University of Applied Sciences and Arts Northwestern Switzerland
School of Engineering

Marius Giger

Unsupervised Anomaly Detection with Variational Autoencoders in Heliophysics

Master's Thesis

Institute for Data Science
University of Applied Sciences and Arts Northwestern Switzerland (FHNW)

Submitted to

Prof. Dr. André Csillaghy

Supervision

Prof. Dr. André Csillaghy

Prof. Dr. Jean Hennebert

Brugg, August 2022

Zusammenfassung

Deep Learning hat sich in vielen Bereichen als sehr erfolgreich erwiesen, da es aussagekräftige Features von Daten erlernen kann, ohne dass manuelles Feature Engineering erforderlich ist. Dies führt zu Modellen mit einer hohen Repräsentationsfähigkeit. Viele dieser Modelle beruhen jedoch auf überwachtem Lernen (supervised learning) und sind daher auf die Verfügbarkeit grosser Datensätze mit hochwertigen Annotationen angewiesen. Diese sind gerade im wissenschaftlichen Bereich oft nicht einfach zu beschaffen, da die Erstellung menschliche Expertise erfordert. Eine allgemeine Herausforderung für Forscher im Bereich der Heliophysik ist die geringe Anzahl von Anmerkungen in vielen der verfügbaren Datensätzen, die entweder nicht gelabeled oder deren Labels nicht ausreichend sind. Um diesen Engpass ("data labelling bottleneck") zu umgehen, ist unüberwachtes Deep Learning zu einer wichtigen Strategie geworden, wobei die Erkennung von Anomalien eine der wichtigsten Anwendungen ist. Unüberwachte Modelle wurden in verschiedenen Bereichen, wie der medizinischen Bildverarbeitung oder der Videoüberwachung, erfolgreich eingesetzt, um normale von anormalen Daten zu unterscheiden. In dieser Arbeit untersuchen wir, wie ein rein unüberwachter Ansatz zur Erkennung und Extraktion von Sonnenphänomenen aus Bildern im extrem ultravioletten-Bereich (EUV) verwendet werden kann. Dies auf der Basis von AIA Daten des Solar Dynamics Observatory (SDO) der NASA.

Wir setzen einen Deep-Learning-Ansatz auf der Grundlage eines Variational Autoencoders (VAE) ein und zeigen, dass ein solches Modell die Wissensentdeckung mit minimalen Annahmen unterstützen kann. Konkret übertragen wir eine Methode aus der medizinischen Bildverarbeitung auf die Heliophysik, den so genannten Context-Encoding Variational Autoencoder (ceVAE) [1], um eine niederdimensionale Repräsentation von Bildern der Sonne zu erlernen. Das daraus resultierende Modell ist in der Lage Bilder zu finden und Pixel zu lokalisieren, die von der erlerneten Verteilung abweichen, und somit abnormale und "interessante" Sonnenaktivität zu erkennen. Damit ermöglicht das Modell verschiedene Anwendungen, wie die Eingrenzung des Suchraums und die Entdeckung potenziell übersehener Phänomene oder Korrelationen in den grossen Mengen an bisher gesammelten Sonnendaten (Erkennung von Neuheiten und Anomalien).

Wir hoffen, durch die Verwendung eines unüberwachten Ansatzes einen Beitrag zu den Werkzeugen der Weltraumwetterüberwachung zu leisten und eine Methode zu entwickeln, die das Verständnis für die Triebkräfte des Weltraumwetters weiter verbessern kann. Diese Arbeit ist ein Schritt auf dem Weg zur unüberwachten Ereigniserkennung in der Heliophysik und bildet die Grundlage für weitere Studien, um das volle Potenzial der verfügbaren Daten auszuschöpfen.

Abstract

Deep learning has had great success in many fields because of its ability to learn strong feature representations without the need for hand-crafted features, resulting in models with a high representational power. However, many of these models are based on supervised learning and therefore depend on the availability of large annotated datasets. These are often difficult to obtain because they require human input. A general challenge for researchers in the heliophysics domain is the sparsity of annotations in many of the available datasets, which are either unlabelled or have inconclusive labels. To alleviate the data bottleneck of loosely annotated datasets, unsupervised deep learning has become an important strategy, with anomaly detection being one of the most prominent applications. Unsupervised models have been successfully applied in various domains, such as medical imaging or video surveillance, to distinguish normal from abnormal data. In this thesis, we investigate how a purely unsupervised approach can be used to detect and extract solar phenomena in SDO AIA images. We show how a variational autoencoder-based model can be used to detect out-of-distribution samples and localize interesting regions to detect solar activity. By using an unsupervised approach, we hope to contribute to the tools for space weather monitoring and further improve the understanding of the drivers of space weather.

Keywords: Unsupervised Deep Learning, Variational Autoencoders, Anomaly Detection, Out-of-Distribution Detection, Heliophysics, Solar Dynamics Observatory.

