Advancing Solar Flare Detection: Integrating LSTM and Convolutional Layers in Neural Networks

Melvin Adkins
New College of Florida
melvin.adkins24@ncf.edu

Fehmi Neffati
New College of Florida
fehmi.neffati24@ncf.edu

Nikolas Santamaria
New College of Florida
nikolas.santamaria@ncf.edu

12/17/2023

Introduction

In the interdisciplinary realm of data science and astrostatistics, the study presented herein embarks on an ambitious endeavor to revolutionize solar flare detection.

Central to this pursuit is the Geostationary Operational Environmental Satellite's X-Ray Sensor (GOES XRS), a pivotal instrument in space weather surveillance since 1975. The XRS meticulously captures solar X-ray irradiance across two critical bandpass channels, XRS-A (0.05-0.4 nm) and XRS-B (0.1-0.8 nm), offering a window into the dynamic processes of the Sun. The XRS Level 2 data, comprising flare summary and 1-minute averages, collected over the span of 2017 to 2023, forms the backbone of our analysis. Our study transcends traditional approaches by leveraging a neural network model integrating Long Short-Term Memory (LSTM) and Convolutional layers.

This innovative model processes wide format flux readings, distinguishing 'flares' from 'non-flares' with enhanced precision. In contrast to existing NASA models, our approach is specifically tailored to address the challenges posed by noisy data, a known limitation in solar flare detection. The objective is not merely to replicate the success of the current models in identifying flares but to push the boundaries by detecting both previously identified and new flares. The preliminary results of this research are promising, suggesting a paradigm shift in the accuracy of solar flare prediction. This advancement has far-reaching implications, from improving our understanding of solar dynamics to enhancing the reliability of space weather forecasting.

By pushing the frontiers of astrostatistics and applying cutting-edge data science techniques, this study aims to make a substantial contribution to the field and pave the way for future innovations.

Literary Review

A flare, or solar flare, is an eruption of electromagnetic radiation which may last minutes or hours on the surface of the Sun. These events are associated with twisting magnetic flux lines that eventually become straight once broken due to increasing stress. Additionally, one can identify solar flares graphically, by looking at the light curves which display flux values (measured in watts per square inch) versus time. The existing NASA algorithm is used for detecting these events.

NASA's algorithm begins with processing 1-minute solar flux data. Variables are initialized in this process, and a real-time data frame is created and checked for quality. Afterwards, a smoothing filter is applied to reduce noise. If the minimum flux is below a certain threshold, the status is recorded as impaired. In the following steps, the algorithm checks for the start of a flare by considering various characteristics such as the inflection points, and comparison of flux values, and the fitted exponential. If certain conditions are satisfied, the status is set as the event starts. This algorithm also tracks the rise, peak, and fall of a flare. Lastly, the algorithm records summaries based on its analysis across the different status points.

One of the challenges this algorithm faces is the signal noise. This noise, as described in "User's Guide for GOES-R XRS L2 Products", causes higher false-positive rates of flare peaks. Apart from this, the algorithm also contains many checkpoints and decision nodes for detecting solar flares. In order to streamline this effort, a machine learning model is proposed in this paper, with the primary goal of detecting all solar flares identified by the original algorithm and more. This process begins with data collection and data preprocessing steps.

Methodology

1. Data Collection and processing

In our study, we used data from the GOES-R Space Weather Satellites 16, 17, and 18, focusing on 1-minute average XRS-a readings. This dataset, which encompasses a comprehensive five-year period, is combined with NOAA's flare summaries containing timestamps and flare statuses(EVENT_START, EVENT_PEAK, EVENT_END). We used an outer merge strategy for data integration across the three satellites, ensuring a robust and complete dataset.

To facilitate our research goals, we implemented a binary classification system for flare detection. In this system, instances corresponding to the flare peak, marked as 'EVENT_PEAK,' were labeled as '1,' while all other instances received a label of '0.' This binary classification schema simplifies the task of identifying significant flare events within the dataset.

Our windowing function, inspired by the methodology in "Flare Statistics for Young Stars from a Convolutional Neural Network Analysis of TESS Data", restructures this data into a wide format. This step positions the flare peak centrally, facilitating more effective identification and analysis of solar flares and associated phenomena. This approach not only enhances the dataset's utility for our neural network model but also aligns with contemporary methodologies in astrostatistics, promising more accurate and insightful solar flare detection and analysis.

To evaluate the optimal configuration for solar flare detection, a Python function was utilized to test various combinations of window sizes and non-peak proportions. The function iterated through window sizes of 20, 30, 50, 70, 80, 90, 100, 120, 150 and non-peak proportions of 0, 1, 2, 3, 4 and 5. The window size is a parameter that determines the number of observed flux values in each row and the non-peak proportions represent the number of non-flares to flares to address data imbalance. For each combination, a dataset was created using a custom window_creater function.

