function [80]. The softmax computes the probability that the image belongs to a certain class, in this case, one of the four classes. The softmax was chosen since the data that needs to be classified is multinomial and it deals with multinomial probability distributions well. With exception to the LeNet5 model, the other models used in this study were configured to make use of the categorical cross-entropy loss function, defined in Equation 3.3.

$$CE(x_i, y_j) = \sum_{i=1}^{C} x_i log(y_j) \tag{3.3}$$

where y_j is the j^{th} value of the mode's possible outputs and x_i being the actual value expected [81]. The categorical cross entropy was chosen because the problem at hand is a multi-class classification problem. The hidden layers used the Rectified Linear Unit (ReLU) activation function defined in Equation 3.4.

$$ReLU(x) = \begin{cases} 0, & \text{if } x < 0 \\ x, & \text{if } x \geq 0 \end{cases} \tag{3.4}$$

where x is the numerical input. This ReLU's strength is not being affected by the vanishing gradients issue. The ReLU was chosen for its positive influence on the model's training speed in contrast to its well-known alternatives. The final layer has four neurons for each CNN model, classifying an image into one of the four classes in the data. The models were configured with early stopping to stop training if the validation loss does not improve after some epochs, this was done to save time. The main means for reducing (optimizing) training time are: using the ReLu, using the EarlyStopping configuration, model optimizers, and the GPU configuration for model training.

5.3 Differences Between The Used Individual CNNs

Differences in CNNs were deliberate means to increase diversity. The use of different optimizers on CNNs increases their diversity [82]. In this study, five different CNN architectures were each configured with a different optimizer. Optimizers were chosen based on their competence as seen in other studies. According to [83], heterogeneous base learners help ensembles to be successful. The

AlexNet model used the Stochastic Gradient Descent, ResNet50 used the Adadelta, NASNetLarge used the Adagrad, XceptionNet used the RMSprop and LeNet5 used the Adam optimizer. Table 3.2 shows how the models differ in their configurations. The learning rate for each model was selected because it was producing better results than alternatives during the experiments. The optimizers were chosen for performance maximizing and variety amongst the CNNs. Each model was shown to learn best (or faster) when using a certain batch size. The selection of batch sizes also varied amongst most of the models except the AlexNet CNN and the XceptionNet CNN.

Larger models like the ResNet50 required larger batches for efficient training. The number of epochs used was largely dependent on the size of the model, batch size, and the optimizer. This resulted in varied epoch configurations for the models for efficient training. The column named 'Patience Epochs' indicates the number of training epochs for which the model was configured to use as a stopping criterion if the validation performance does not improve for those specific epochs consecutively in training. For example, the 5 for AlexNet means that if the validation loss does not decrease in 5 consecutive epochs, then the model should stop training. This was done to prevent over-fitting. The choice for patience epochs was subjective but based on experimental observations. This number can vary depending on the complexity of the model and the data.

Table 3.2: Configurations For CNN Models

Model	Learning Rate	Optimizer	Batch Size	Epochs	Patience Epochs	Momentum
AlexNet	0.002	SGD	32	25	5	0.5
XceptionNet	0.0011	RMSProp	32	60	11	0.001
LeNet5	0.015	Adam	16	18	8	-
NASNetLarge	0.0017	Adagrad	24	65	10	-
ResNet50	0.0018	Adadelta	64	100	6	-

The following bullet points each provide justifications of relevance of the methods used. Each model is justified for its success in a relevant application, even if it was in a different context. It is important to note that the bullet points are not meant to provide a comprehensive review of the relevant studies but a brief justification of relevance. Tables 3.3, 3.4, 3.5, 3.6, and 3.7 each shows the architectural details of the CNN models used. In cases where a base model was used, it was taken from TensorFlow's Python framework developed by Google Brain [84]. Models whose bases were taken from TensorFlow were imported without pre-trained parameters. The additional layers were customized in the experiment for this study to best suit the models for the task at hand (e.g., the obvious final layer with 4 classes). Other layers were added to create variety while maximizing performance and creating room for exploring how larger models will deal with the problem, e.g., the ResNet50 model. This was done because, in

Chapter 1, it was hypothesized that the architecture of models has an impact on performance. The use of the dropout was done to combat over-fitting.

- **AlexNet CNN** - AlextNet was introduced by [85] in the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) where it was the top-performing model. So the model on its own has a good reputation. The properties of medical images are very similar to those of solar flare ribbons [51]. A study [86] shows how the AlexNet model was successful in the visual recognition of diabetic retinopathy images. It obtained at least 95.6% accuracy in various stages of diabetes. The images of the retina used in that study are very relatable to the problem at hand. Its original architecture can be found in [87], which shows that it is a simple model that is easy to train. It is the ability of this model to do well in medical imaging that led to the justification of its relevance in this context based on the aforementioned facts.

Table 3.3: The Fine-tuned AlexNet CNN Architecture

Layer Type	**Parameters**	**Output Shape**
Convolutional	Filters: 128, Kernel: (11, 11), Strides: (4, 4)	(62, 62, 128)
Batch Normalization	-	(62, 62, 128)
Max Pooling	Pool Size: (2, 2)	(31, 31, 128)
Convolutional	Filters: 256, Kernel: (5, 5), Strides: (1, 1)	(31, 31, 256)
Batch Normalization	-	(31, 31, 256)
Max Pooling	Pool Size: (3, 3)	(10, 10, 256)
Convolutional	Filters: 256, Kernel: (3, 3), Strides: (1, 1)	(10, 10, 256)
Batch Normalization	-	(10, 10, 256)
Convolutional	Filters: 256, Kernel: (1, 1), Strides: (1, 1)	(10, 10, 256)
Batch Normalization	-	(10, 10, 256)
Convolutional	Filters: 256, Kernel: (1, 1), Strides: (1, 1)	(10, 10, 256)
Batch Normalization	-	(10, 10, 256)
Max Pooling	Pool Size: (2, 2)	(5, 5, 256)
Flatten	-	6400
Dense (Fully Connected)	Neurons: 1024, Activation: ReLU	1024
Dropout	Rate: 0.5	1024
Dense (Fully Connected)	Neurons: 1024, Activation: ReLU	1024
Dropout	Rate: 0.5	1024
Dense (Output)	Neurons: 4, Activation: Softmax	4

- **XceptionNet CNN** - In a COVID-19-based study [88], A modified version of XceptionNet demonstrated success in image recognition when compared to some of the most recent models. As mentioned earlier, medical images are similar to solar events' images, this makes the use of XcpetionNet more reliable. This study however uses the standard XceptionNet model with modifications as presented in Table 3.4.

Table 3.4: The Fine-tuned XceptionNet-based Neural Network Architecture

Layer Type	**Parameters**	**Output Shape**
Xception Base	Input Shape: (250, 250, 1), Pooling: Max	(None, 2048)
Dense (Fully Connected)	Neurons: 512, Activation: ReLU	512
Dense (Fully Connected)	Neurons: 64, Activation: ReLU	64
Dense (Output)	Neurons: 4, Activation: Softmax	4

- **LeNet5 CNN** - The LeNet CNN was first introduced in [55]. Named after Yann LeCun, the model has been applied in the recognition of complex graphical figures like handwritten digits [55]. A later variation named LeNet5 was applied in solar event detection [89], where it was successful. The model has also been successfully applied in predicting the arrival time of coronal mass ejections (solar events) [90]. The slightly tuned version used in this study is shown in Table 3.5.

Table 3.5: LeNet-5 Inspired Neural Network Architecture

Layer Type	**Parameters**	**Output Shape**
Convolutional	Filters: 32, Kernel: (5, 5), Input Shape: (250, 250, 1)	(250, 250, 32)
Max Pooling	Strides: 2	(125, 125, 32)
Convolutional	Filters: 48, Kernel: (5, 5) Activation: ReLU	(121, 121, 48)
Max Pooling	Strides: 2	(60, 60, 48)
Flatten	-	172800
Dense (Fully Connected)	Neurons: 256, Activation: ReLU	256
Dense (Fully Connected)	Neurons: 84, Activation: ReLU	84
Dense (Output)	Neurons: 4, Activation: Softmax	4

- **NASNetLarge CNN** - In another COVID-19-related study [91] using X-ray images, the NASNetLarge CNN model was among other well-performing models in a recognition task meant to aid an automated diagnosis of COVID-19-related infections. X-ray images fall under the category of medical images. Other recent works also applied this model to medical

images [92]. The fine-tuned version of this model used in this study is provided in Table 3.6.

Table 3.6: The Fine-tuned NASNetLarge-Based Neural Network Architecture

Layer Type	**Parameters**	**Output Shape**
NASNetLarge Base	Input Shape: (250, 250, 1),	(None, None, 4032)
Dropout	Rate: 0.5	(None, None, 4032)
Flatten	-	4032
Batch Normalization	-	4032
Dense (Fully Connected)	Neurons: 32, Kernel Initializer: He Uniform	32
Batch Normalization	-	32
Activation (ReLU)	-	32
Dropout	Rate: 0.5	32
Dense (Fully Connected)	Neurons: 32, Kernel Initializer: He Uniform	32
Batch Normalization	-	32
Activation (ReLU)	-	32
Dense (Output)	Neurons: 4, Activation: Softmax	4

- **ResNet50 CNN** - In the task of brain tumor detection, a study [93] showed that the ResNet50 model was a good candidate for the recognition/classification of images with MRI scans. MRI scans are medical images and these are similar to solar events' images, thus making the ResNet50 model a good candidate base learner. The model has been slightly fine-tuned for use in this study and the resulting architecture is summarized in Table 3.7.

5.4 Measuring Diversity of individual CNNs

Table 3.8 shows some tabulated metrics of the diversity measure on some of the models. Previous utilization of the diversity measure can be found in [60]. Let P_{m_1,m_2} be a prediction of models m_1 and m_2 on some test data. Assume that 0 and 1 mean wrong and correct classification, respectively, such that $P_{0,1}$ is a set of cases were m_1 was wrong and m_2 was correct. Using a similar analogy, the diversity measure for m_1 and m_2 is:

$$D_{m_1,m_2} = \frac{|P_{0,1}| + |P_{1,0}|}{|P_{0,1}| + |P_{1,0}| + |P_{0,0}| + |P_{1,1}|} \tag{3.5}$$

where D_{m_1,m_2} measures how often the models m_1 and m_2 disagree when they are tested. For the sake of presentation. Table 3.8 shows the disagreement of the models. The weakness of this metric is that it does not give clues about whether models were both wrong in their disagreements or not.

Table 3.7: The Fine-tuned ResNet50-Based Neural Network Architecture

Layer Type	**Parameters**	**Output Shape**
ResNet50 Base	Input Shape: (250, 250, 1)	(None, None, 2048)
Dropout	Rate: 0.5	(None, None, 2048)
Flatten	-	2048
Batch Normalization	-	2048
Dense (Fully Connected)	Neurons: 2048, Kernel Initializer: He Uniform	2048
Batch Normalization	-	2048
Activation (ReLU)	-	2048
Dropout	Rate: 0.5	2048
Dense (Fully Connected)	Neurons: 1024, Kernel Initializer: He Uniform	1024
Batch Normalization	-	1024
Activation (ReLU)	-	1024
Dropout	Rate: 0.5	1024
Dense (Output)	Neurons: 4, Activation: Softmax	4

Table 3.8: The disagreement measure for some of the CNNs

	AlexNet	XceptionNet	LeNet5	NASNetLarge	ResNet50
AlexNet	-	2.055×10^{-2}	6.85×10^{-3}	1.370×10^{-2}	8.23×10^{-2}
XceptionNet	-	-	1.370×10^{-2}	2.050×10^{-3}	8.90×10^{-2}
LeNet5	-	-	-	6.85×10^{-3}	8.90×10^{-2}
NASNetLarge	-	-	-	-	8.21×10^{-2}
ResNet50	-	-	-	-	-

A better alternative method to measure diversity is the Q-statistic as discussed in [60]. Its result can be used to deduce more information about a pair of classifiers than with the disagreement measure. Using a similar notation with the disagreement measure, the Q-statistic for 2 classifier models m_a and m_b is computed as:

$$Q_{a,b} = \frac{(|P_{1,1}| * |P_{0,0}|) - (|P_{0,1}| * |P_{1,0}|)}{(|P_{1,1}| * |P_{0,0}|) + (|P_{0,1}| * |P_{1,0}|)} \tag{3.6}$$

For any pair of classifiers, m_a and m_b, $Q_{a,b} > 0$ implies that m_a and m_b have a tendency of making correct predictions. $Q_{a,b} < 0$ implies that m_a and m_b have a tendency to both make wrong predictions. $\forall Q_{a,b}$, $Q_{a,b} \in [-1, 1]$, $Q_{a,b} \in R$. The computation of the Q-statistic show that both the AlexNet and XceptionNet have a value of -1 against the NASNetLarge and also against each other. Both the AlexNet and XceptionNet have a value of 0 against the LetNet5. The AlexNet and NASNetLarge both have a value of 1 against the ResNet50 model. The XceptionNet has a value of 0.563 against the ResNet50 model and lastly the LeNet5 with a value of 0 against the ResNet50. There are more non-negative values in the Q-statistic values for pairs of models, which is a good indication of the models' correct classifications. The negative values indicate room for the models to complement each other's weaknesses in the ensemble model if they can.

5.5 Computing Platform Specifications

The individual models were trained sequentially. The Google Collab environment was utilized for the experiments. The Graphics Processing Unit (GPU) configuration was used to save time. The device details are Intel (R) Xeon (R) CPU @2..00GHz, Random Access Memory (RAM): 13297228 Kilobytes, CPU cores: 2 threads per core and CPU, Programming Language used were Python3 (3.7.15) and C++.

6 Conclusion

This chapter discussed the methods used in this study to close the gaps highlighted in previous research. It also validates the use of features from magnetograms where there is minimal preprocessing. The chapter presented the candidate methods and justified their suitability for the study. The next chapter focuses on the discussion of the results obtained.

Chapter 4: Results and Discussion

1 Introduction

This chapter reports and discusses the results achieved in this study. Firstly, the results of ensembles and individual RNNs are presented, discussed, and compared. Thereafter, the results are further analysed to show how reparameterization can be used in ensembles to make a further impact on calibration and ensemble performance. Lastly, the chapter focuses on the analysis of how ensembles can be used to improve visual recognition of solar flares observed at wavelength 1,600$\overset{\circ}{A}$.