Acknowledgment

I would like to thank Prof. Dr. André Csillaghy and Prof. Dr. Jean Hennebert for their supervision and mentorship of this thesis. Further, I am grateful to the members of the AstroML group at i4DS@FHNW who contributed to the informative discussions and asked challenging questions during our weekly meetings. Special thanks to Jonathan Donzallaz for downloading and providing the SDO ML v1 dataset and to Elena Fritschi and David Giger for providing feedback and corrections. I would also like to thank my parents, my mother, Beatrice Giger, for her continuous support, and my father, Matthias Giger, for sparking my interest and giving me a basic understanding of the sun.

Contents

Nomenclature

Acronyms and Abbreviations

AE	Autoencoder
AIA	Atmospheric Imaging Assembly
ceVAE	Context-Encoding Variational Autoencoder
CLI	Command-Line Interface
CNN	Convolutional Neural Network
DDF	Distributed Data Frame
DL	Deep Learning
DLVM	Deep Latent-Variable Model
ELBO	Evidence Lower Bound
EUV	Extreme Ultraviolet
EVE	EUV Variability Experiment
GAN	Generative Adversarial Network
GPU	Graphical Processing Unit
HMI	Helioseismic and Magnetic Imager
JSOC	Joint Science Operations Center
MAE	Mean-Absolute Error
ML	Machine Learning
MSE	Mean-Squared Error
NAS	Network Accessible Storage
SDO	Solar Dynamics Observatory
SGD	Stochastic Gradient Descent
TPU	Tensor Processing Unit
UV	Ultraviolet
VAE	Variational Autoencoder

Chapter 1

Introduction

1.1 Machine Learning: An Indispensable Tool

Machine learning (ML) has undeniably had a meteoric rise in recent years, finally competing with human-level performance for several different tasks. Examples include *image classification* such as Google's recently introduced CoCa model achieving a top-1 performance of 91% on ImageNet [2]; *natural language processing* such as GPT-3 achieving an unprecedented performance on several natural language tasks and benchmarks such as translation and question-answering [3]; and the ability to *learn to play games* such as AlphaZero, which is a reinforcement agent that teaches itself to play a number of games such as chess, shogi and Go from scratch, beating other world-champion models [4]. Machine learning methods have been applied for almost 40 years, with Rumelhart et al. successfully showing how to use back-propagation to train an artificial neural network in 1986 [5], and have been studied for an even longer time (since the early 1950s). Recent breakthroughs were made possible by the availability of massive datasets and the necessary compute power introduced by much better processors, much more memory and the availability of powerful GPUs/TPUs.

Deep learning (DL) is viewed as one of the most promising machine learning approaches because it makes it possible to learn directly from raw data, thereby making labour-intensive and often suboptimal feature engineering obsolete.[1] The resulting models have a very high representational power and dominate the current state of the art for most machine learning tasks. However, this comes at the cost of massive compute requirements and the lack of interpretability of the resulting models. Because deep learning models are often purely data-driven and trained in an end-to-end manner (i.e., from raw data to prediction), the data used to train the model has to be representative and must capture the underlying distribution observed in the real world, which is not always easy to achieve (e.g., in the presence of heavily unbalanced data). Regardless of the drawbacks, deep learning has enabled unprecedented advances, in particular in areas with large amounts of complex data (e.g., image processing).

Recently, machine learning has also had increasing success in astronomy fuelled by the massive amount of data gathered by a variety of missions and instruments. The most common use cases include filtering and archiving massive amounts of data (big data processing), event detection and classification, event tracking, forecasting events (space weather prediction), image correction (super resolution, degradation correction) and data generation (e.g., magnetograms). For heliophysics in particular, the application of machine learning methods is very promising because much high-resolution scientific data is available. The field lies at the convergence of the study of the sun and space weather monitoring and attracts interest from both academia and industry due to the imminent impact space weather has on Earth. Large solar storms have the ability to disrupt and damage critical infrastructures such as satellites and power grids. For example, in 2022, SpaceX has lost 40 satellites to a geomagnetic storm, costing

[1] A good introduction into deep learning can be found in *The Deep Learning Book* by Goodfellow et al. [6]

millions of dollars.[2] To reliably and successfully predict these events and simultaneously understand the underlying physical processes that power the sun, machine learning plays an important role with increasing amounts of research dedicated to the use of ML methods.

A common problem in applied machine learning is the lack of labelled data, while unlabelled data is often found in large quantities. Labelling data is a challenging and often extremely labour-intensive task. A common practice involves crowdsourcing the labelling task,[3] which means that data is uploaded to a crowdsourcing platform (such as Amazon Mechanical Turk[4], Appen[5] or Upwork[6]) where users are asked to label the data for a reward. This works very well for simple tasks such as preparing text datasets for natural language processing or labelling everyday images. However, for scientific data such as biomedical or astrophysical observations, expert knowledge is required, and crowdsourcing is usually not an option. With ever-increasing amounts of data being collected, this task is becoming infeasible due to time and cost constraints.

