Project Report on

Unsupervised Learning-based Prediction of Earth-Impacting Coronal Mass Ejections (CMEs)

Submitted to

Dr. M.V. Sunil Krishna

Bachelor of Technology

In

Engineering Physics

Submitted By

Anamika Patel 23123006	Kanchan 23123020	Manisha 23123025

Indian Institute of Technology, Roorkee

Introduction

Coronal Mass Ejections (CMEs) are massive bursts of plasma and magnetic fields from the Sun's corona. When directed toward Earth, they can trigger geomagnetic storms, disrupt satellite communication, damage power grids, and affect GPS systems. With the increasing reliance on technology, accurate and timely prediction of CMEs has become critical for minimizing their adverse effects.

NASA's Solar Dynamics Observatory (SDO), through its Atmospheric Imaging Assembly (AIA), captures high-resolution images of the Sun in various wavelengths. Among these, the 171 Ångström channel is particularly valuable for observing coronal loops and detecting early signs of CME activity.

This project focuses on the first step toward building a predictive system: the automated extraction and preprocessing of solar images and associated metadata. The aim is to create a dataset ready for unsupervised machine learning models, which can identify CME-related patterns without requiring labeled data.

Methodology

The methodology followed in this project includes a systematic pipeline for extracting, processing, and organizing solar image data suitable for further analysis and machine learning tasks. The steps are as follows:

Step 1: Library Installation and Setup

To begin with, the required scientific and solar physics libraries are installed using `pip`:

```
pip install sunpy astropy parfive beautifulsoup4 lxml zeep drms
pip install reproject
pip install mpl_animators
```

These libraries are essential for:

- Accessing solar data (SunPy, DRMS)
- Handling astronomical units (AstroPy)
- Image and metadata processing
- Plotting and animation (Matplotlib, mpl_animators)

Step 2: Import Required Libraries

Essential Python libraries are imported to handle:

- Dates, files, and directories (`datetime`, `os`, `shutil`)

- Data processing (`pandas`, `numpy`)
- Progress tracking (`tqdm`)
- Plotting (`matplotlib`)
- Solar image retrieval and processing (`sunpy`, `astropy`)

Step 3: Output Folder Creation

os.makedirs("extracted_images", exist_ok=True)

Creates a directory `extracted_images/` to store processed PNG images.

Step 4: Random Date Generation

start_date = datetime(2023, 1, 1)

end_date = datetime(2023, 8, 31)

num_samples = 20

Randomly generate 20 distinct dates between January and August 2023.

These dates serve as the basis for solar data queries.

Step 5: Data Retrieval from SunPy

For each of the 20 dates:

1. Fix the query time to 12:00 PM UTC.
2. Define a 2-hour time window.
3. Search for AIA 171 Å images using SunPy's `Fido` interface:

```
result = Fido.search(
    a.Time(start_time, end_time),
    a.Instrument.aia,
    a.Wavelength(171 * u.angstrom)
)
```

4. Download the **first available** image.
5. Read it using `sunpy.map.Map`.

Step 6. Image Conversion and Saving

Read image data from the .fits file.

Normalize the image to 8-bit grayscale.

Save as a .png image using matplotlib.pyplot.imsave() in the extracted_images/directory.

```
plt.imsave(png_path, norm_data, cmap='gray')
```

Step 7. Metadata Extraction

From the image FITS header, extract and store the following metadata:

date_obs: Observation timestamp

wavelength: Wavelength in Å

exposure_time: Exposure duration

instrument, telescope, detector: Hardware details obs_type: Type of observation

x_scale, y_scale: Spatial resolution

obs_title: Observation ID

image_path: Saved image path

All metadata are stored as a list of dictionaries and then written to a CSV file:

```
df = pd.DataFrame(meta_data)
df.to_csv("aia_images_metadata.csv", index=False)
```

Step 8. Copying All FITS Files

Since FITS files are downloaded and stored in the SunPy cache, they are copied to a dedicated local folder:

```
sunpy_cache_path = Path(sunpy.config.get('downloads', 'download_dir')).expanduser()
destination_path = Path("aia_fits_files")
```

Use Path.rglob("*.fits") to locate all FITS files recursively.