2 Ensemble RNNs For $\geq$ C Solar Flare Recurrence Prediction

Figures 4.3, 4.6, 4.9, 4.12, 4.15 show the confusion matrices obtained from some of the models. It is clear that all the models had True Positives (TP) and True Negatives (TN) that are at least 0.8. This means that at least 80% of $\geq$C solar flares were actually predicted to be solar flares and similarly with $<$C instances. Figures 4.6 and 4.3 show how the GRU and LSTM tend to correctly predict the negative class $<$C much better than the $\geq$C instances, as seen in their TNs being significantly larger than the rest of the models. By looking at the context of the application, the most preferred model(s) are those that are most superior in predicting TPs, e.g., the Simple RNN (see Figure 4.9) and Soft Voting Ensemble (see Figure 4.12). The ROC curves in Figures 4.2, 4.5, 4.11, 4.8, 4.14 show how the RNN models compare in terms of their degree of separability of positive and negative instances of $\geq C$ class flares. The AUC measures the area under the ROC curve. A lower AUC value means that the associated model has a lower ability to distinguish between sequences that lead to $\geq C$ or $< C$. A higher AUC is usually associated with higher precision scores as the trend can be seen in the corresponding confusion matrices. A good example is the Soft Voting Ensemble (SVE) with the best AUC = 0.87 in Figure 4.11 and the corresponding best TP = 0.9 in Figure 4.12. The general trend of the precision vs. recall curves in Figures 4.1, 4.4, 4.7, 4.10, 4.13 shows that the models tend to trade off precision for recall for $\geq$C flares (class 1 in the mentioned figures). The downside of this is that

Table 4.1: Q-statistic for LSTM, GRU and SimpleRNN

	LSTM	GRU	SimpleRNN
LSTM	-	0.998	0.995
GRU	-	-	0.998
SimpleRNN	-	-	-

the false positive rate increases. In a real application, the models are likely to predict a sequence of observations in a $\geq C$ flare when it is a $< C$ instance. The advantage of this trade-off is the models will be less likely to miss a strong solar flare e.g. an X class solar flare as it is in ≥C solar flares. Overall, the aim is to save lives and reduce the risk of space equipment damage. It is not difficult to see that if the model is not good enough, it must at least predict most of the dangerous events ($\geq C$ flares) correctly even if it means having false alarms.

Table 4.1 shows the Q-statistic for the LSTM, GRU, and Simple RNN models which were trained on the entire training set (i.e., the data that spans 2010-2014). A Q-statistic value q is such that $q \in R$, $q \in\approx [-1, 1]$. A -1 shows that the two models are highly disagreeable, a 1 indicates that models are highly agreeable, and a 0 shows that the models are making worse guesses than random [60] guesses. All the models LSTM, GRU, and Simple RNN demonstrated to be highly agreeable in the experiment. The high agree-ability is part of the reason for the heterogeneous stacking ensemble's failure to surpass the individual models. This is mainly because, in the training of the meta-model, there was no significantly better way to combine the predictive powers of the models such that they compensate for each other's weaknesses due to their mostly similar predictions. The base learners of the heterogeneous ensemble lacked sufficient and practically demonstrable diversity hence the failure of the ensemble [48]. The results of the heterogeneous stacking ensemble can be seen in Table 4.2 where it is denoted as HtrSE. SVE and HVE abbreviating Soft-voting and hard-voting ensembles, respectively. Table 4.1 highlights the weakness of the Q-statistic. It is true that the models had a high agree-ability. However, it does not rule out the fact that there were differences in categorical probability distribution in the predictions even when they both had the same one hot encoding or the same prediction. For example, if for classes $[\geq C, < C]$ two models predict $[0.1, 0.9]$ and $[0.4, 0.6]$, according to the Q-statistic, both models agree 100% of the time but it does not reflect how they differ at the probabilistic level.

The third row of Table 4.2 shows that the model with the most ideal performance is the Simple RNN. This follows after it scores the highest on the critical metrics, namely, f1-score, precision, recall, and BACC.