1.2 Attempting to Solve the Data Labelling Bottleneck

The data labelling bottleneck of having large amounts of data that are not annotated or only loosely annotated is a challenge for heliophysics researchers. Because the amount of data collected by the various observatories continues to increase, it is crucial to develop methods to enrich this data with semantic information that can be used by downstream tasks such as event detection or space weather forecasting. Having a system that automatically detects and categorizes different kinds of solar activity would therefore be of great value to space weather monitoring applications and solar physicists and might also benefit other fields with similar challenges.

Solar physics often relies on the analysis of single observations, where an event is studied in great detail. This leads to a large part of the massive amounts of available data remaining unobserved. Similarly, machine learning-based methods for space weather monitoring often rely on labels that have been extracted using traditional image processing methods (e.g., for active region tracking or coronal hole detection) and are therefore limited by the availability and quality of labels. Thus, in most cases, they are not able to benefit from the fully available datasets. Furthermore, these supervised methods often rely on manually designed image features that are only partially data-driven and require a great deal of fine tuning. This prevents generalisability to other tasks and imposes assumptions on the model. Although these approaches work quite well, and given the large amounts of available solar imagery, we believe there is still much potential for a more data-driven approach that could contribute to further advances in heliophysics and realize the full potential of the available data.

1.3 A Way Forward with Unsupervised Methods

To alleviate the problem of lacking semantic information ("data labelling bottleneck"), a new approach to knowledge discovery and data mining is needed that does not require existing labels. One way to overcome this challenge is to apply unsupervised deep learning techniques as they do not require labels at training time. To better understand whether this applies to the field of heliophysics, this thesis explores unsupervised methods to extract semantic information from solar imagery without relying on existing labels. To that end, we studied a purely unsupervised method for identifying and localizing abnormal regions in solar images. We employed an approach that is not constrained by existing labels and directly learns from raw data without making use of handcrafted features and is therefore capable of leveraging the entire dataset resulting in a truly data-driven approach.

The retrieval of semantic information from images, such as the detection and localization of abnormal regions, is challenging due to the high dimensional structure imposed by the pixel space (height x

[2] https://www.bbc.com/news/world-60317806
[3] https://datacentricai.org/labeling-and-crowdsourcing/
[4] https://www.mturk.com/
[5] https://appen.com/
[6] https://www.upwork.com/

width x channels). This is especially true in solar imagery, where the data is highly dimensional due to the spatiotemporal nature of the observations (different images for different wavelengths at different timestamps). Learning a lower-dimensional representation of the data is therefore a necessary step in the pursuit of retrieving semantically relevant information. In order to achieve this, we made use of variational autoencoders (VAEs) and show how this method can be used to detect out-of-distribution events and regions in EUV images from NASA's Solar Dynamics Observatory (SDO). More specifically, we transferred a method that has been successfully applied in medical imaging to detect brain tumours by identifying and localizing abnormal regions in medical images (MRIs). We used the so-called *context-encoding variational autoencoder* (CeVAE) introduced by Zimmerer et al. [1] to learn a latent feature representation and to subsequently detect out-of-distribution samples on both an image and pixel-level. We then completed a model evaluation and demonstrated the applicability of the model to heliophysics.

This thesis documents several key contributions made to machine learning in heliophysics:

- We present a method that can extract semantic information from images without having any labels thereby making it possible to use the entirety of available data for training.
- We exhibit a way to extract a lower-dimensional feature representation from solar EUV images and significantly compress input images.
- We demonstrate how an unsupervised approach can be used to detect anomalous data on both the sample (full image) and pixel level thereby identifying unusual solar activity.
- We show how to effectively prepare and use Stanford's SDO ML dataset to train a deep learning model.

The rest of this thesis is structured as follows: Chapter 2 provides a brief introduction of the subject matter of this thesis, presents the state of the art relevant to the field and states the main research questions. Chapter 3 introduces the mathematical foundations of the machine learning methods used throughout this study and provides the necessary steps for out-of-distribution detection. Chapter 4 introduces the datasets used and lists the different data preparation steps. Chapter 5 summarizes different experiments that were completed using the proposed method, and Chapter 6 outlines the results of the experiments. Chapter 7 shows possible applications. Chapter 8 discusses the results and lists recommendations for future research. Finally, Chapter 9 concludes this thesis.

Chapter 2

Background

The following sections provide a brief introduction into the subject matter of this thesis, starting with a description of the sun, followed by an introduction into NASA's Solar Dynamics Observatory, a brief overview of machine learning in heliophysics and the state of the art for out-of-distribution detection models. Readers mainly interested in the machine learning topics of this thesis can skip to Section 2.3.