The dataset was then partitioned into training, validation, and test sets, with a 70-15-15 split, and stratified based on the target variable. These sets are created to train the model, improve its performance, and evaluate its ability to generalize. The data was scaled using StandardScaler to increase convergence of the Adam optimization algorithm, and reshaped to fit the neural network's input requirements. Each unique setup was saved in a list for further analysis, enabling the assessment of different configurations for model optimization.

2. Model Architecture

a. NN Structure

The neural network architecture chosen for this research is carefully designed to address the unique challenges posed by time series data. Time series data are inherently sequential, demanding a model that can effectively capture temporal dependencies. As such, our architecture starts with a Bidirectional Long Short-Term Memory (LSTM) layer as the foundational element.

The core of our model is a Bidirectional LSTM layer, which processes input data in both forward and backward directions. This bidirectional nature enables the network to capture temporal dependencies across past and future time steps. LSTMs are particularly well-suited for this task due to their ability to mitigate the vanishing gradient problem and maintain long-term dependencies.

b. Layers

The neural network is composed of several additional layers, each contributing to feature extraction and classification:

Convolutional layers are employed to detect local patterns and variations within the time series data. Our architecture incorporates two convolutional layers.

- The first convolutional layer consists of 16 filters with a filter size of 2, employing the Rectified Linear Unit (ReLU) activation function. This layer extracts lower-level features.
- The second convolutional layer comprises 64 filters with a filter size of 3, also utilizing ReLU activation. This layer captures higher-level features.

MaxPooling layers serve to downsample feature maps produced by convolutional layers, reducing computational complexity and preventing overfitting. In our architecture, two MaxPooling layers are integrated.

- The first MaxPooling layer employs a pool size of 7, significantly reducing dimensionality.
- The second MaxPooling layer adopts a pool size of 2, further reducing feature map dimensions.

Dropout layers are incorporated to combat overfitting by randomly deactivating a fraction of neurons during training. A dropout rate of 0.1 is applied after the first convolutional layer. An additional dropout rate of 0.5 is applied after the second convolutional layer. Lastly, a final dropout rate of 0.1 is applied between the densely connected layers.

The flatten layer transforms the output into a one-dimensional output, so that binary classification can be performed in the final layers. The fully connected layers process the extracted features that are present from the flatten layer, and produce the ultimate model output.

- The first fully connected layer is composed of 32 neurons, employing ReLU activation.
- The second fully connected layer consists of a single neuron with sigmoid activation, making it well-suited for binary classification tasks.

c. Activation functions

ReLU activation functions are employed in both convolutional and fully connected layers. ReLU introduces non-linearity, allowing the model to capture complex patterns within the data. Its computational efficiency helps mitigate the vanishing gradient problem, and allows for faster training. It is also a common choice in the field of machine learning.

In the output layer, the sigmoid activation function is used. Sigmoid squashes the network's raw output into a probability range between 0 and 1, making it suitable for binary classification tasks like the one at hand.

d. Architectural Advantages

The architecture's choice and configuration are particularly advantageous for time series datasets:

- Bidirectional LSTM effectively captures temporal dependencies by considering both past and future information, an essential aspect of modeling time series data.
- Convolutional layers excel at detecting local patterns and variations within the time series through the use of learned filters, which is vital for identifying short-term trends and anomalies.
- The ReLU activation functions introduce non-linearity, enabling the model to learn complex, non-linear relationships within the time series data. It is also computationally efficient.
- The sigmoid activation function in the output layer is well-suited for binary classification tasks, such as distinguishing between solar flare events and non-events in the time series.
- The flatten and dense layers are needed to reduce the dimensionality of the extracted information, and to output a single value to identify flares in the data set.

Results

For windows of size 70, The results indicate high performance for both configurations. For the 4:1 ratio, precision and recall are exceptionally high (0.98-0.99) for both classes, indicating accurate predictions and minimal false negatives/positives. The 5:1 ratio shows slightly lower precision for class 1 (0.92) but maintains high recall (0.98), suggesting more false positives but still high correct flare detection. The 4:1 ratio may be more balanced, effectively managing the trade-off between precision and recall, while the 5:1 ratio, despite its slightly lower precision, still achieves remarkable overall accuracy.

The 4:1 ratio model, with its higher precision and recall for class '1' (flares), is likely better for detecting flares that the NASA algorithm missed. Its balanced approach minimizes both false negatives and positives, crucial for identifying missed flares. The slightly lower precision in the 5:1 ratio, despite high recall, might lead to more false positives, which could be less ideal for detecting missed flares with high confidence.

Discussion

Our findings suggest that for windows of size 70, two specific configurations stand out in terms of performance. These configurations are based on the ratio of non-peak to peak instances (4:1 and 5:1), and they exhibit high accuracy and precision.