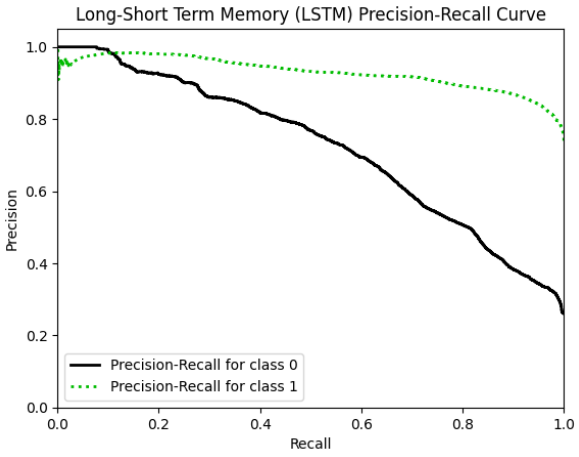

Fig. 4.1: LSTM Precision-Recall Curve

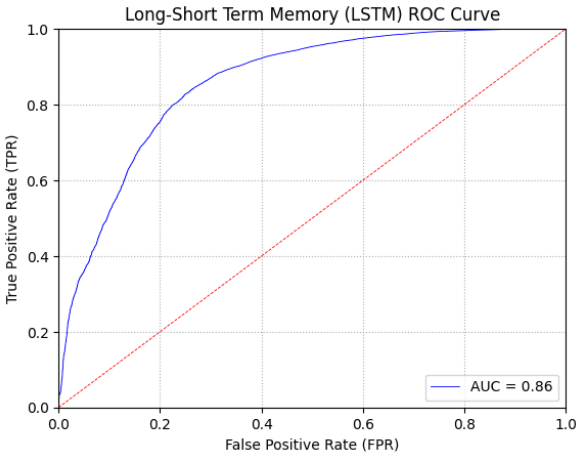

Fig. 4.2: LSTM ROC Curve

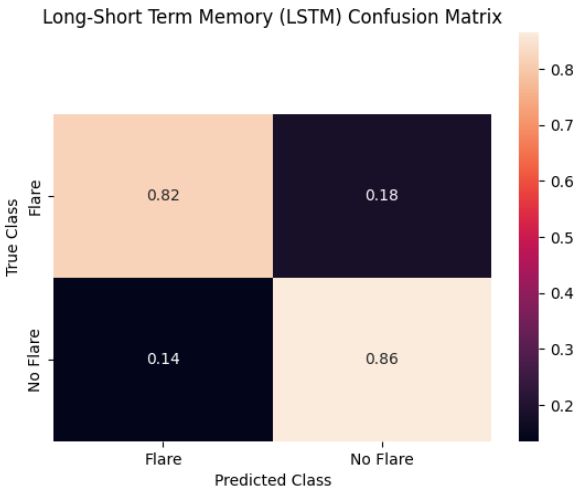

Fig. 4.3: LSTM Confusion Matrix

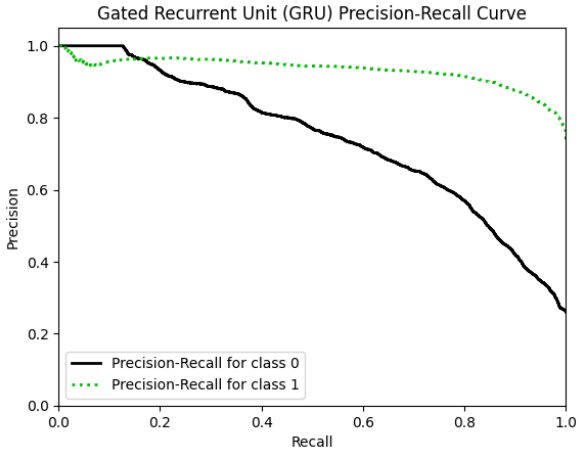

Fig. 4.4: GRU Precision-Recall Curve

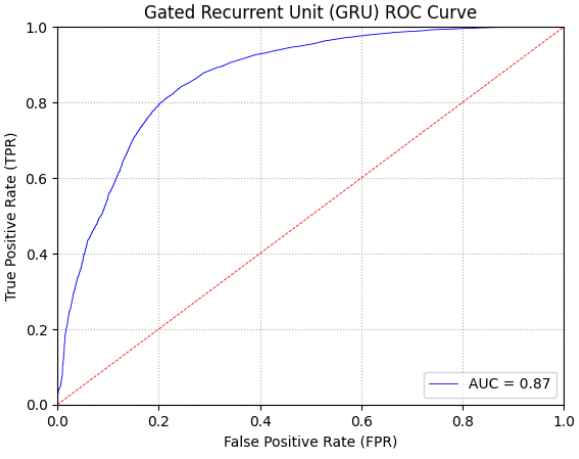

Fig. 4.5: GRU ROC Curve

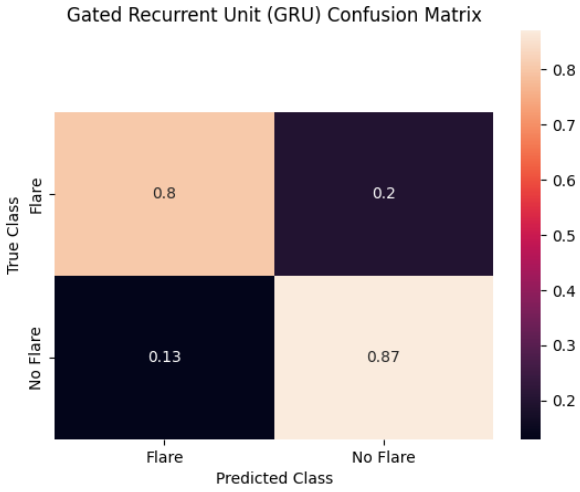

Fig. 4.6: GRU Confusion Matrix

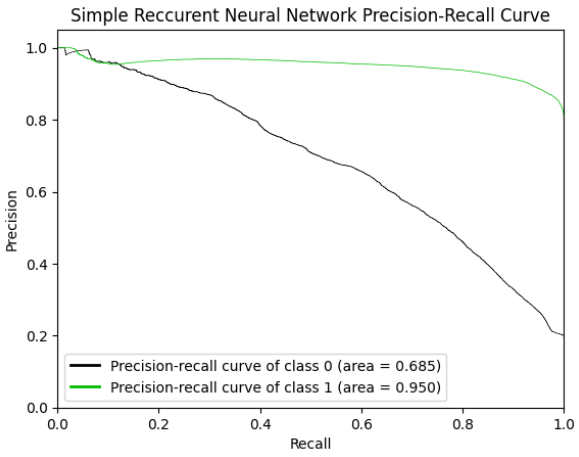

Fig. 4.7: Simple RNN Precision-Recall Curve

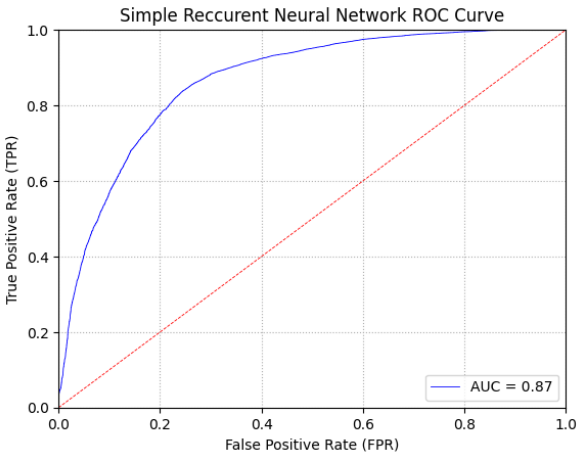

Fig. 4.8: Simple RNN ROC Curve

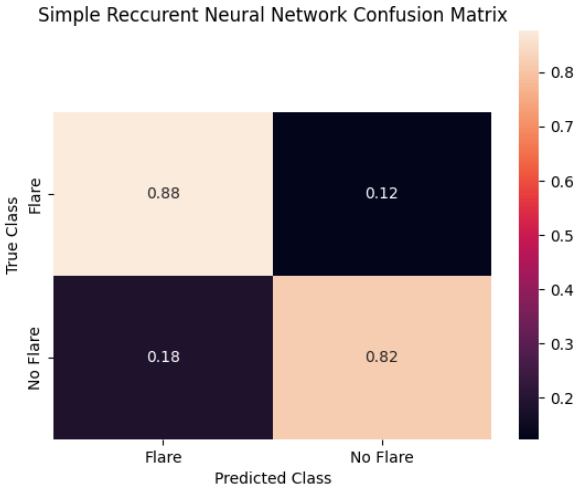

Fig. 4.9: Simple RNN Confusion Matrix

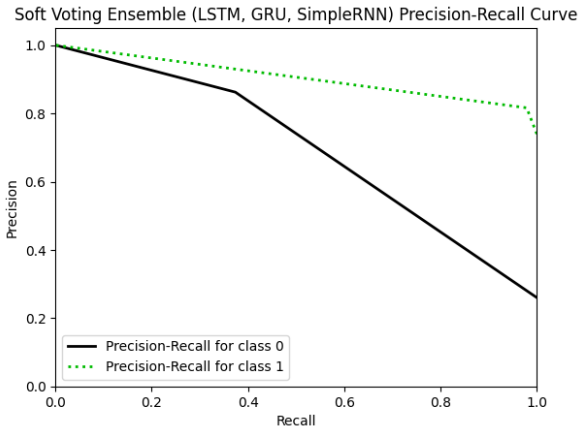

Fig. 4.10: SVE Precision-Recall Curve

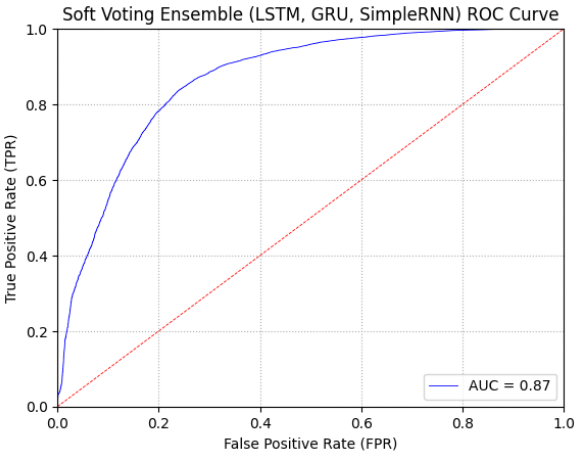

Fig. 4.11: SVE ROC Curve

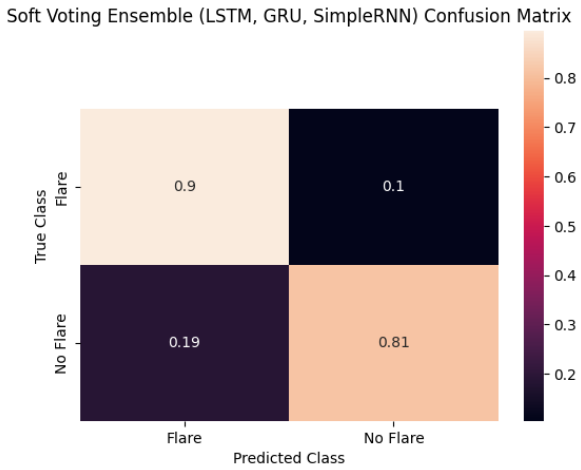

Fig. 4.12: SVE Confusion Matrix

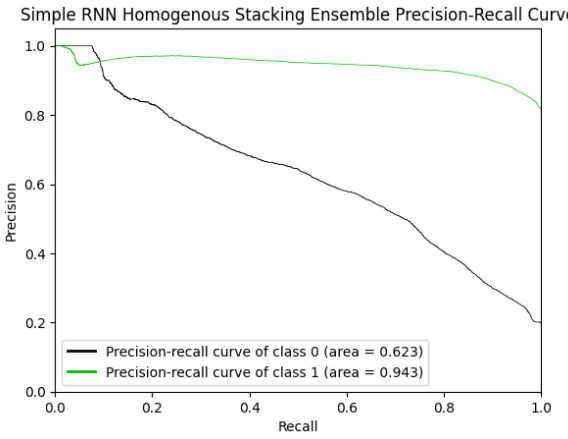

Fig. 4.13: SimpleRNN SE Precision-Recall Curve

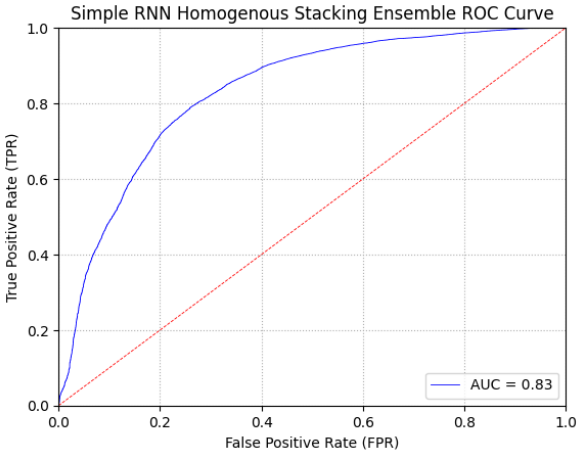

Fig. 4.14: SimpleRNN SE ROC Curve

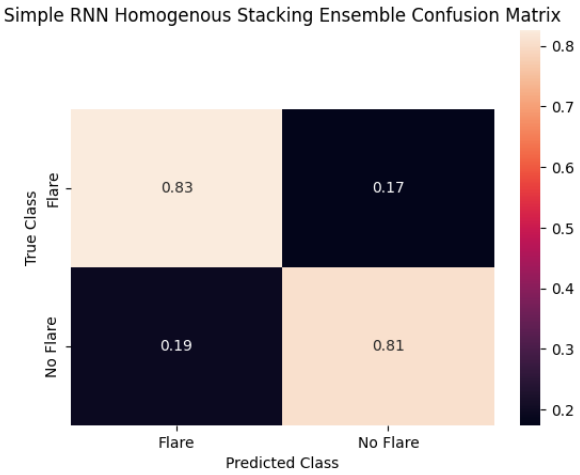

Fig. 4.15: SimpleRNN SE Confusion Matrix

Table 4.2: Results on TSS, BACC, f1-score and HSS2 Scores

Model	TSS	BACC	f1-score	HSS2	Precision	Recall
LSTM	**0.70**	**0.85**	0.77	0.38	**0.83**	0.81
GRU	0.67	0.83	**0.79**	0.40	0.82	**0.82**
SimpleRNN	**0.70**	**0.85**	**0.79**	**0.43**	**0.83**	**0.82**
HVE	**0.70**	**0.85**	0.78	0.40	**0.83**	**0.82**
SVE	**0.70**	**0.85**	**0.79**	0.42	**0.83**	**0.82**
HtrSE	0.63	0.81	0.78	0.38	0.81	0.81
GRU SE	0.68	0.84	0.78	0.39	**0.83**	0.81
LSTM SE	0.63	0.81	0.78	0.38	0.81	0.81
SimpleRNN SE	0.63	0.81	0.78	0.38	0.81	0.81

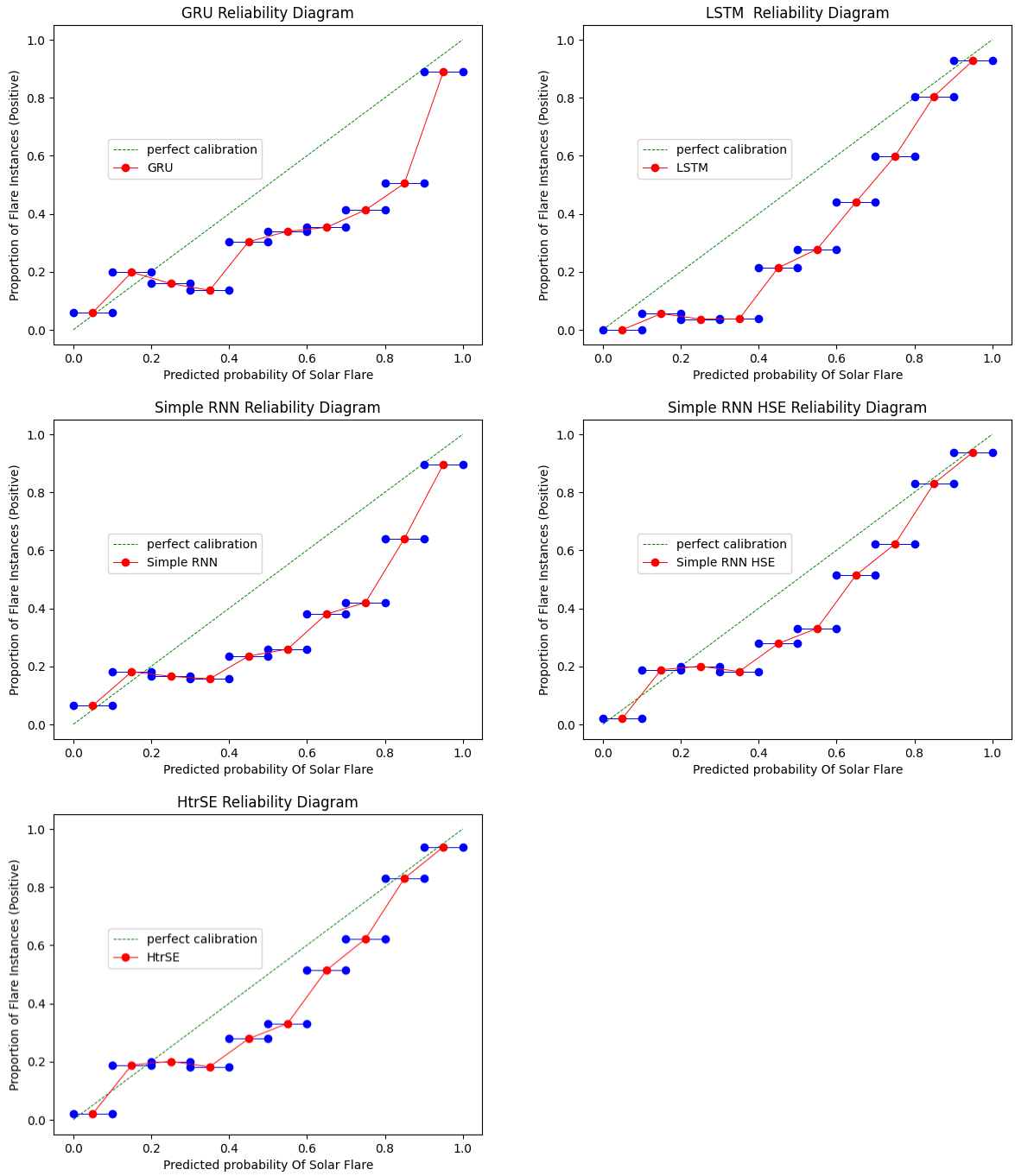

Fig. 4.16: Reliability Diagrams for RNN models

Figure 4.16 shows reliability/calibration diagrams for GRU (top left in Figure 4.16), LSTM (top right in Figure 4.16), SimpleRNN (top right in Figure 4.16), and SimpleRNN Homogeneous (top mid-right in Figure 4.16) and Heterogeneous Stacking Ensemble (bottom left in Figure 4.16) models used in this study. The abbreviations Simple RNN HSE (bottom right graph in Figure 4.16) and HtrSE (bottom left graph in Figure 4.16) are for Simple RNN Homogeneous Stacking Ensemble and Heterogeneous Stacking Ensemble, respectively. The diagonal from $(0, 0)$ to $(1, 1)$ represents perfect calibration. That is a case when the model gives the best possible forecasting of the $\geq C$ flares. The mean predicted value for the range marked in blue is plotted against the fraction of positive instances in the data (the actual flares belonging to $\geq C$). A high prediction probability relative to the proportion of corresponding instances means the model was more confident. Besides the inferior performance of the ensemble models, the calibration curves show that the ensemble models are well-calibrated (see Figure 4.16 mid-right and bottom left) in comparison to individual models (see Figure 4.16 top left, top right and mid-left). Part of the reason is that the meta model's objective was to find the best way to combine the predictive ability of its base learners. The different predictions for the same input from each base learner allowed the ensembles to generalize better in the case of the stacking ensembles.

2.1 Comparison With Previous Works

Some of the previous studies results can be seen in Table 4.3. Studies that share some of the data and type of method(s) used are [7] and [2]. The study in [5] used a different dataset and implemented the individual RNNs. The study in [21] used a method specialized for solar flare forecasting. The models implemented in this study can be seen in bold. This study obtained a higher TSS, f1-score, balanced accuracy, precision, and recall compared to previous studies. It is important to note that [7] used cross-validation but did not stratify their training samples. This study used stratified sampling but not cross-validation. There are other studies on solar flares that may have obtained better results in different contexts, i.e. using other models or a different dataset. The relationship between TSS and HSS2 shows that the RNNs model obtains a better TSS score sacrificing the HSS2 metric, in comparison to the result reported in [7]. The TSS is more important than the HSS2 in this context. That is because the RNN models perform better than random forecasting in all tests. So, the concern was how better than random forecasting are they. The TSS metric addresses that concern. Some studies have been completely dedicated to increasing the TSS [21].

Based on the third row of Table 4.2, the Simple RNN model emerges as best for the task of solar flare forecasting based on the data used in the study. By Occam's razor principle [94], the Simple RNN is preferred for its simplicity, in contrast to the most comparable LSTM model. It can be noted that

Table 4.3: Previous solar flare forecasting studies for C or $\geq C$-class flares

Author	TSS	BACC	f1-score	HSS2	Precision	Recall
Liu et al. [7]	0.61	0.80	-	0.54	0.54	0.76
Wang et al., [2]	0.56	-	-	0.57	0.68	-
Platts et al., [5]	-	-	0.66	-	066	0.66
LSTM	0.70	0.85	0.77	0.38	0.83	0.81
SimpleRNN	0.70	0.85	0.79	0.43	0.83	0.82

$\geq M$ and $\geq M.5$ are solar flares that are more predictable using the dataset that was utilized in the experiments in this study [7]. Let's recall that this study forecasts $\geq C$ class flares. That includes M, $M.5$ and X class flares. One possible room for difficulty for the models was that C class flares rely more on historical data to be predictable [17]. M and $M.5$ class flares are not so reliant on historical information as C class flares. The models in this study may have tried to use historical information on instances of M and $M.5$ class flares with the same weights (i.e. W_{fh}) as in instances that fall strictly under C class flares. This can result in erroneous predictions. To alleviate this issue, multi-class classification seems to be a possible solution in the hope that there is enough data to allow the models to learn the feature ranking for each class. The individual classes C, M and $M.5$ were available in almost equal proportions in training, testing, and validation data. Based on existing literature, the ensemble models are more likely to strictly outperform the individual models. The size of the data used in training standalone individual models is bigger than that used to train base learners for stacking ensembles. This is because the data partitions used to train each base learner are distinct from those used to train the meta-model. Although in some cases this might not be a problem, it may at times decrease the generalization ability of the base learners since they are trained on less data. Ultimately, the ensemble model is not guaranteed to outperform the individual models. This explains why in some metrics the standalone models did better.

3 Reparameterization for ≥C Solar Flare Prediction Ensemble Calibration

Figure 4.17 shows a sample relationship of τ and the True Skill Statistic (TSS), Balanced Accuracy (BACC), and validation accuracy. In that sample, T_min, T_max, and delta_T were set to 0.05, 2.9, and 0.1 respectively (refer to the Simulated Annealing (SA) algorithm in Figure 3.4). The parameters of the SA were selected to allow τ to take on values in ranges $\tau \in (0, 1)$ and $\tau > 1$ to search for values of τ that have various implications on the performance of the LSTM as defined in the Gumbel-Softmax

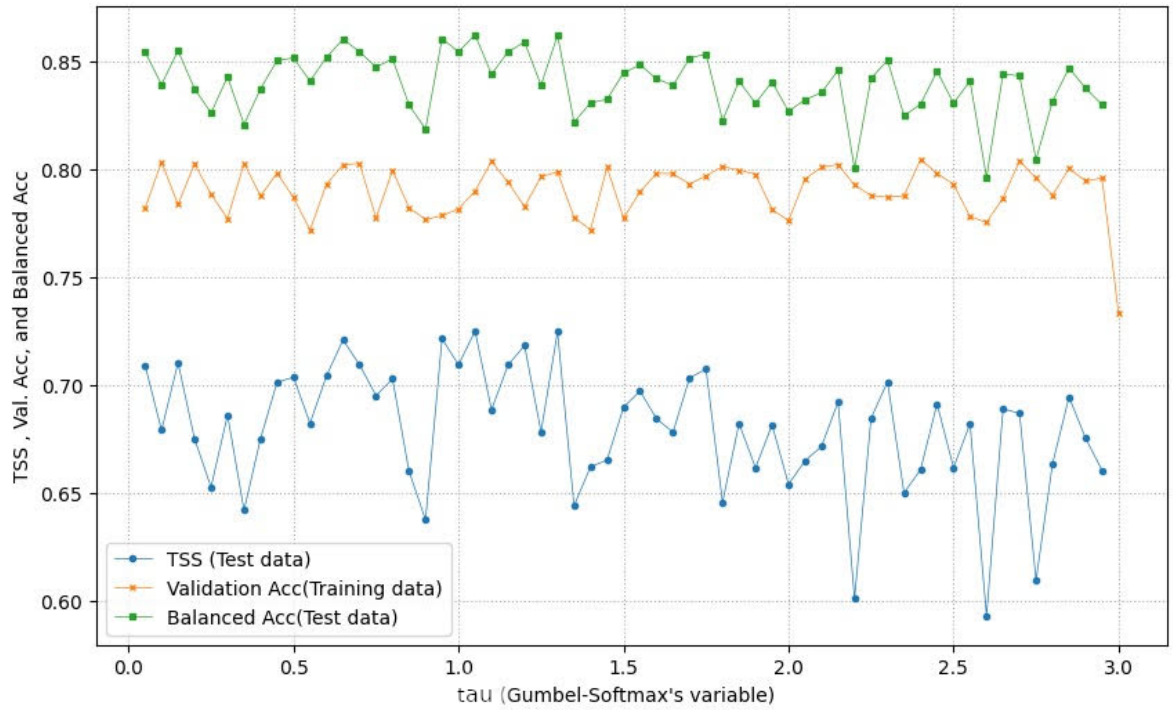

Fig. 4.17: Sample Simulated Annealing Algorithm Search for Optimal τ

equations. The SA was run several times with various τ start, stop, and increment parameters. Each point in the 3 line graphs in Figure 4.17 is associated with a unique value of τ and is generated from a full training and testing of the LSTM model configured with the value of τ. On average, $\approx 58 \pm 1$ iterations with 20 epochs each were used. Based on the 58 iterations, there are some quantitative descriptors that enable to give a stronger intuition about the relationship of the variables involved. The correlation coefficient of the TSS & BACC, $\rho_{TSS,BACC}$ was $\approx 0.\overline{99}$. This is an expected and most trivial relationship since the metrics have similar meanings. Both the TSS and the BACC had a correlation $\rho \approx -0.34 \pm 0.03$ with τ. The relationship means that increasing τ will not always increase the TSS and the BACC . Although theoretically $\rho \in R^{-}$ does not mean that one variable causes a decline in the other, in this setting, most factors are controlled or fixed across iterations. These include training data size, partition sizes, data instances, epochs, model architecture, the range of the random variable U_i of the GS which is constrained in $(0, 1)$, and τ in each data point in the graph. The fact that other influential factors were controlled (or constrained within fixed ranges) can be used to hypothesize that the increase in τ allows for more relaxation in the binary categorical distribution of the predictions made by the LSTM model which declines performance. Another critical relationship is the correlation of the TSS with the validation accuracy, which was $\rho \approx 0.09 \pm 0.005$. This means that there is no linear relationship between the validation accuracy and TSS. This fact hints at the caution that should be exercised when one chooses to do internal annealing of τ in model training

because it will be adjusted based on the model's validation performance yet that does not reflect fully how the model will perform on the test data. The mean BACC $\bar{x}_{BACC}$ was $\approx 0.83 \pm 0.014$, which means that for most of the values of τ, the model performed generally well. The TSS had a mean $\bar{x}_{TSS} \approx 0.68 \pm 0.029$, which is significantly better than the best for an LSTM [7], even though the $\bar{x}_{TSS}$ calculation included a wide range of τ values. The validation accuracy on the other hand had a mean $\bar{x}_{val_{acc}} \approx 0.79 \pm 0.009$. The standard deviation of the validation accuracy, $\sigma_{val_{acc}} \approx 3.18x\ \sigma_{TSS}$, which indicates that the variability of the validation accuracy as $\tau \to 3$ was smaller compared to that of the TSS. If τ is being internally moderated by the model, then this means that changing τ would appear to be insignificant yet the TSS gives an opposing result. The TSS appears to be more sensitive to changes in τ.