2.1 The Sun

The sun is the closest star to Earth. It is located at the center of our solar system and our primary source of energy providing light and heat for eons. The sun plays a vital role in our existence by constantly providing exactly the right amount of energy to heat but not overheat our planet and thereby making life possible. It was formed around 4.6 billion years ago[1] and is currently a main sequence star as defined by a Hertzsprung-Russel diagram (H-R diagram) which is illustrated in Figure 2.1. The H-R diagram (refer to [7]) shows the relation of luminosity and spectral type (temperature classification based on the surface temperature of a star). Most of the stars plot on the main sequence, the cooler red stars (also known as "red dwarfs") are smaller than the sun, yellow-whitish stars like the sun form the average and white and blue stars are bigger and hotter than the sun.

The sun is a giant spinning ball of very hot plasma (electrically charged gas in various states of ionisation depending on the temperature). It is mainly composed of hydrogen (73%) and a smaller amount of helium (25%). Other, heavier, elements such as oxygen, carbon and iron only make up about 2%.[2] 99.86% of our solar system's mass is concentrated in the sun. It has 1047-times the mass of Jupiter and 333'000 Earth masses. Its diameter is about 1.39 million kilometers (864,000 miles) or 109 times that of Earth thus it could easily swallow the entire Earth-Moon-system (diameter about 400'000 km). The sun has enough volume to fit about 1.3 million Earths.[3][4]

[1] https://www.space.com/58-the-sun-formation-facts-and-characteristics.html
[2] http://solar-center.stanford.edu/vitalstats.html
[3] https://handwiki.org/wiki/Astronomy:Sun
[4] https://solarsystem.nasa.gov/solar-system/sun/overview/

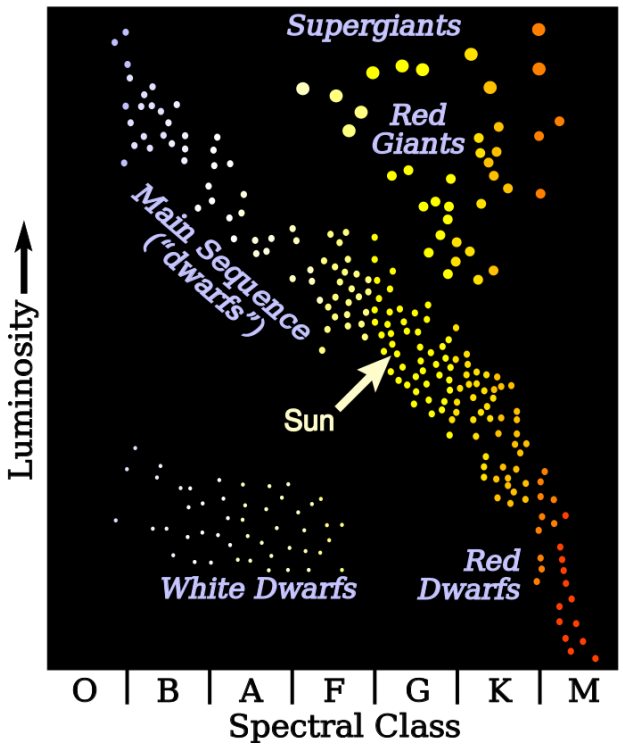


Figure 2.1: The Hertzsprung-Russel diagram (H-R diagram) shows the relationship between the luminosity and the spectral class (depending on surface temperature) for a group of stars. Most of the stars plot along the main sequence (diagonal). The sun is an average "yellowish-white" star. The cooler red stars have a surface temperature of 2000 to 3000 K. The sun has a surface (photospheric) temperature of 5800 K. The hottest blue stars have surface temperatures of over 30000 K and red giants have relatively low surface temperatures (less than 5000 K). White dwarfs are small, very dense, with low luminosity and with high surface temperatures. At the end of the lifecycle in the main sequence, stars evolve to "giants" (also called "red giants") and finally collapse to become "white dwarfs", once the fuel supply is exhausted. Eventually, in about 5 billion years, this will also be the fate of our sun. Source: Wikimedia

Eventually, in about 5 billion years, the sun will expand and become a red giant as the hydrogen fusion in its core diminishes. The core will then contract due to gravitation and heat up. Finally, the sun will inflate into a red giant due to the incipient fusion of helium into carbon. By that time, Mercury and Venus will probably be swallowed by the sun. After further burning, the sun will collapse due to the lack of fuel and become a white dwarf in about 6 billion years.[5]

The sun has been studied for centuries. Large sunspots visible to the naked eye were known as early as the Zhou Dynasty (before 800 BC)[8]. Telescopic observations started in 1610 but the 11-year activity cycle was not discovered before the mid 18th century. This can be explained by the fact that between 1645 and 1715 only very few sunspots were detected at all ("Maunder Minimum"[6]). This period coincides with a considerable cooling of the Earth ("little ice age"), but also other important triggers of the cooling are discussed (e.g., volcanism). With the recent technological advancements and space exploration we got an ever-increasing understanding of the sun's composition and the underlying physical processes. The theoretical base for the description of solar phenomena is *magnetohydrodynamics*, describing the behavior of moving hot plasma, the related magnetic fields, and electric currents. The broad field of *solar physics*, also called *heliophysics*, is dedicated to the study of the sun and its impact on Earth and space.