The model using the 4:1 ratio demonstrates exceptional precision and recall for both classes. With precision and recall values ranging from 0.98 to 0.99 for both classes, this configuration showcases accurate predictions and minimal occurrences of false negatives and false positives. This indicates the model's proficiency in accurately detecting flares and non-flare instances.

The 5:1 ratio configuration, while slightly lower in precision for class 1 (0.92), maintains a high recall (0.98). This implies that while it may produce more false positives, it still achieves a commendable level of correct flare detection. In essence, this configuration excels in identifying flares but might have a marginally higher rate of false positives.

The model utilizing the 4:1 ratio, characterized by higher precision and recall for class '1' (flares), is well-suited for detecting flares that the NASA algorithm might have missed. Its balanced approach effectively manages the trade-off between precision and recall, minimizing both false negatives and false positives. This is particularly critical for identifying missed flares with confidence, making it a strong candidate for accurate flare detection.

On the other hand, the 5:1 ratio configuration, despite slightly lower precision, maintains high recall, ensuring that it captures a significant portion of the actual flare events. However, its potential for more false positives might make it less ideal for situations where high confidence in flare detection is paramount.

For the 4:1 Ratio: using the precision and recall values, we calculated the following:

True Positives (TP) = Recall * Support = 0.96 * 1136 = 1090.56 (rounded to 1090)

False Positives (FP) = Support - TP = 1136 - 1090 = 46

We calculated number of flares detected by our model that NASA also detected:
True Positives Detected by Both (NASA and our Model) = TP = 1090.

For the 5:1 Ratio: using the precision and recall values, we calculated the following:

True Positives (TP) = Recall * Support = 0.98 * 1136 = 1112.48 (rounded to 1112)

False Positives (FP) = Support - TP = 1136 - 1112 = 24

We calculated number of flares detected by our model that NASA also detected:
True Positives Detected by Both (NASA and our Model) = TP = 1112

Using the 4:1 proportion, the model successfully identified 1090 flares that were also identified by NASA. This value increased to 1112 successfully identified flares for the data with a 5:1 proportion.

Limitations:

While this study has yielded valuable insights into solar flare detection using neural network models applied to XRS-A flux readings, it is essential to acknowledge several limitations that may impact the interpretation and generalizability of the results.

One notable limitation pertains to gaps in the dataset. Specific time intervals, such as those from 2017-02-07 18:29:00 to 2017-02-07 20:04:00 and from 2017-02-08 12:46:00 to 2017-02-09 00:25:00, exhibited a lack of data readings. While these gaps were relatively small in proportion to the overall dataset, their presence introduced incomplete temporal coverage.

In addressing data gaps, a decision was made to remove rows with missing values. While this approach was practical given the limited extent of missing data, it should be noted that data imputation was not employed, potentially impacting the dataset's completeness.

The dataset utilized in this research focuses exclusively on XRS-A flux readings, capturing soft X-ray fluxes within a specific energy range. As a result, the model's applicability is limited to detecting solar flares within this energy band and does not encompass a broader range of flux measurements.

The reliance solely on labels provided by the NASA algorithm for flare detection introduces a limitation related to automated labeling. The model's training and evaluation are based on these automated labels, which may not be infallible and could influence model performance.

One potential avenue for future research involves enhancing the dataset by incorporating flares detected by the model but missed by the NASA algorithm. The process of manually verifying and labeling these additional flares, as well as their subsequent integration into the dataset for retraining the model, represents a promising direction for further investigation.

In conclusion, while this study has provided valuable insights into solar flare detection using neural network models and XRS-A flux readings, these limitations should be considered when interpreting the results and when contemplating the model's broader applicability to different datasets and scenarios in the field of astrostatistics.

References

Machol, J.m Codrescu, S., & Peck, C. (2023). User’s Guide for GOES-R XRS L2 Products. Retrieved from https://data.ngdc.noaa.gov/platforms/solar-space-observing-satellites/goes/goes16/l2/docs/GOES-R_XRS_L2_Data_Users_Guide.pdf

Wang, X., Chen, Y., Toth, G., Manchester, W. B., Gombosi, T. I., Hero, A. O., Jiao, Z., Sun, H., Jin, M., & Liu, Y. (Year). Predicting Solar Flares with Machine Learning: Investigating Solar Cycle Dependence. *The Astrophysical Journal*, 895:3, Retrieved from Canvas.

Feinstein, A. D., Montet, B. T., Ansdell, M., Nord, B., Bean, J. L., Günther, M. N., Gully-Santiago, M. A., & Schlieder, J. E. (Year). Flare Statistics for Young Stars from a Convolutional Neural Network Analysis of TESS Data. *The Astronomical Journal*, DOI: 10.3847/1538-3881/abac0a

Acknowledgements

1. Dr. Vinay Kashyap
2. Dr. Bernhard Klingenberg