It can be argued that the use of the SA to optimize τ is not always possible or applicable. A possible argument is that in a real-world setting, the production test data will only be accessible upon the deployment and only a validation set can be used to optimize τ. Based on the implications of the sometimes absent proportionality of the validation set model metrics and test set model metrics, there is no global best τ value that can be found as each will not always be relevant. Other similar arguments can be that based on the solar cycle dependence nature of the data [2], in a different solar cycle, even the effective range of τ values may adjust based on fluctuations in the data. Hence this proposed a framework for producing a well-calibrated model not a fixed solution for τ. In the case that the SA is not used, a regularization term can be added to the model's loss function to regulate the τ value during training so that the model decides for itself. Even with that alternative approach, the aforementioned disadvantages still hold.

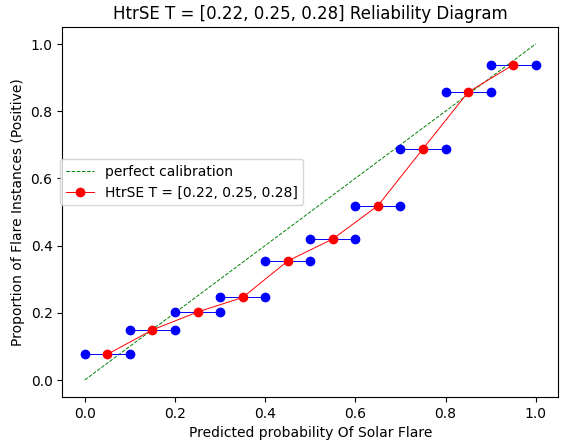

Fig. 4.18: HtrSE Calibration Plot $\tau = 0.25 \pm 0.03$

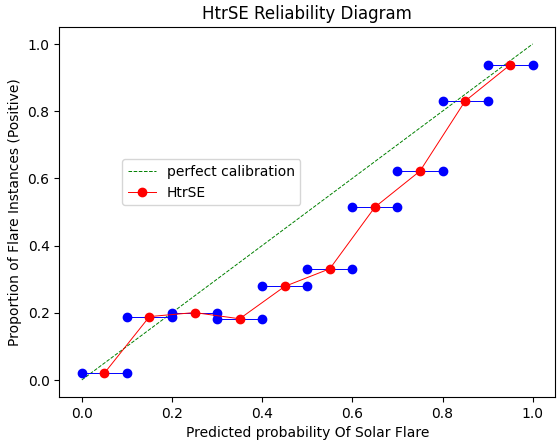

Fig. 4.19: HtrSE Calibration Plot (Not Using GS)

There are multiple reasons why the ensemble whose calibration plot is shown in Figure 4.18 is better calibrated than its competitor with the plot n Figure 4.19. Figure 4.18 is the average or common

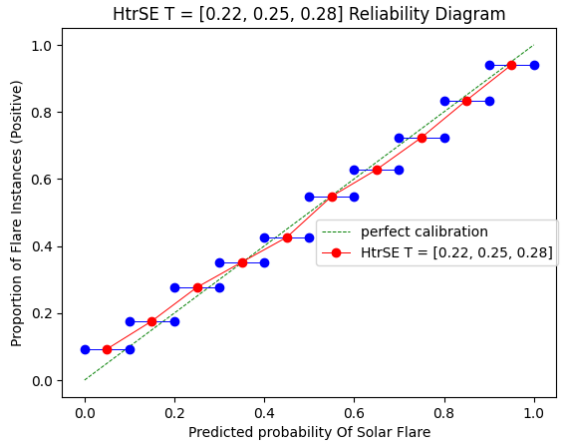

Fig. 4.20: Near Perfect Calibration $\tau = 0.25 \pm 0.03$

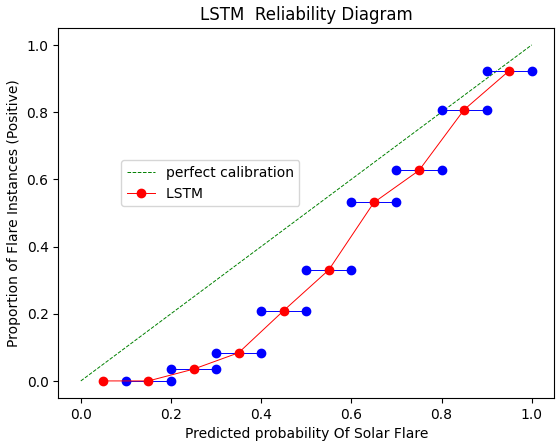

Fig. 4.21: Common LSTM Calibration (Not Using GS)

Fig. 4.22: The Best vs. Worst Average Calibrations

calibration plot generated when the model is trained and tested. The HtrSE τ= [0.22,0.25,0.28] means that $\tau_{LSTM} = 0.22, \tau_{GRU} = 0.25$, and $\tau_{SimpleRNN} = 0.28$, where the LSTM, GRU, and SimpleRNN are the base learners of the HtrSE. The best case can be seen in Figure 4.20. The use of the multiple values for τ in the configuration of the base learners increases the search space of each base learner and hence the meta-model. So basically, the ensemble exploits the search space which is a combination of the search spaces of the base learners as it learns how best to combine them. The combined search space for the base learners that use the Gumbel-Softmax is greater than the search space of the ensemble whose base learners do not use the Gumbel-Softmax. Using a larger search space helps the model get a wider comprehension of the stochastic nature of the solar flare environment. Finally, the meta-model is more robust as reflected in the calibration plot. It is critical to use values of τ that allow the base learners to have good performances individually, preferably such that each base learner's average performance on that τ value is better than that of a base learner not using the GS. In the setup of the ensemble, τ takes on values such that $\tau = 0.25 \pm 0.03$. large differences($\approx$ 0.2) in the τ values across base learners cause the meta-model to have poor performances. This is partly because the ensemble struggles to combine the significantly varied randomness from each base learner's predictions in the training in that case or it would need a larger dataset to perform well. Overall, careful injection of randomness in base learners is one of the main causes of improved calibration.

The use of a Gumbel-Softmax introduces an element of randomness in the model's learning. While this comes as an advantage, it should be used with caution. If the base learners are configured with larger τ values, the ensemble's calibration becomes poorer. Even if the larger τ values are the best for the individual models, i.e., $\tau = 0.5 \pm 0.23$. The chosen base learner values of $\tau = 0.25 \pm 0.03$

(uses the lowest range in $\tau = 0.5 \pm 0.23$, which is best for each individual model) had a good consistency of the heterogeneous stacking ensemble's performance whose base learners used the τ values. In the best-calibrated ensemble, there were some improvements in comparison to the one whose base learners did not use the GS. The BACC improved from ≈ 0.81 to ≈ 0.83. The TSS improved from ≈ 0.63 to ≈ 0.64. The HSS2 improved from ≈ 0.38 to ≈ 0.46 . This shows a significant improvement in the forecasting ability. The f1-score improved ≈ 0.78 to ≈ 0.80 . The precision & recall each improved from ≈ 0.81 to ≈ 0.83. Part of the consistency of the results of the ensemble using base learners configured with the Gumbel-Softmax is displayed in Table 4.4. The relationship of the mean ($\bar{x}$) and the measures of dispersion namely standard deviation (σ) and variance (Var) shows that for the range of $\tau = 0.25 \pm 0.03$ (across three base learners), the model's performance is almost consistent. In conclusion, it is important when using this framework to take into consideration its consistency in producing better results than its alternative, and the fact that larger τ values introduce more randomness and hence inconsistency. The calibration is not always the same for multiple tests. On average, the calibration is better than that of the alternative where base learners do not use the Gumbel-Softmax. The good thing is that the model can at times be near perfect calibration, one instance of this is shown in Figure 4.20. The variation in calibration comes with no surprise as the U_i random variable used by the Gumbel-Softmax (see Equation 3.1) injects the randomness in a non-uniform way.

Train & Test Iteration	BACC	TSS	HSS2	f1-Score	Precision	Recall
1	0.83	0.65	0.47	0.82	0.83	0.83
2	0.83	0.63	0.47	0.81	0.83	0.83
3	0.82	0.65	0.46	0.82	0.82	0.83
4	0.82	0.65	0.45	0.81	0.82	0.82
5	0.83	0.66	0.46	0.81	0.83	0.83
$\bar{x}$	0.83	0.65	0.46	0.81	0.83	0.83
σ	4.9x10^{-3}	9.8x10^{-3}	7.5x10^{-3}	4.9x10^{-3}	4.9x10^{-3}	4.0x10^{-3}
Var	2.4x10^{-5}	9.6x10^{-5}	5.6x10^{-5}	2.4x10^{-5}	2.4x10^{-5}	1.6x10^{-5}

Table 4.4: HtrSE Metrics Consistency Examination Table

It is of interest to examine the differences in the results from the case when the GS was not used in models in contrast to when it was used. Table 4.5 shows the results of the RNN models and their ensembles. HVE, SVE, and HtrSE stand for hard-voting, soft-voting, and heterogeneous stacking ensemble respectively. All models that used the Gumbel-Softmax used $\tau = 0.22$. The LSTM's TSS

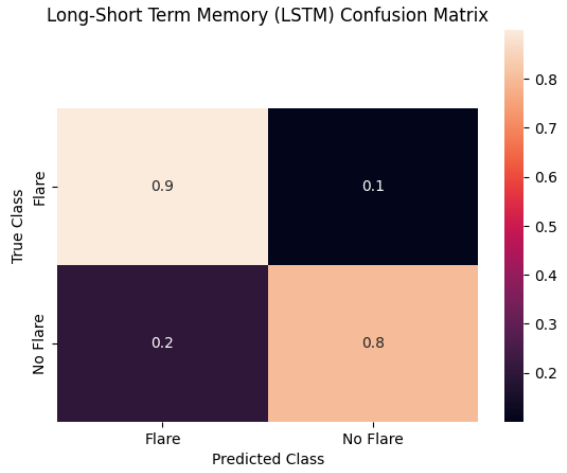

Fig. 4.23: LSTM Calibration (Not Using GS)

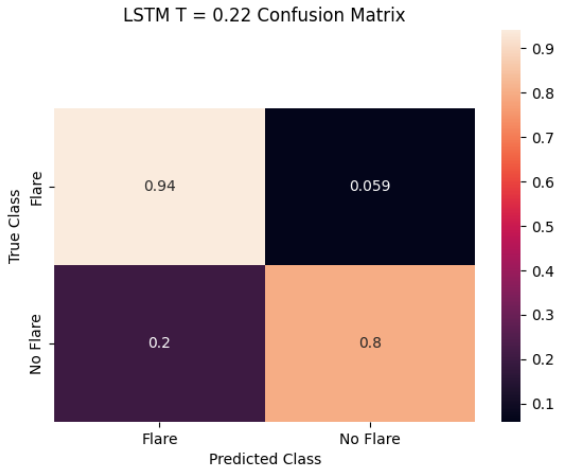

Fig. 4.24: LSTM Calibration Using $\tau = 0.22$

Fig. 4.25: The Best vs. Worst Average Calibrations

Table 4.5: RNN model Performances when using $\tau = 0.25 \pm 0.03$

Model	TSS	BACC	f1-score	HSS2	Precision	Recall
LSTM	0.72	0.86	0.77	0.36	0.84	0.81
GRU	0.71	0.85	0.77	0.36	0.83	0.81
Simple RNN	0.66	0.85	0.78	0.39	0.82	0.81
HVE	0.72	0.86	0.77	0.36	0.83	0.81
SVE	0.69	0.83	0.78	0.38	0.83	0.81
HtrSE	0.65	0.83	0.81	0.46	0.83	0.83

score improved from 70% to 72%. This configuration also reduces the false positive rate (FPR) for the LSTM from ≈ 0.1 to ≈ 0.06 (see Figure 4.25). This means that the model becomes more likely to correctly forecast $\geq C$ positive solar flare instances. However, the false negative rate (FNR) does not seem to change much in all the individual models, either using the Gumbel-Softmax or not, it remains at ≈ 0.2. This value is significant. In real applications, the threshold for the forecasting probabilities can be adjusted to strike a balance on these metrics. A model that showed a decline in its TSS is the SimpleRNN model. When the Simple RNN was not using this framework, its TSS was ≈ 0.7 and its HSS2 was ≈ 0.43. When configured with the GS using $\tau = 0.25 \pm 0.03$, its TSS decreased on average to ≈ 0.67, and its HSS2 to ≈ 0.39. While the LSTM and GRU also had their HSS2 decrease when using this framework, their TSS increased from ≈ 0.7 to ≈ 0.72 and ≈ 0.67 to ≈ 0.71 respectively. The Simple RNN is the only model whose predictive capability declined. It can be argued that the deliberately injected noise in all training iterations led to a failure of the model's proper weight

learning. It is critical to note that if the SimpleRNN and GRU are not using the GS, the Simple RNN performs better than a GRU. This is because in that case, it can reduce the weight of the noisy features during training and it has fewer parameters to optimize due to its simpler architecture (in contrast to the GRU), but in this case, the Gumbel distributed noise may affect back-propagation and the intensity of weight penalties. In conclusion, the more complex models (LSTM and GRU) improved performance when the GS was used whereas the simpler model (Simple RNN) had its performance decline. In fact, the degree of improvement corresponded with the degree of complexity for each model.