[5] https://handwiki.org/wiki/Astronomy:Sun#After_core_hydrogen_exhaustion
[6] https://www.britannica.com/science/Maunder-minimum

The sun is composed of several layers, which can broadly be split into the sun's interior and its atmosphere. A schematic illustration of the sun's layers is shown in Figure 2.3. This thesis specifically focuses on the solar atmosphere and studies data observing the upper transition region and corona. The next two paragraphs briefly introduce the different layers of the sun as well as solar phenomena that can be observed.

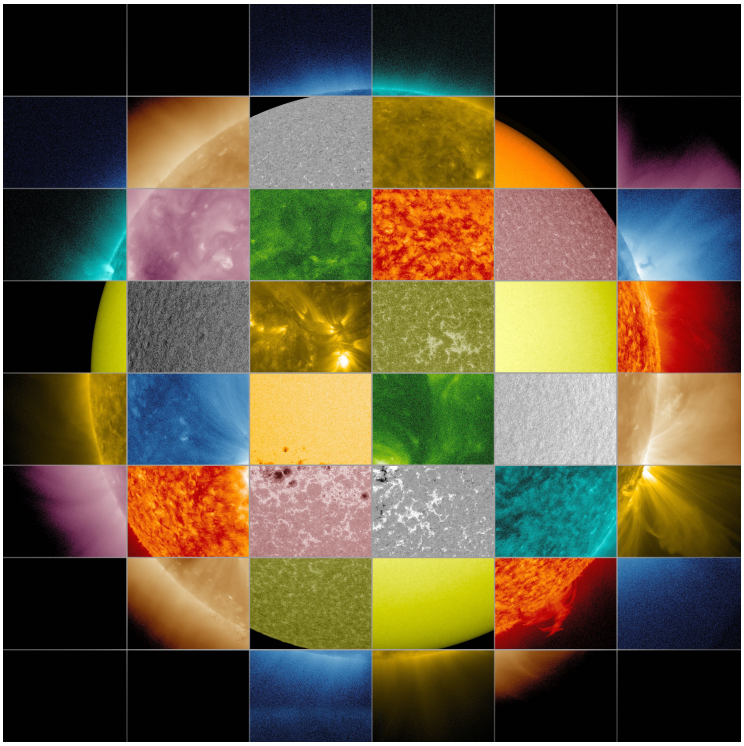


Figure 2.2: The different faces of the sun: Collage of images from NASA's Solar Dynamics Observatory (SDO) that shows observations of the sun in different wavelengths covering the surface and atmosphere. Source: NASA Goddard Space Flight Center

The Solar Interior The innermost layer of the sun is the *core*, where thermonuclear reactions are fusing hydrogen to form helium thereby creating extreme temperatures and producing energy that is eventually emitted as photons from the solar surface. The temperature of the sun's core is about 15 million Kelvin (K) and thereby the hottest part of the sun. Above the core lies the *radiative zone*, where energy slowly moves outward, taking more than 170'000 years to radiate through this layer by the means of radiative diffusion.[7] The *convection zone* is the outermost layer of the solar interior. At the base of this layer the temperature has dropped to about 2 million Kelvin and energy continues to move towards the surface through convection currents. These currents transport hot plasma to the surface where it is losing heat to space and cooling down again. As the plasma cools down, it sinks back to the bottom of the convection zone. The convective motions are visible as granules and supergranules on the solar surface.[8]

[7] https://www.nasa.gov/mission_pages/sunearth/science/solar-anatomy.html
[8] https://solarscience.msfc.nasa.gov/interior.shtml