Copy them into aia_fits_files/ for offline use.

Final Output

PNG images stored in: extracted_images/

Metadata CSV: aia_images_metadata.csv

FITS files: aia_fits_files/

Output Summary

Files and Folders:

- **`extracted_images/`: Contains all PNG-format solar images**
- **`aia_images_metadata.csv`: CSV file containing metadata for each image**

Sample Metadata Row:

date_obs	wavelength	exposure_time	instrument	image_path
2023-05-12T12:01:00Z	171	2.0 s	AIA	extracted_images/aia_20230512_120100.png

Machine Learning Phase

After preparing the image dataset, a two-phase machine learning pipeline was implemented to analyze and classify the solar images:

Phase 1: Unsupervised Clustering

- Algorithm Used: KMeans Clustering (with 3 clusters)
- Objective: Group similar images into categories such as No Event, Solar Flare, and CME.
- Clusters were visualized and manually labeled to map each cluster to an event class.

Phase 2: CNN-based Classification

- **Model Type**: Convolutional Neural Network (CNN)
- **Libraries Used**: TensorFlow, Keras
- **Input Shape**: 224x224 grayscale images
- **Architecture**: Conv2D → MaxPooling → Conv2D → MaxPooling → Flatten → Dense → Output
- **Output**: 3 classes (No Event, Flare, CME)
- **Training**:
    - Data split into training and testing (70/30)
    - Accuracy achieved on test data was printed and confirmed with predictions on sample images.

This phase demonstrated the use of a small labeled dataset (from clustering) to train a supervised classifier capable of identifying CME-like structures.

Tools and Technologies Used

Tool/Library	**Purpose**
Python	Core programming language
sunpy, astropy	Download and handle solar image data
matplotlib	Image conversion and saving
numpy, pandas	Data processing and CSV creation
tqdm	Progress visualization
datetime, os	File system and time range operations

Applications and Future Scope

- Clustering solar images to identify CME-related patterns
- Anomaly detection using autoencoders or PCA
- Creating synthetic labels for supervised learning
- Combining with solar wind and magnetogram data for impact prediction

This dataset forms the foundation for intelligent space weather forecasting systems that can reduce the risk of technological disruptions on Earth.

Conclusion

This project successfully completes the first and most crucial step in building a data-driven system for the prediction of Earth-impacting Coronal Mass Ejections (CMEs). By automating the extraction of solar images from NASA's Solar Dynamics Observatory (SDO) and preprocessing them into a machine learning-ready format, we have created a high-quality dataset that includes both image and metadata components.

The use of random date sampling ensures variability in solar activity, and the collected metadata provides valuable context for future unsupervised learning models. This well-structured dataset will serve as the foundation for advanced techniques such as clustering, anomaly detection, and autoencoders, which can help identify hidden patterns and potential CME signatures without the need for labeled data.

In summary, the project:

- Demonstrates the capability to collect and preprocess real solar observational data.
- Provides a scalable framework for continuous data extraction.
- Sets the stage for applying machine learning to solve one of the key problems in space weather forecasting — early detection and prediction of CMEs.

This work is a stepping stone toward building intelligent systems that can learn from solar observations and contribute to the broader scientific and technological effort in protecting Earth’s infrastructure from solar hazards.

References

1. SunPy Documentation: https://docs.sunpy.org/
2. SDO/AIA Instrument Overview: https://sdo.gsfc.nasa.gov/
3. Astropy Project: https://www.astropy.org/

Github Respo

- https://github.com/Kanchanyadav12/cme_detection

Google Drive

- https://drive.google.com/drive/folders/11aRZdAmS2noyjSH3RyGH1xmN22lcfJ7I?usp=sharing