Poor calibration does not mean a model will always make poor predictions in contrast to a model that is well-calibrated. While the LSTM model has the best prediction performance, the model with the best calibration is the HtrSE (see Figure 4.20). The differences show that as much as the LSTM is good at recurrence prediction, it has a poor realization of the uncertainty associated with its predictions. The TSS of the LSTM is $\approx 8\%$ greater than that of the HtrSE. The calibration curves of the LSTM and the HtrSE show a significant calibration difference between the two. What then becomes a matter of concern is which model is getting more true positives (TP) correctly, which in this case is the LSTM. It can be argued that with fairly large datasets, the HtrSE may outperform the LSTM. Even though the common diagnostics cannot be used to argue that the LSTM is over-fitting in a way, that does not rule out the possibility of some form of over-fitting.

Voting ensembles can be preferred for different reasons based on the context. The results show that the hard-voting ensemble (HVE) slightly outperforms the soft-voting ensemble (SVE). Both models use the LSTM, GRU, and Simple RNN as base learners. The HVE and SVE TSS scores were ≈ 0.72 and ≈ 0.69 respectively. The HVE does not take into consideration the probability distribution of the binary outcomes, it only considers the actual prediction, i.e. whether a given input X is associated with a ≥C flare or not. The SVE considers the actual predicted binary categorical probability values. Due to poor calibration of the individual models as seen in the LSTM instance in Figure 4.22, the SVE suffered from the uncertainty of its base learners. The HVE ensemble does not consider the base learner's certainty.

3.1 Context of Relevance and Limitations of the Proposed Framework

This study keeps using the stochastic property of the solar flare environment to justify the relevance of its methods, especially the reparameterization trick. In other cases, calibration can be improved by simply using random stratified sampling on the data. The argument is that if the environment is really a stochastic one, then the data should reveal that property without requiring manipulation. The challenge with that approach is that it needs samples that actually do adequately reveal the stochastic

nature of the environment where the quantity of samples allows the model to learn adequately. It requires larger datasets. The chronological split of data for solar flare predictions somehow challenges this aspect. The chronological split has been used and justified in many studies [7, 17, 21, 38]. It helps avoid the use of data from the same active regions. The relevance of the application and advantage of this framework is that even with smaller datasets, it can simulate the stochastic environment itself while learning through the randomness injected into the predictions. So in the case of small datasets that do not fully represent the stochastic nature of the environment, the framework is recommended.

4 Line of Sight Magnetogram Preprocessing

In this section, the results from the original magnetograms are compared with the clopened full magnetograms. Thereafter, the two are compared against the clopened magnetograms where a region of interest has been cropped out. Figures 4.26, 4.29, 4.28, 4.30, 4.27 show the confusion matrix, calibration curve, precision-recall curves, the Receiver Operating Characteristics (ROC) curve, and a combination of the training curves (AUC, validation loss, validation AUC, loss) obtained from the Convolutional LSTM on original magnetograms. When one looks at Figure 4.27 and Figure 4.31, it is evident that the learning process is smoother for clopened magnetograms when compared to the original magnetograms. The smoothness increases when the clopened magnetogram's region of interest (ROI) is used (see Figure 4.34). This means that the removal of the noisy pixels improves the learning process. With the results from the clopened ROIs (see Figure 4.34), it means that focusing on the largest darkish or whitish regions in the magnetogram further improves the learning process.

On the other hand, the AUC curve in Figure 4.27's validation curve goes higher than 0.84 whereas the one in Figure 4.31's validation curve is lower. This means that the ConvLSTM had a lower degree of separability between clopened magnetograms when compared to the original magnetograms (during training, this also holds for the test phase). When the ROI is used, the degree of separability (AUC) appears to return to normal (during training) but significantly drops in the test phase to 0.82. The results support hypothesis H_4 formulated in Section 1.4. These findings however reveal that the detail in the original magnetograms is useful in solar flare predictions (denoising reduces predictability). It also reveals that the area surrounding the largest sunspots is also useful in predicting solar flares (cropping reduces predictability). Looking into other metrics, the original magnetograms yielded a TSS score of 0.63, and a BACC score of 0.82. The HSS was also on average above 0.55. On the other hand, the clopened magnetograms yielded a TSS score of 0.62, and a BACC score of 0.8. The HSS was also on average above 0.5. Looking at the other metrics, it is clear that the original magnetograms are better as a source of features when compared to clopened magnetograms. However, the results

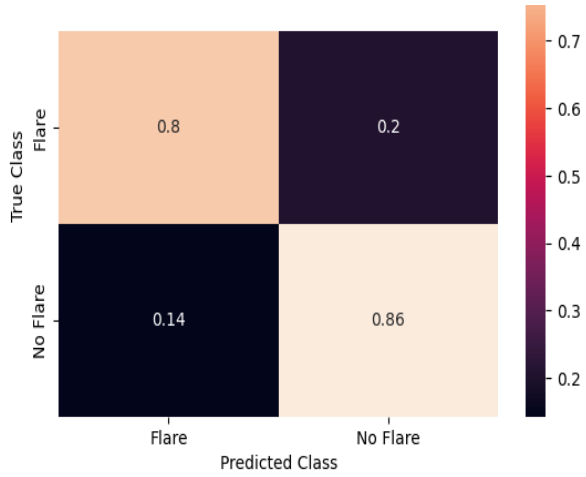

Fig. 4.26: ConvLSTM Confusion Matrix (Original)

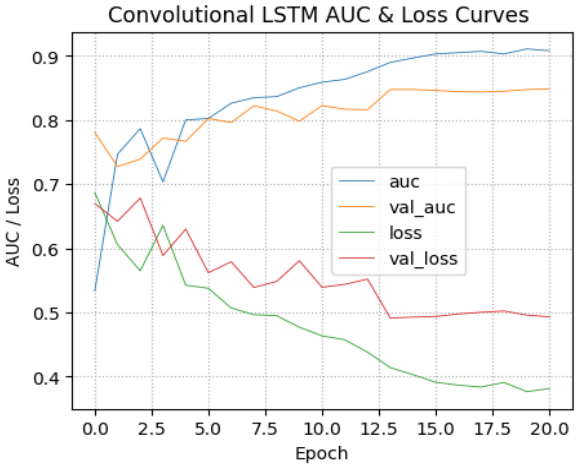

Fig. 4.27: ConvLSTM Training Graphs (Original)

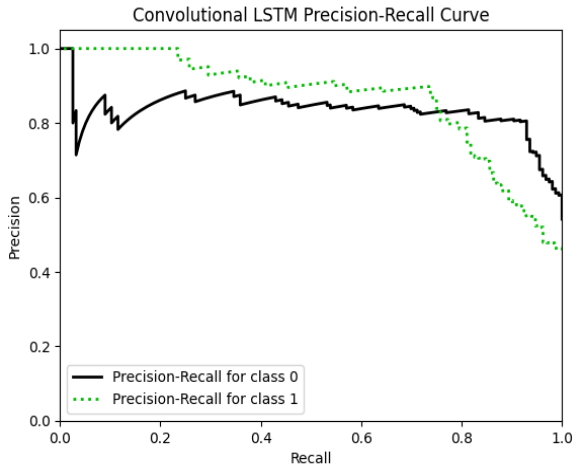

Fig. 4.28: ConvLSTM PR Curves (Original)

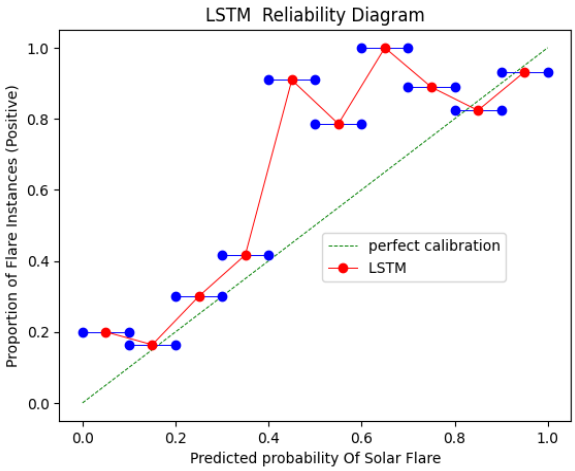

Fig. 4.29: ConvLSTM Calibration Curve (Original)

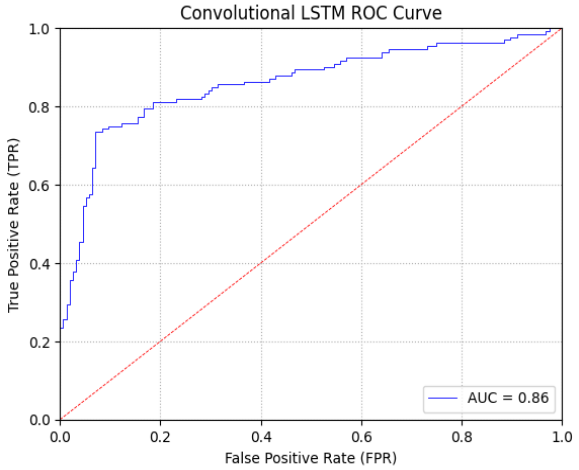

Fig. 4.30: ConvLSTM ROC Curve (Original)

have to be taken with a pinch of salt. The structuring elements used in the clopening process may be the ones that need adjusting. This would typically require iterative setting and testing or other

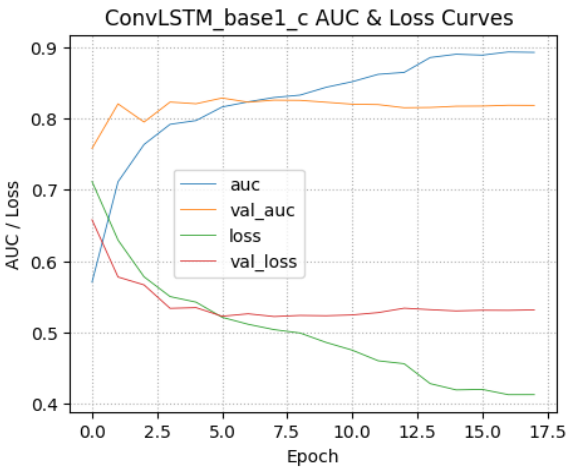

Fig. 4.31: ConvLSTM Training Graphs (Clopen)

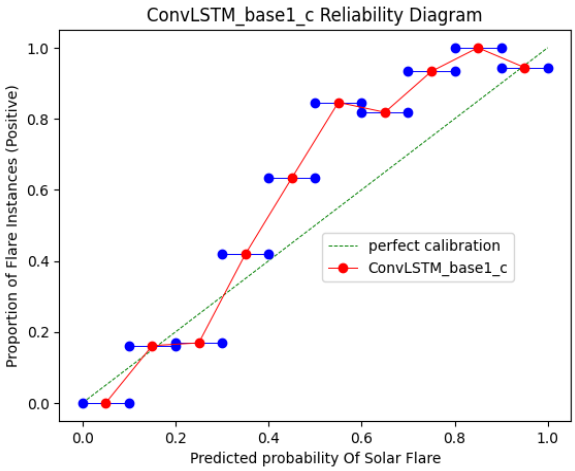

Fig. 4.32: ConvLSTM Calibration Curve (Clopen)

Fig. 4.33: Some Results From Clopened Full Magnetograms (Clopen)

advanced methods.

One of the central themes of this dissertation is the comparison of ensembles with their base learners. For the Convolutional LSTM, a weighted soft-voting convolutional LSTM (ConvLSTM SVE) ensemble model was implemented. Only one base learner was used, the only Convolutional LSTM described earlier. What differed amongst the base learners was the data used (for training, validation, and testing). There were three base learners used. The first one used the original magnetograms. The second one used the clopened magnetograms. The third one used the clopened magnetograms where the region of interest was cropped out. The ensemble is a feature fusion ensemble since it uses various data sources (not just different partitions). Based on the results obtained from the individual models, the class probability weights were configured as 2,1,1 for the positive class ($\geq C$) for the three base learners respectively. The model with the weight 2 was the one using the original magnetograms due to its higher performance on the positive class. Increasing the weight to 3 or decreasing it to 1 declines performance. The negative class ($< C$) weights were left at default (equal contribution from the models). The ConvLSTM SVE obtained an HSS score of 0.63, a TSS score of 0.67, and a BACC score of 0.84. The metrics collectively reflect improved predictive power in contrast to the individual models each using a different version of the data. The ConvLSTM SVE obtained an AUC of 0.88 (see Figure 4.38), which also reflects improved discriminating ability for data instances that lead to $\geq C$ or $< C$ solar flares. Figure 4.37 shows the calibration curve obtained. Comparing figures 4.37 and 4.36 shows slight but almost negligible differences in the calibration. Experimenting with removing one data source shows that the combined use of the data sources leads to a better feature fusion model in contrast to a pair or individual. Overall, the results echo the

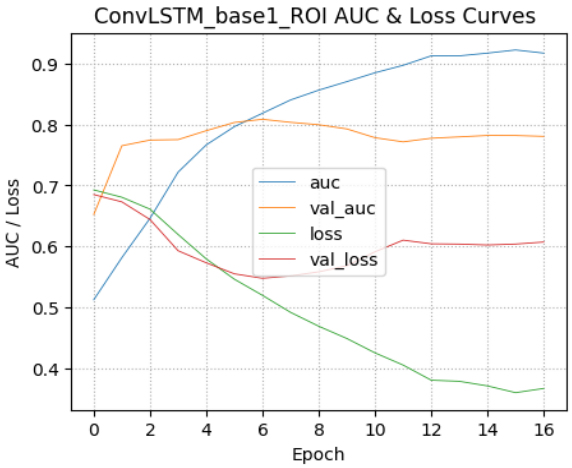

Fig. 4.34: ConvLSTM Training Graphs (Clopen, ROI)

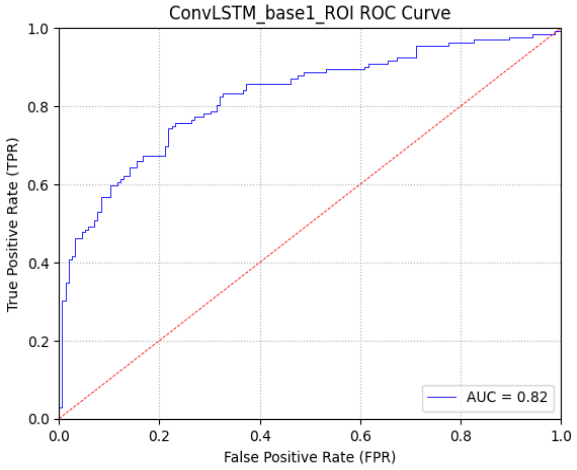

Fig. 4.35: ConvLSTM ROC (Clopen, ROI)

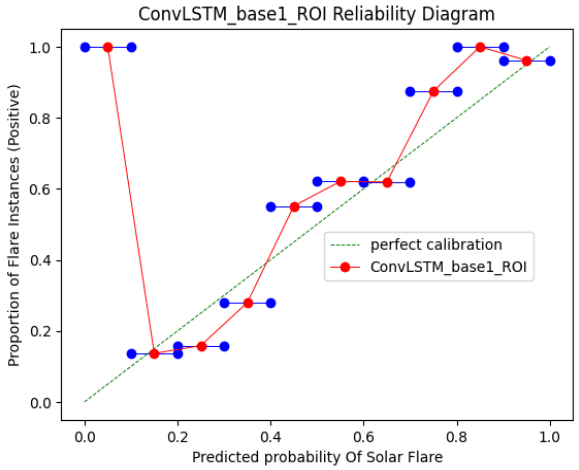

Fig. 4.36: ConvLSTM Calibration Curve (Clopen, ROI)