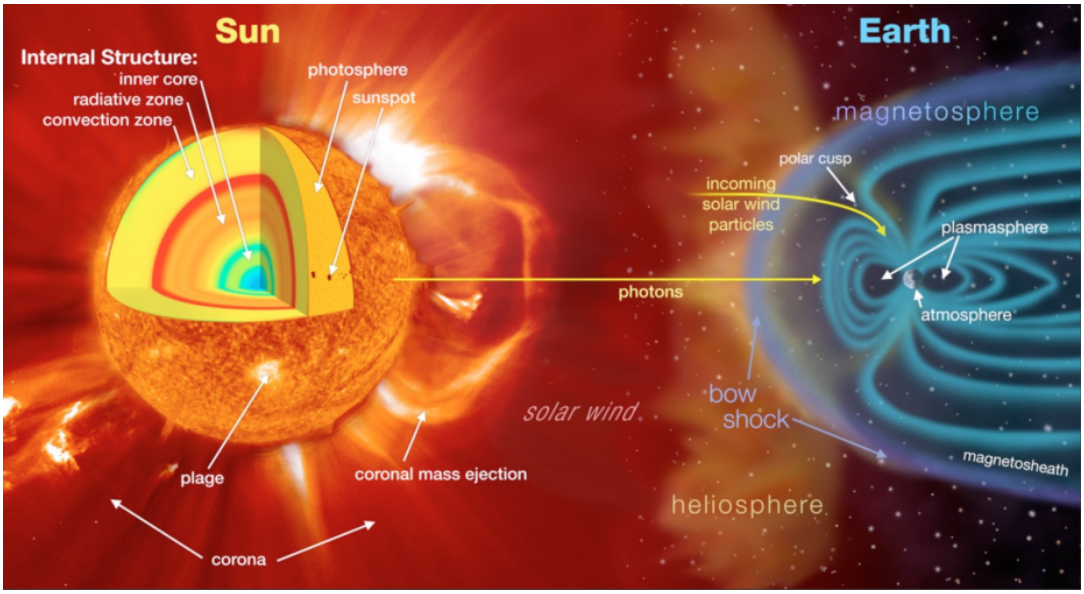


Figure 2.3: The solar interior and Sun-Earth interactions that influence space weather. Source: NASA Goddard Space Flight Center

The Solar Atmosphere The *photosphere* is the visible "surface" of the sun, emitting the white light we can see with our eyes (visible part of the electromagnetic spectrum). Because the sun is a giant ball of plasma, it does not have a solid surface, rather the density increases when getting closer to the core. The temperature of the photosphere is around 5'800 Kelvin.[9] There are several features that can be observed: *Sunspots*[10] appear as dark regions, caused by enhanced magnetic activity that inhibits the convective activity and therefore cools down the plasma. *Faculae*[11] appear as "bright spots" which are produced by concentrations of magnetic field lines. *Granules*,[12] which are the top of convection cells from the convection zone below. Figure 2.4 shows a close-up image of the photosphere exhibiting the typical granular pattern. The flow of material in photosphere can be measured by using the Doppler effect which reveals additional features such as *supergranules*[13] as well as large scale flows and patterns of waves and oscillations. The sun rotates around its axis in about 27 days, which was first observed by the motion of sunspots. Because the sun is a ball of gas, the rotation velocity varies based on the distance from the solar equator, which is called *differential rotation* (the angular velocity decreases with increased latitude). This means that the poles make one rotation approximately every 38 days and the equator every 24 days.[14]

[9] https://www.jpl.nasa.gov/infographics/mind-melting-facts-about-the-sun
[10] https://solarscience.msfc.nasa.gov/feature1.shtml#Sunspots
[11] https://solarscience.msfc.nasa.gov/feature1.shtml#Faculae
[12] https://solarscience.msfc.nasa.gov/feature1.shtml#Granules
[13] https://solarscience.msfc.nasa.gov/feature1.shtml#Supergranules
[14] https://www.nasa.gov/mission_pages/sunearth/science/solar-rotation.html

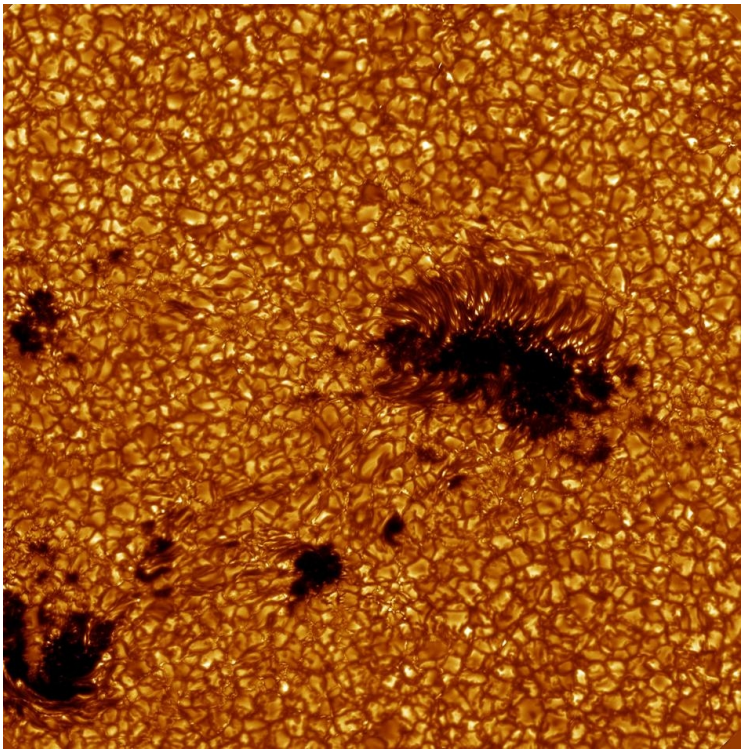


Figure 2.4: Close-up of the photosphere, showing different sunspots and the typical pattern of the convection cells, also called granules, where hot plasma is rising at the centre of the cells (brighter colour) and falling down on its edges (with a darker colour since it is cooler). The dark sunspots are photospheric regions of reduced surface temperature (about 4000 K), this caused by concentrations of magnetic flux that inhibits convection. The number of sunspots is a measure for the solar activity. The image was recorded by the 1 metre Solar Telescope of the Royal Swedish Academy of Sciences on the island of La Palma. Source: sun.org