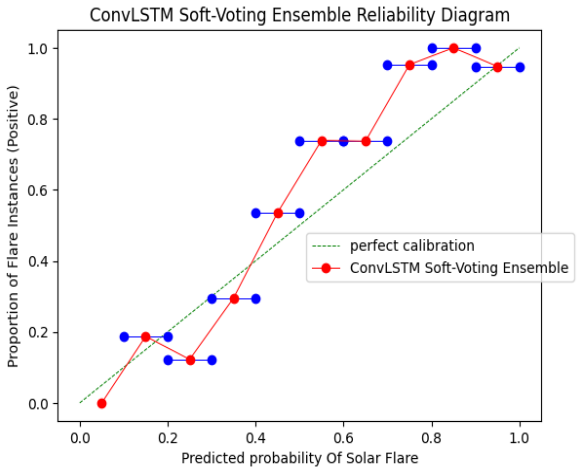

Fig. 4.37: ConvLSTM SVE Calibration Curve (Fused Features)

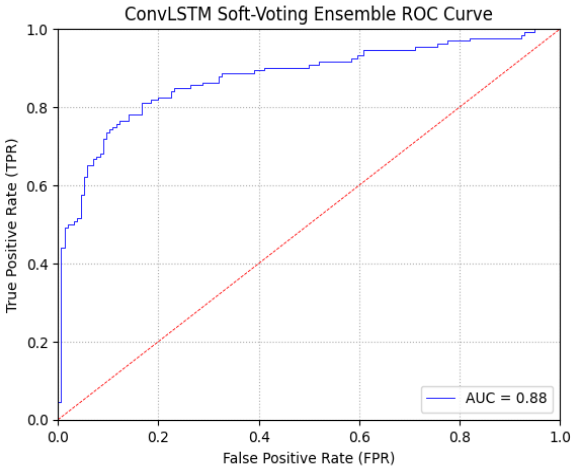

Fig. 4.38: ConvLSTM SVE ROC Curve (Fused Features)

hypothesized findings based on the review by [48] and the findings on solar flare recurrence prediction by [95].

This study employs a methodology that prevents it from being fairly comparable with other studies. Differences include the pipeline (denoising and region of interest computational techniques), sequence sizes, and data selection/elimination criteria, duration in years of the data used etc. Some studies have a mask for missing values e.g., [7], whereas this study excludes the entire sequence.

5 Ensemble CNNs for ≥C Solar Flare Recognition At Wavelength 1,600$\mathring{A}$

5.1 Performance of Individual Models

The configurations meant to introduce variety in the CNN models are reflected at the training stage. The shapes and smoothness of the graphs in Figure 4.39 show the variations among models. To be presentable, the training and validation losses were squashed from the range $[0, \infty)$ to $(0, 1)$ using the sigmoid (normally used as an activation function), defined as $\sigma(x) = \frac{1}{1+e^{-x}} = \frac{e^x}{e^x+1}$. This comes from the fact that $\forall x, x \in R, \sigma(x) \in (0, 1)$. What is more important to observe is the trend/shape of the graphs over time than their original values, for this study. It should be noted however that the diversity that matters is the one which the models can show when making predictions [48].

The various configurations and architecture affect the training time of the models and cause variation. The training time descending order is NASNetLargeNetLarge, XceptionNet, ResNet50, AlexNet, and LeNet5. The AlexNet and LeNet5 CNNs not only required less time, but they also performed well.

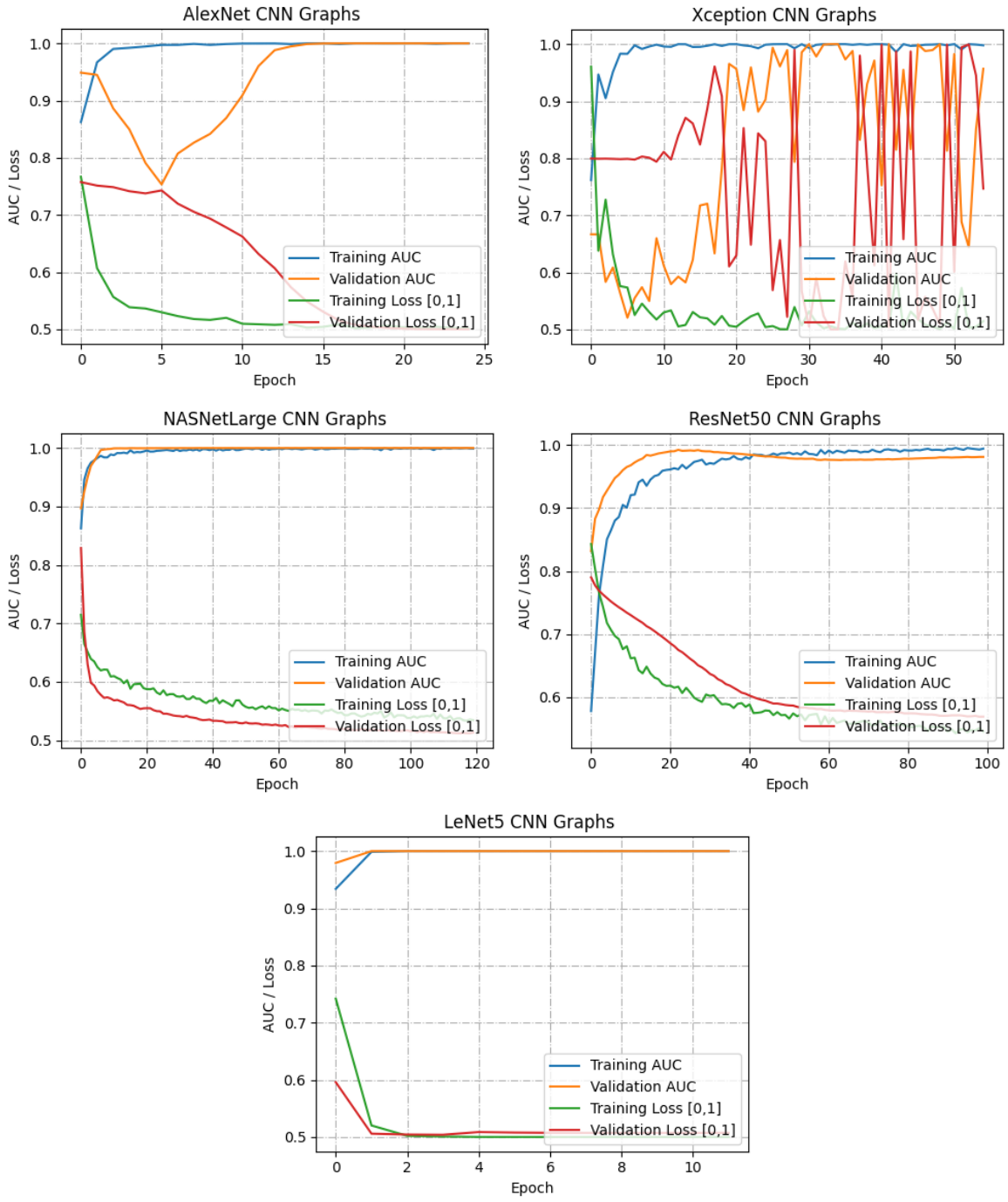

Fig. 4.39: AUC & Loss Curves for CNN Models

In Figure 4.39 it can be noted that for the NASNetLarge model, the validation loss curve is below the training loss curve. This is due to the effect of the sigmoid applied on the loss curve to constrain it in the range $[0, 1]$ for the sake of being presentable (i.e., force the curves to neatly fit in a small graph while preserving the trends in the curves). In cases where training time is critical yet one model will be insufficient, the AlexNet and LeNet5 would be top candidates to be used. This also applies to models trained in parallel.

Figure 4.40 shows that all models on average did well in recognizing the Limb & Two Ribbon (labeled as '2 Ribbon') class. With a common misclassification of the Compact class. The simplest class to recognize is the Background, the confusion matrices show that there were no false positives in that class. As the average accuracy shows, the AlexNet model average had more true positives, with some false positives of the Limb class when the true positives would have been Compact. As evidenced on the confusion matrices, LeNet5 and XceptionNet misclassified the Compact and Background events. What may be interesting is that for each of those misclassifications, the models had false positives on different classes. Overall, the false alarm rates (FAR) were low, meaning the models performed well and could be usable in a real-world setting. Table 4.6 shows results from individual models.

5.2 Ensemble CNNs & Their Performance

The choice of different CNN architectures was motivated by the prerequisite for ensemble success, which is diversity. In this experiment, it was observed that the error correlation of the models was low. In this study, more than two classifiers were used, making diversity quantification a slightly challenging task as stated in [60].The Soft-Voting ensemble is mathematically defined as: $\hat{y} = argmax_j \sum_{i=1}^{n} w_i p_{i,j}$ where $p_{i,j}$ is the predicted class of the i^{th} classifier for the j^{th} class label. The term w_i,$\forall w_i \in \{w_1, w_2, ..., w_n\}$ is a weight parameter that is optional, it's default value is $\frac{1}{n}$ where the weights are equally shared. In this study, the default weight was used. To define the Hard-Voting/Majority Voting ensemble, let $S = \{m_1, m_2, m_3, ..., m_n\}$ denote a set of n models. Then if x is one of the test data items and $p(m_i(x))$ is the prediction of model i given on x, the ensemble vote on x's classification is defined as:
$\hat{z} = mode\{p(m_1(x)), p(m_2(x)), p(m_3(x)), ..., p(m_n(x))\}$.

Both the Hard Voting and Soft Voting ensemble did very well obtaining a 100% accuracy in the final tests on average. A tabulated summary of previous works can be seen in Table 4.7. The findings in this study support the idea brought forward by [50] where it is hypothesized that ensemble CNNs are more likely to do better than their best base learner.

The following observation was noted when conducting this study. The overall conclusion is that error

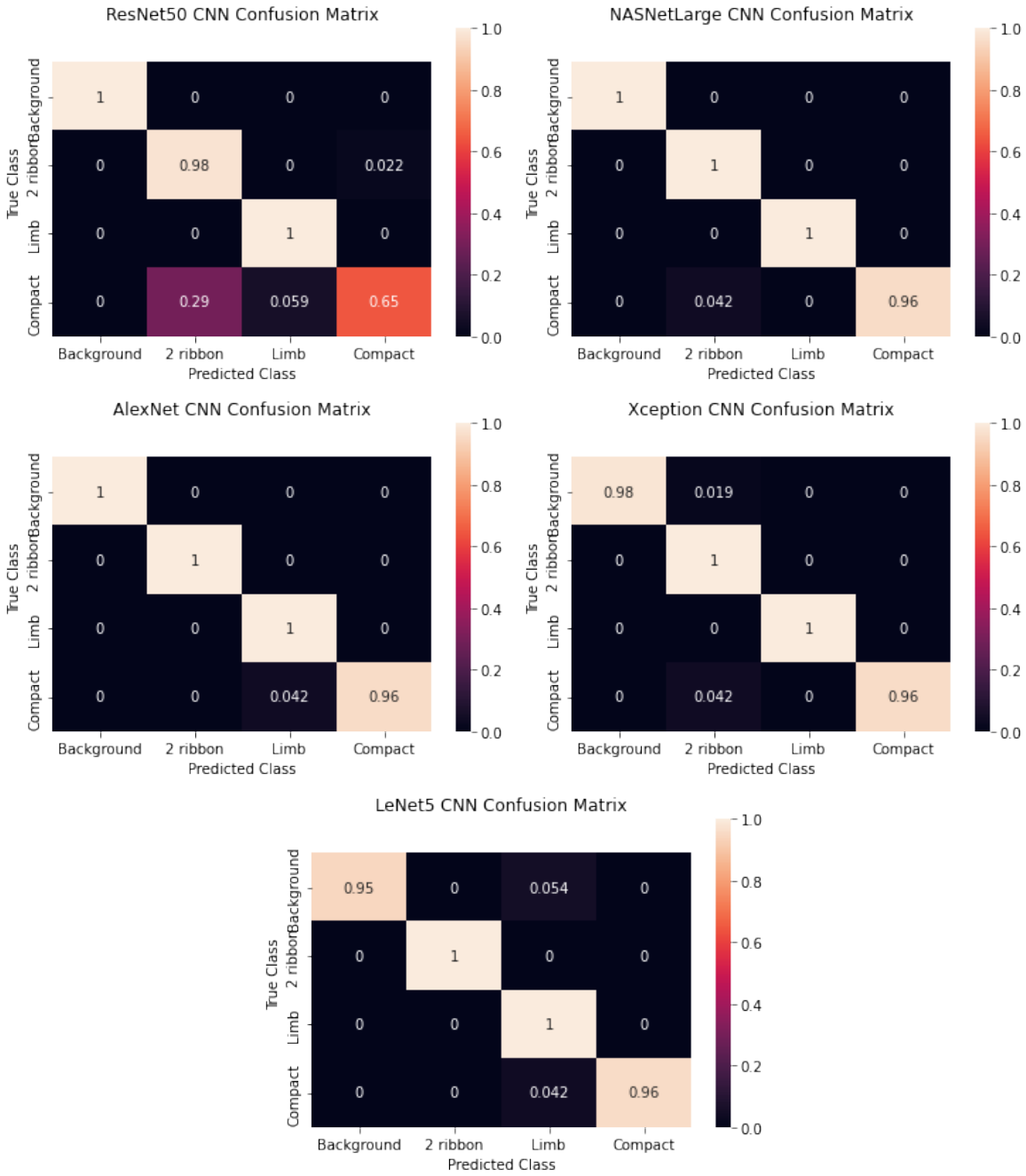

Fig. 4.40: Sample Confusion Matrices for CNN Models

Table 4.6: Individual CNN Model Performance Summary

CNN	Accuracy	Specificity	Sensitivity	AUC	f1-score
AlexNet	0.99 ± 0.01	0.99 ± 0.01	0.98 ± 0.02	0.99 ± 0.01	0.99 ± 0.01
XceptionNet	0.98 ± 0.01	0.98 ± 0.01	0.96 ± 0.03	0.97 ± 0.02	0.94 ± 0.01
LeNet5	0.99 ± 0.01	0.99 ± 0.01	0.98 ± 0.02	0.99 ± 0.01	0.99 ± 0.01
NASNetLarge	0.98 ± 0.01	0.99 ± 0.00	0.98 ± 0.01	0.99 ± 0.00	0.99 ± 0.00
ResNet50	0.92 ± 0.01	0.96 ± 0.01	0.90 ± 0.01	0.99 ± 0.00	0.9 ± 0.02

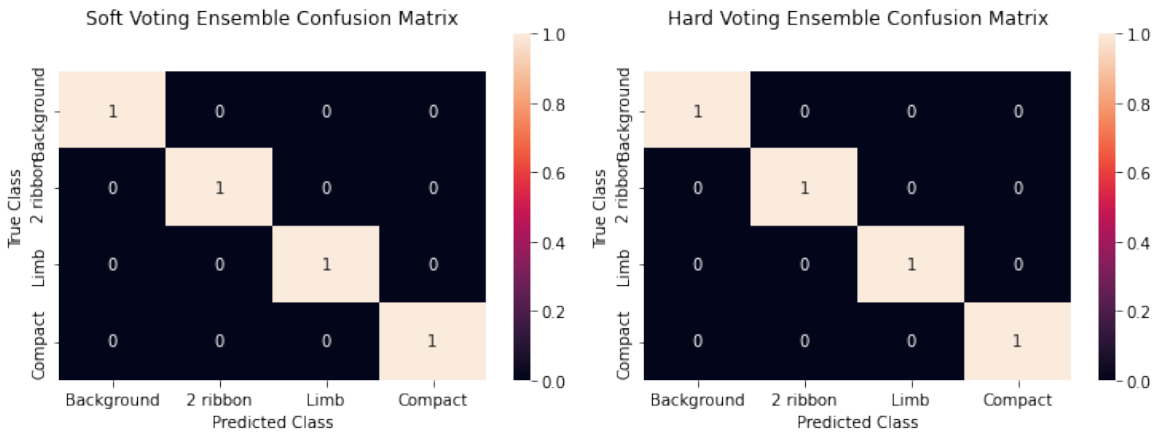

Fig. 4.41: Confusion Matrices for Soft and Hard Voting Ensemble

Table 4.7: Related Works (On ≥C Flares Captured At Wavelength 1,600$\mathring{A}$)

Paper Reference	Background	Compact	Two Ribbon	Limb
Love et al., 2020 (Individual CNN) [18]	95%	88%	97%	92%
This study(Ensemble CNNs)	100%	100%	100%	100%

correlation need not always be low as most studies claim, it can be high at times and have no effect on the ensemble's choices. Let P_m as $P_m(x) = \{m_i p(x), m_{i+1} p(x), m_{i+2} p(x), ..., m_n p(x)\}$ be a set of predictions on input x by n models, where $n \in Z$, such that $m_i p(x)$ is the i^{th} model's classification prediction on input x. In this experiment, $n = 5$, the five models were AlexNet, LeNet5, ResNet50, NASNetLarge, and XceptionNet. Let's also define $W_m(x)P_m(x)$ where $W_m(x)$ is the set of wrong classifiers on predicting x's class. In this study, it was noted that if $|W_m(x)| < |P_m(x) \setminus W_m(x)|$ then the error correlation of models in $W_m(x)$ will have no impact on the ensemble's final correct classification, even if the models in $W_m(x)$ are fully correlated in their predictions. This fact applies to the hard-voting ensembles. For the aforementioned fact to be true in soft voting ensembles, it is required that the sum of probabilities for the correct class in the models in $P_m(x) \setminus W_m(x)$ must be greater than the sum of probabilities of any class in $W_m(x)$.

Initially, the null hypothesis H_6 was that there would be no difference between an ensemble and the best individual CNN. A McNemar's statistical significance test [96] was used to test H_6. To avoid division by 0, the prediction count values were increased by 1. This follows after the ensemble had 100% accuracy. The results show that the Chi-square statistic is ≈ 13.482 and p-value $\approx 2.440\text{x}10^{-4}$, using $\alpha = 0.05$. The statistical significance implies that H_0^4 is rejected.

6 Conclusion

This chapter presented and discussed the results obtained in this study. The next chapter concludes this study drawing insights from the findings from this chapter.

Chapter 5: Conclusion

1 Conclusion Observations

1.1 Individual RNN vs Ensemble RNN Performances

The experimental results of this study show that in general, ensemble RNNs are better calibrated in comparison to individual RNN models (see the HtrSE and LSTM calibration curves in Figures 4.20 and 4.25, respectively. This further supports existing literature on the claim that ensembles have a better generalization than individual models as stated in [48, 95]. The common difference in ensembles predicting the recurrence of $\geq$C flares in contrast to individual models is that the individual models tend to have a higher TSS, as seen in this study with HtrSE vs. LSTM (see Tables 4.2 and 4.5, 1^{st} and 6^{th} rows respectively in both tables). This relationship has also been seen in [7] (the Random Forest vs. the LSTM). So, the individual models have a poor awareness of the uncertainty associated with their predictions, but they have higher performance scores. The observation can be used to argue a possible form of over-fitting. The reason why the models are not diagnosed with over-fitting in this study is because their validation accuracy is on average lower than their test accuracy. Recalling that $\geq$C flares include, $\{C, M, X\}$ and $<$C include $\{A, B\}$ class flares and their 10 variations each, this study argues that their distributions across the chronological splits are the reason for the differences in validation and test set performances due to their unequal distribution in the training, validation, and test sets. The total number of sub-classes the model had to deal with is 50. In conclusion, the calibration results support hypothesis H_0^1 that was conjectured in Section 1.4.

1.2 Effectiveness of Reparameterization in Reliability

The main contribution of the reparameterization using the Gumbel distribution in the binary categorical predictions was an increased search space for base learners (LSTM, GRU, SimpleRNN). That improved the calibration of the Heterogenous Stacking Ensemble (HtrSE) as seen in even better generalization. In general, the search space of the ensemble is a combination of the search spaces of

its base learners [48]. In the absence of reparameterization, the base learners try to learn a nearly deterministic function, which has a more constrained solution search space in comparison to that involved in the use of the Gumbel distribution. The $U_i \in (0, 1)$ is a random variable in the Gumbel-Softmax that injects the noise in the predictions and ensures that even in a small sample dataset, the model would not learn a nearly deterministic function and possibly over-fit as small datasets make models prone to over-fitting [97]. The advantage of the Gumbel Softmax(GS) in this case is that the predictions that resemble a Gumbel distribution are used in back-propagation and affect the way the weights (e.g., $W_{fh}, W_{fx}, W_{ox}, W_{\hat{c}h}$ e.t.c, for the LSTN) are learned. If context allows for τ to be kept constant like in this study, then $\tau \in (0, 1)$ is preferable in contrast to $\tau \in (1, 10)$ since $\tau \to 0$ implies that the GS's binary categorical distribution of predictions on $\geq$C and $<$C flares have less noise hence the model will converge. Findings show that the best τ for an individual model may not be the ideal one for an ensemble where the same model is a base learner. Sometimes, larger τ values on the base learners can cause a delayed convergence of the meta-learner and that is a serious problem if the dataset is small. The base learners may not have enough data to anneal τ to be optimal and so the meta-model (if internal annealing is configured). So at times, a subjective choice for τ is conditionally justifiable like in this study. In conclusion, the findings support H_0^2 formulated in Section 1.4.

1.3 On the Preprocessing of Line of Sight Magnetograms

In this study, a Convolutional Long-Short Term Memory model was used to examine the effectiveness of various line-of-sight magnetogram preprocessing approaches. The magnetograms were arranged in sequential formats and used to predict the intensity ($\geq C$ or $<$C) of an upcoming solar flare in a supervised learning approach. In summary, each magnetogram had a thresholded version in which the closing and opening (clopen) operations were applied. This was hypothesized to reduce noise in the magnetogram (see Figure 3.11). The second version was the previously described magnetogram where a region of interest (ROI) was cropped. This was hypothesized to reduce the size of the data and possibly eliminate noise. The third version was the original magnetogram. The results show that the original magnetograms are much better as data sources of learning. The results of the ROC curves and AUC values (including TSS scores) led to a rejection of hypothesis H_0^3 (refer to Figures 4.35 and 4.30). The results show that the details in line-of-sight magnetograms are critical in the sequential relationships of their observations and $\geq$C or $<$C predictions.

1.4 Effectiveness of Ensembles in Visual Recognition At Wavelength 1,600$\overset{\circ}{A}$

This part of the study basically corrected two shortcomings in the methodology that was used by [18]. The first is simple, it is the use of efficient CNN architectures. CNN Models that are too large or too small will not produce the best results. This study also showed how the use of base learner diversity techniques suggested in [48] can be used effectively to improve ensemble success. The motivation to choose the better architectures was given in [50]. Although the study in [50] was based on solar events in general, it suggested the use of ensemble CNNs for better performance. This study adopted the suggested approach (ensembles). The results showed an improvement of the classification accuracy from 94% (obtained in [18]) to 100% in this study. The cause of the improvement is the use of suitable architectures and ensembles. The hypotheses test discussed earlier in this study show that the results support hypothesis H_1^4.

2 Conclusive Remarks

This study focused on the recurrence prediction and visual recognition of $\geq$C solar flares. Ensembles were compared to individual models in both recurrence prediction and visual recognition. The results show that ensemble models have a better generalization than individual models. This is reflected by a significant improvement in calibration for the ensemble RNNs and better classification metric scores for the CNNs. Although the individual RNNs obtained higher TSS scores on average, they are significantly less reliable in comparison to their ensembles [95]. In a real-world setting, the ensemble would be preferable for both recurrence prediction and visual recognition.

3 Proposed Future Works

Future studies can explore the efficiency of internal annealing of τ vs using fixed values on stochastic RNNs under the condition that the RNNs will be used in an ensemble RNN model. The main discussions in such a study can be centered on how that affects the calibration of the resulting ensemble models. The results are likely to have a rich literature with broad applications. For the magnetogram preprocessing approaches, it would be worthwhile exploring various combinations of the structuring elements involved in the closing and opening operations. Although the results obtained fail to support the hypothesized outcome, it does not rule out the fact that it is possible that with optimized combinations of structuring elements, the results can be along the lines of the hypothesized outcome when predicting the recurrence of $\geq$ C solar flares from the line of sight magnetograms when using the Convolutional LSTM. In the visual recognition subsection, this study

limits itself to $\geq$ C solar flares observed at wavelength 1,600$\overset{\circ}{A}$. Future works can explore the performance of CNNs vs. CNN ensembles across various solar flare classes e.g., A, B, C, M, X in various Atmospheric Imaging Assembly (AIA) observational wavelengths. Depending on interests, the related magnetograms of each observation can also be used.

Chapter 6: Appendix

A Some Formulae Used as Metrics Of Performance

A.1 Common Acronyms in Formulas

1. TP - True Positives (Positive instances identified as positive)
2. TN - True Negatives (Negative instances identified as negative)
3. FP - False Positives (Non-positive instances identified as positive)
4. FN - False Positives (Non-negative instances identified as negative)

A.2 Measures of Performance

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{6.1}$$

$$\text{Sensitivity} = \frac{TP}{TP + FN} \text{ (True Positive Rate)} \tag{6.2}$$

$$\text{Specificity } = \frac{TN}{TN + FP} \text{ (True Negative Rate)} \tag{6.3}$$

$$\text{Balanced Accuracy (BACC)} = \frac{Sensetivity + Specificity}{2} \tag{6.4}$$

$$\text{f1-score} = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \tag{6.5}$$

$$\text{True Skill Statistic (TSS) } = \frac{TP}{TP + FN} - \frac{FP}{FP + TN} \tag{6.6}$$

$$\text{Heidke Skill Score (HSS)} = \frac{2(TP * TN - FP * FN)}{(TP + FN)(FN + TN) + (TP + FP)(FP + TN)} \tag{6.7}$$

References

[1] Z. Sun, M. G. Bobra, X. Wang, Y. Wang, H. Sun, T. Gombosi, Y. Chen, and A. Hero, "Predicting solar flares using cnn and lstm on two solar cycles of active region data," *The Astrophysical Journal*, vol. 931, no. 2, p. 163, 2022.

[2] X. Wang, Y. Chen, G. Toth, W. B. Manchester, T. I. Gombosi, A. O. Hero, Z. Jiao, H. Sun, M. Jin, and Y. Liu, "Predicting solar flares with machine learning: Investigating solar cycle dependence," *The Astrophysical Journal*, vol. 895, no. 1, p. 3, 2020.

[3] F. Benvenuto, M. Piana, C. Campi, and A. M. Massone, "A hybrid supervised/unsupervised machine learning approach to solar flare prediction," *The Astrophysical Journal*, vol. 853, no. 1, p. 90, 2018.

[4] L. Fletcher, B. R. Dennis, H. S. Hudson, S. Krucker, K. Phillips, A. Veronig, M. Battaglia, L. Bone, A. Caspi, Q. Chen *et al.*, "An observational overview of solar flares," *Space science reviews*, vol. 159, pp. 19–106, 2011.

[5] J. Platts, M. Reale, J. Marsh, and C. Urban, "Solar flare prediction with recurrent neural networks," *The Journal of the Astronautical Sciences*, vol. 69, no. 5, pp. 1421–1440, 2022.

[6] C. Campi, F. Benvenuto, A. M. Massone, D. S. Bloomfield, M. K. Georgoulis, and M. Piana, "Feature ranking of active region source properties in solar flare forecasting and the uncompromised stochasticity of flare occurrence," *The Astrophysical Journal*, vol. 883, no. 2, p. 150, 2019.

[7] H. Liu, C. Liu, J. T. Wang, and H. Wang, "Predicting solar flares using a long short-term memory network," *The Astrophysical Journal*, vol. 877, no. 2, p. 121, 2019.

[8] G. Barnes and K. Leka, "Evaluating the performance of solar flare forecasting methods," *The Astrophysical Journal*, vol. 688, no. 2, p. L107, 2008.

[9] R. Zhang, L. Liu, H. Le, and Y. Chen, "Equatorial ionospheric electrodynamics during solar flares," *Geophysical Research Letters*, vol. 44, no. 10, pp. 4558–4565, 2017.

[10] "Forbes currency converter," https://www.forbes.com/advisor/money-transfer/currency-converter/zar-usd/, accessed on September 23, 2023.

[11] K. Kaneda, Y. Wada, T. Iida, N. Nishizuka, Y. Kubo, and K. Sugiura, "Flare transformer: Solar flare prediction using magnetograms and sunspot physical features," in *Proceedings of the Asian Conference on Computer Vision*, 2022, pp. 1488–1503.

[12] ESA eoPortal, "SDO," https://www.eoportal.org/satellite-missions/sdo, accessed 31 January 2022.

[13] "Solar dynamics observatory artwork," https://www.sciencephoto.com/media/151059/view/solar-dynamics-observatory-artwork, accessed on September 23, 2023.

[14] M. Aktukmak, Z. Sun, M. Bobra, T. Gombosi, W. B. Manchester, Y. Chen, and A. Hero, "Incorporating polar field data for improved solar flare prediction," *arXiv preprint arXiv:2212.01730*, 2022.

[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] A. Raboonik, H. Safari, N. Alipour, and M. S. Wheatland, "Prediction of solar flares using unique signatures of magnetic field images," *The Astrophysical Journal*, vol. 834, no. 1, p. 11, 2016.

[17] 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.

[18] T. Love, T. Neukirch, and C. E. Parnell, "Analyzing aia flare observations using convolutional neural networks," *Frontiers in Astronomy and Space Sciences*, vol. 7, p. 34, 2020.

[19] A. H. Toledo, R. Flikkema, and L. H. Toledo-Pereyra, "Developing the research hypothesis," *Journal of Investigative Surgery*, vol. 24, no. 5, pp. 191–194, 2011.