Above the photosphere is the *chromosphere*, named after the bright reddish color it emits (H-alpha emission). The temperatures in the chromosphere rise from 6000 to about 20,000 Kelvin. In its lower parts, the solar material moves as gas or fluid, while in the upper parts the motion is dominated by strong magnetic forces. Different features can be observed when isolating the emissions stemming from the chromosphere: the *chromospheric network*[15] is a web-like pattern that outlines the supergranule cells; *filaments*[16] are dense, cooler clouds of material that are suspended by loops of the magnetic field; *plages* are bright patches surrounding sunspots; *prominences*[17] are in fact the same as filaments but are seen projecting out above the limb or the edge of the sun. As filaments and prominences get unstable, they can either fall back to the sun or erupt into space, causing powerful coronal mass ejections which can have an impact on Earth and are therefore closely monitored by space weather forecasters.

Separating the corona and the much cooler chromosphere is the *transition region*. In this thin layer (only about 100km wide), the temperature rises abruptly from about 20'000 to 1 million Kelvin. In this region, the plasma is fully ionized (stripped of its electrons). It is still not clear why the temperature rises so much in these outermost layers of the sun (coronal heating problem). It is suspected that the solar magnetic field might be responsible for the drastic temperature changes (magnetic reconnection theory).

[15] https://solarscience.msfc.nasa.gov/feature2.shtml#Network
[16] https://solarscience.msfc.nasa.gov/feature2.shtml#Filaments
[17] https://solarscience.msfc.nasa.gov/feature2.shtml#Prominences

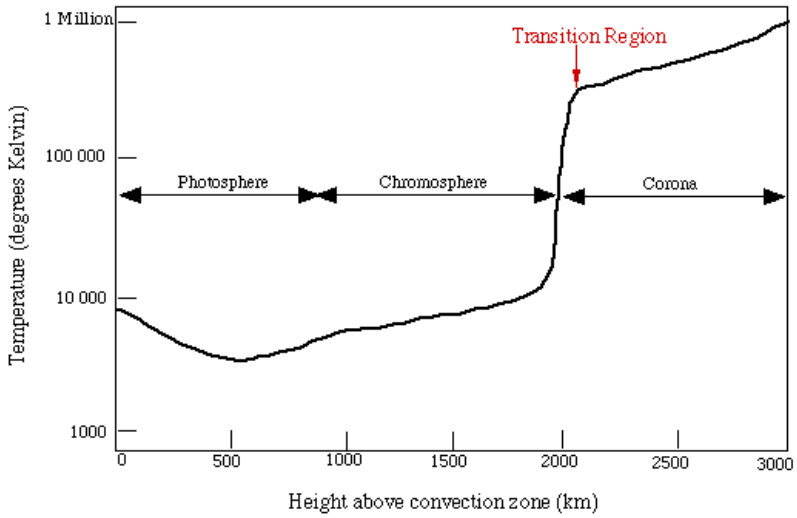


Figure 2.5: Temperatures in the Solar Atmosphere. Source: Montana State University

The outermost layer of the solar atmosphere is the *corona*. It extends up to several solar radii out from the solar surface. The temperature in the corona is 1 to 2 million Kelvin. In white light, the corona is only visible during total eclipses of the sun as a white crown surrounding the sun (see Figure 2.6). However, the corona can be observed in the X-ray and EUV spectrum, exposing several coronal features: *Helmet streamers*[18] are bright loop-like structures with long pointed peaks which connect regions of opposite magnetic polarity and usually develop over sunspots and active regions. Often prominences or filaments can be found at the base of a Helmet Streamer. *Polar plumes*[19] are long thin ray-like streamers that project outward from the limb of the solar north and south pole. They can be associated with the "open" magnetic field lines at the sun's poles. *Coronal loops*[20] are bright arc-like structures that are often associated with sunspots and active regions. They are made of hot plasma that flows along the curving lines and are associated with closed magnetic field lines that connect magnetic regions. They appear in different sizes extending up to many thousands of kilometers above the photosphere and usually change quite rapidly but can also last for days or even weeks. *Coronal holes*[21] appear as dark areas in the corona. They are cooler, less dense regions than the surrounding plasma and are associated with "open" magnetic field lines. The open magnetic field line structure allows the solar wind to escape more easily and at higher speeds, which can lead to elevated geomagnetic activity and can cause strong geomagnetic storms.[22] For this reason, they are monitored closely by space weather forecasters. Coronal holes are more common during the solar minimum but can develop at any time and can last through several solar rotations. *Coronal rain*,[23] or plasma rain, consists of giant globs of plasma that drip back from the sun's atmosphere onto the surface. It occurs because the plasma is cooled down and becomes denser caused by certain magnetic field line configurations and local heating events in the corona making it fall back onto the photosphere.