[20] S. B. Hulley, S. R. Cummings, W. S. Browner, D. G. Grady, N. Hearst, and T. Newman, "Conceiving the research question," *Designing clinical research*, vol. 335, 2001.

[21] N. Nishizuka, K. Sugiura, Y. Kubo, M. Den, and M. Ishii, "Deep flare net (defn) model for solar flare prediction," *The Astrophysical Journal*, vol. 858, no. 2, p. 113, 2018.

[22] M. D. Kazachenko, B. J. Lynch, A. Savcheva, X. Sun, and B. T. Welsch, "Toward improved understanding of magnetic fields participating in solar flares: statistical analysis of magnetic fields within flare ribbons," *The Astrophysical Journal*, vol. 926, no. 1, p. 56, 2022.

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

[24] M. Wheatland, "A bayesian approach to solar flare prediction," *The Astrophysical Journal*, vol. 609, no. 2, p. 1134, 2004.

[25] M. S. Wheatland, "A statistical solar flare forecast method," *Space Weather*, vol. 3, no. 7, 2005.

[26] M. Qu, F. Y. Shih, J. Jing, and H. Wang, "Automatic solar flare detection using mlp, rbf, and svm," *Solar Physics*, vol. 217, pp. 157–172, 2003.

[27] R. Li, H.-N. Wang, H. He, Y.-M. Cui, and Z.-L. Du, "Support vector machine combined with k-nearest neighbors for solar flare forecasting," *Chinese Journal of Astronomy and Astrophysics*, vol. 7, no. 3, p. 441, 2007.

[28] R. Qahwaji and T. Colak, "Automatic short-term solar flare prediction using machine learning and sunspot associations," *Solar Physics*, vol. 241, pp. 195–211, 2007.

[29] P. Bornmann and D. Shaw, "Flare rates and the mcintosh active-region classifications," *Solar physics*, vol. 150, pp. 127–146, 1994.

[30] X. Huang, D. Yu, Q. Hu, H. Wang, and Y. Cui, "Short-term solar flare prediction using predictor teams," *Solar Physics*, vol. 263, pp. 175–184, 2010.

[31] C. Pham, V. Pham, and T. Dang, "Solar flare prediction using two-tier ensemble with deep learning and gradient boosting machine," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 5844–5853.

[32] J. A. Guerra, A. Pulkkinen, and V. M. Uritsky, "Ensemble forecasting of major solar flares: First results," *Space Weather*, vol. 13, no. 10, pp. 626–642, 2015.

[33] J. A. Guerra, S. A. Murray, D. S. Bloomfield, and P. T. Gallagher, "Ensemble forecasting of major solar flares: methods for combining models," *Journal of Space Weather and Space Climate*, vol. 10, p. 38, 2020.

[34] F. Ribeiro and A. L. S. Gradvohl, "Machine learning techniques applied to solar flares forecasting," *Astronomy and Computing*, vol. 35, p. 100468, 2021.

[35] E. Wuest, "Solar flare detection from solar magnetogram images using convolutional and recurrent neural networks," Ph.D. dissertation, New Mexico State University, 2021.

[36] M. Hagyard, J. Smith, D. Teuber, and E. West, "A quantitative study relating observed shear in photospheric magnetic fields to repeated flaring," *Solar physics*, vol. 91, pp. 115–126, 1984.

[37] M. Kull, M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, and P. Flach, "Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration," *Advances in neural information processing systems*, vol. 32, 2019.

[38] N. Nishizuka, Y. Kubo, K. Sugiura, M. Den, and M. Ishii, "Reliable probability forecast of solar flares: deep flare net-reliable (defn-r)," *The Astrophysical Journal*, vol. 899, no. 2, p. 150, 2020.

[39] W. D. Pesnell, B. J. Thompson, and P. Chamberlin, *The solar dynamics observatory (SDO)*. Springer, 2012.

[40] J. R. Lemen, A. M. Title, D. J. Akin, P. F. Boerner, C. Chou, J. F. Drake, D. W. Duncan, C. G. Edwards, F. M. Friedlaender, G. F. Heyman *et al.*, "The atmospheric imaging assembly (aia) on the solar dynamics observatory (sdo)," *Solar Physics*, vol. 275, pp. 17–40, 2012.

[41] M. G. Bobra, W. T. Barnes, T. Y. Chen, M. Cheung, L. A. Hayes, J. Ireland, M. Janvier, M. S. Kirk, J. P. Mason, S. J. Mumford *et al.*, "Science platforms for heliophysics data analysis," *arXiv preprint arXiv:2301.00878*, 2023.

[42] Z. Jiao, H. Sun, X. Wang, W. Manchester, T. Gombosi, A. Hero, and Y. Chen, "Solar flare intensity prediction with machine learning models," *Space weather*, vol. 18, no. 7, p. e2020SW002440, 2020.

[43] Magnetograms - national space observatory (nso). Accessed on 28 January 2023. [Online]. Available: https://nso.edu/data/nisp-data/magnetograms/

[44] P. T. Gallagher, Y.-J. Moon, and H. Wang, "Active-region monitoring and flare forecasting–i. data processing and first results," *Solar Physics*, vol. 209, pp. 171–183, 2002.

[45] S. Haykin, Z. Chen, and S. Becker, "Stochastic correlative learning algorithms," *IEEE Transactions on Signal Processing*, vol. 52, no. 8, pp. 2200–2209, 2004.

[46] K. Florios, I. Kontogiannis, S. H. Park, J. A. Guerra, F. Benvenuto, D. S. Bloomfield, and M. K. Georgoulis, "Forecasting solar flares using magnetogram-based predictors and machine learning," *Solar Physics*, vol. 293, no. 2, p. 28, 2018.

[47] Y. Yuan, F. Y. Shih, J. Jing, and H.-M. Wang, "Automated flare forecasting using a statistical learning technique," *Research in Astronomy and Astrophysics*, vol. 10, no. 8, p. 785, 2010.

[48] O. Sagi and L. Rokach, "Ensemble learning: A survey," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 8, no. 4, p. e1249, 2018.

[49] E. Jang, S. Gu, and B. Poole, "Categorical reparametrization with gumble-softmax," in *International Conference on Learning Representations (ICLR 2017)*. OpenReview. net, 2017.

[50] A. Kucuk, J. M. Banda, and R. A. Angryk, "Solar event classification using deep convolutional neural networks," in *Artificial Intelligence and Soft Computing: 16th International Conference, ICAISC 2017, Zakopane, Poland, June 11-15, 2017, Proceedings, Part I 16*. Springer, 2017, pp. 118–130.

[51] J. M. Banda, R. A. Angryk, and P. C. Martens, "On the surprisingly accurate transfer of image parameters between medical and solar images," in *2011 18th IEEE International Conference on Image Processing*. IEEE, 2011, pp. 3669–3672.

[52] X. Chen, Y. Xu, D. W. K. Wong, T. Y. Wong, and J. Liu, "Glaucoma detection based on deep convolutional neural network," in *2015 37th annual international conference of the IEEE engineering in medicine and biology society (EMBC)*. IEEE, 2015, pp. 715–718.

[53] D. Ciresan, A. Giusti, L. Gambardella, and J. Schmidhuber, "Deep neural networks segment neuronal membranes in electron microscopy images," *Advances in neural information processing systems*, vol. 25, 2012.

[54] D. Wang, A. Khosla, R. Gargeya, H. Irshad, and A. H. Beck, "Deep learning for identifying metastatic breast cancer," *arXiv preprint arXiv:1606.05718*, 2016.

[55] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[56] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[57] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[58] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real world classification problems?" *The journal of machine learning research*, vol. 15, no. 1, pp. 3133–3181, 2014.

[59] T. G. Dietterich *et al.*, "Ensemble learning," *The handbook of brain theory and neural networks*, vol. 2, no. 1, pp. 110–125, 2002.

[60] L. I. Kuncheva and C. J. Whitaker, "Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy," *Machine learning*, vol. 51, no. 2, p. 181, 2003.

[61] Y. Xiao, J. Wu, Z. Lin, and X. Zhao, "A deep learning-based multi-model ensemble method for cancer prediction," *Computer methods and programs in biomedicine*, vol. 153, pp. 1–9, 2018.

[62] Z. Niu, G. Zhong, and H. Yu, "A review on the attention mechanism of deep learning," *Neurocomputing*, vol. 452, pp. 48–62, 2021.

[63] Y. Yu, X. Si, C. Hu, and J. Zhang, "A review of recurrent neural networks: Lstm cells and network architectures," *Neural computation*, vol. 31, no. 7, pp. 1235–1270, 2019.

[64] S. Li, W. Li, C. Cook, C. Zhu, and Y. Gao, "Independently recurrent neural network (indrnn): Building a longer and deeper rnn," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5457–5466.

[65] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[66] F. A. Gers, J. Schmidhuber, and F. Cummins, "Learning to forget: Continual prediction with lstm," *Neural computation*, vol. 12, no. 10, pp. 2451–2471, 2000.

[67] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[68] G. Weiss, Y. Goldberg, and E. Yahav, "On the practical computational power of finite precision rnns for language recognition," *arXiv preprint arXiv:1805.04908*, 2018.

[69] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, "Massive exploration of neural machine translation architectures," *arXiv preprint arXiv:1703.03906*, 2017.

[70] C. Marzban, "The roc curve and the area under it as performance measures," *Weather and Forecasting*, vol. 19, no. 6, pp. 1106–1114, 2004.

[71] R. A. Rutenbar, "Simulated annealing algorithms: An overview," *IEEE Circuits and Devices magazine*, vol. 5, no. 1, pp. 19–26, 1989.

[72] i4ds, "SDOBenchmark: Synthetic Data for Outlier Detection Benchmark," https://i4ds.github.io/SDOBenchmark/#, 2018, accessed: December 05, 2023.

[73] S. Toriumi, C. J. Schrijver, L. K. Harra, H. Hudson, and K. Nagashima, "Magnetic properties of solar active regions that govern large solar flares and eruptions," *The Astrophysical Journal*, vol. 834, no. 1, p. 56, 2017.

[74] L. Vincent, "Morphological area openings and closings for grey-scale images," in *Shape in picture: mathematical description of shape in grey-level images*. Springer, 1994, pp. 197–208.

[75] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," in *Artificial Intelligence in Medicine: 8th Conference on Artificial Intelligence in Medicine in Europe, AIME 2001 Cascais, Portugal, July 1–4, 2001, Proceedings 8*. Springer, 2001, pp. 63–66.

[76] G. Pearce and R. Harrison, "Sympathetic flaring," *Astronomy and Astrophysics (ISSN 0004-6361), vol. 228, no. 2, Feb. 1990, p. 513-516.*, vol. 228, pp. 513–516, 1990.

[77] E. Gabriel, G. E. Fagg, G. Bosilca, T. Angskun, J. J. Dongarra, J. M. Squyres, V. Sahay, P. Kambadur, B. Barrett, A. Lumsdaine *et al.*, "Open mpi: Goals, concept, and design of a next generation mpi implementation," in *Recent Advances in Parallel Virtual Machine and Message Passing Interface: 11th European PVM/MPI Users' Group Meeting Budapest, Hungary, September 19-22, 2004. Proceedings 11*. Springer, 2004, pp. 97–104.

[78] X. Huang, H. Wang, L. Xu, J. Liu, R. Li, and X. Dai, "Deep learning based solar flare forecasting model. i. results for line-of-sight magnetograms," *The Astrophysical Journal*, vol. 856, no. 1, p. 7, 2018.

[79] T. Sakamoto, T. Furukawa, K. Lami, H. H. N. Pham, W. Uegami, K. Kuroda, M. Kawai, H. Sakanashi, L. A. D. Cooper, A. Bychkov *et al.*, "A narrative review of digital pathology and artificial intelligence: focusing on lung cancer," *Translational Lung Cancer Research*, vol. 9, no. 5, p. 2255, 2020.

[80] S. Sharma, S. Sharma, and A. Athaiya, "Activation functions in neural networks," *Towards Data Sci*, vol. 6, no. 12, pp. 310–316, 2017.

[81] K. Koidl, "Loss functions in classification tasks," *School of Computer Science and Statistic Trinity College, Dublin*, 2013.

[82] A. M. Taqi, A. Awad, F. Al-Azzo, and M. Milanova, "The impact of multi-optimizers and data augmentation on tensorflow convolutional neural network performance," in *2018 IEEE Conference on Multimedia Information Processing and Retrieval (MIPR)*. IEEE, 2018, pp. 140–145.

[83] D. Gopika and B. Azhagusundari, "An analysis on ensemble methods in classification tasks," 2014.

[84] M. Abadi, "Tensorflow: learning functions at scale," in *Proceedings of the 21st ACM SIGPLAN international conference on functional programming*, 2016, pp. 1–1.

[85] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[86] T. Shanthi and R. Sabeenian, "Modified alexnet architecture for classification of diabetic retinopathy images," *Computers & Electrical Engineering*, vol. 76, pp. 56–64, 2019.

[87] L. C. Yan, B. Yoshua, and H. Geoffrey, "Deep learning," *nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[88] K. K. Singh, M. Siddhartha, and A. Singh, "Diagnosis of coronavirus disease (covid-19) from chest x-ray images using modified xceptionnet," *Romanian Journal of Information Science and Technology*, vol. 23, no. 657, pp. 91–115, 2020.

[89] C. Chola and J. B. Benifa, "Detection and classification of sunspots via deep convolutional neural network," *Global Transitions Proceedings*, vol. 3, no. 1, pp. 177–182, 2022.

[90] Y. Wang, J. Liu, Y. Jiang, and R. Erdélyi, "Cme arrival time prediction using convolutional neural network," *The Astrophysical Journal*, vol. 881, no. 1, p. 15, 2019.

[91] N. S. Punn and S. Agarwal, "Automated diagnosis of covid-19 with limited posteroanterior chest x-ray images using fine-tuned deep neural networks," *Applied Intelligence*, vol. 51, no. 5, pp. 2689–2702, 2021.

[92] Y. Zhang, "Lung segmentation with nasnet-large-decoder net," *arXiv preprint arXiv:2303.10315*, 2023.

[93] A. Çinar and M. Yildirim, "Detection of tumors on brain mri images using the hybrid convolutional neural network architecture," *Medical hypotheses*, vol. 139, p. 109684, 2020.

[94] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. K. Warmuth, "Occam's razor," *Information processing letters*, vol. 24, no. 6, pp. 377–380, 1987.

[95] M. Mngomezulu, M. Gwetu, and J. V. Fonou-Dombeu, "Solar flare forecasting using individual and ensemble rnn models," in *International Conference on Innovative Techniques and Applications of Artificial Intelligence*. Springer, 2023, pp. 307–320.

[96] Y. Roggo, L. Duponchel, and J.-P. Huvenne, "Comparison of supervised pattern recognition methods with mcnemar's statistical test: Application to qualitative analysis of sugar beet by near-infrared spectroscopy," *Analytica Chimica Acta*, vol. 477, no. 2, pp. 187–200, 2003.

[97] L. Rice, E. Wong, and Z. Kolter, "Overfitting in adversarially robust deep learning," in *International Conference on Machine Learning*. PMLR, 2020, pp. 8093–8104.