[18] https://solarscience.msfc.nasa.gov/feature3.shtml#Helmet%20Streamers
[19] https://solarscience.msfc.nasa.gov/feature3.shtml#Polar%20Plumes
[20] https://solarscience.msfc.nasa.gov/feature3.shtml#Coronal%20Loops
[21] https://solarscience.msfc.nasa.gov/feature3.shtml#Coronal%20Holes
[22] https://www.swpc.noaa.gov/phenomena/coronal-holes
[23] https://www.nasa.gov/mission_pages/sunearth/the-heliopedia

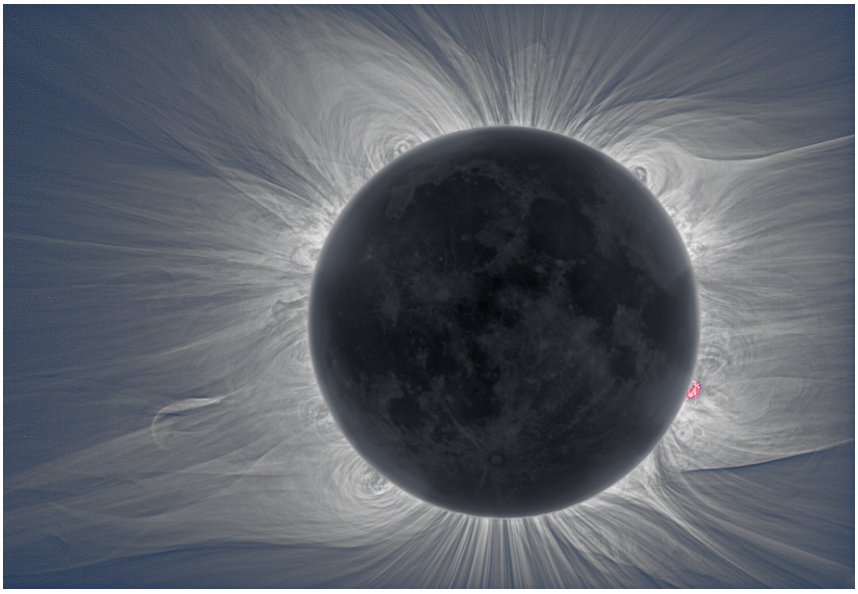


Figure 2.6: High resolution image of the corona and a red protuberance (at the lower the right) taken during the total solar eclipse in 2010 in the Pacific region. This image shows the impressively complex structure of the corona, especially overlapping loop-like streamers and the ray-like polar streamers. The very prominent polar-streams are typical for times with reduced solar activity in the 11-year cycle. Until the 1930s (invention of coronagraphs) total solar eclipses (with a maximum duration of several minutes) were the only way to observe the corona and chromosphere. Today, there are many imaging instruments on Earth but also in space, allowing an almost permanent survey of the sun in different wavelengths. Source: Eclipse Photography Home Page; Miloslav Druckmüller

The corona is the source of various solar activities such as solar wind, solar flares, and coronal mass ejections and is the layer of the sun on which this thesis focuses.

The following solar activity patterns are the most common: The *solar wind*[24] is a stream of charged particles flowing away from the sun, carrying the magnetic field out into space. Typically, the solar wind has a velocity of about 400 km/s and can be faster above coronal holes and slower above streamers.[25] When interacting with Earth's magnetic field, the solar wind can cause magnetic disturbances and geomagnetic storms. The extent to which the solar wind reaches is known as the *heliosphere* and marks the region of influence of the sun within interstellar space. *Solar flares*[26] are bursts of light and particles triggered by the release of magnetic energy. They are the most powerful explosions in the solar system travelling nearly at the speed of light and can reach Earth in under 20 minutes. Solar flares can have an associated *coronal mass ejection*. Coronal mass ejections [27] (CME) are clouds of solar plasma and embedded magnetic fields that are ejected into space as part of a solar eruption. They travel at much slower speeds than flares. The faster CMEs can reach Earth in 15-18 hours, while a slower CME can take up to several days. Besides co-occurring with flares, they can also be caused by unstable filaments or prominences that erupt into space. CMEs expand as they travel away from the sun and can collide with Earth's magnetic field, causing geomagnetic disturbances, having the potential to short-circuit satellites, damaging power grids and endangering the lives of astronauts in orbit. *Solar energetic particles*[28] (SEP) are high-energy charged particles that are accelerated by activity on the sun and usually co-occur with flares and CMEs. Because of their nature as charged particles, their movement is guided by magnetic

[24] https://www.nasa.gov/mission_pages/sunearth/the-heliopedia#Solar%20wind
[25] https://solarscience.msfc.nasa.gov/SolarWind.shtml
[26] https://www.nasa.gov/mission_pages/sunearth/the-heliopedia#Solar%20Flare
[27] https://www.nasa.gov/mission_pages/sunearth/the-heliopedia#Coronal%20Mass%20Ejection%20(CME)
[28] https://www.nasa.gov/mission_pages/sunearth/the-heliopedia#Solar%20energetic%20particles%20(SEPs)

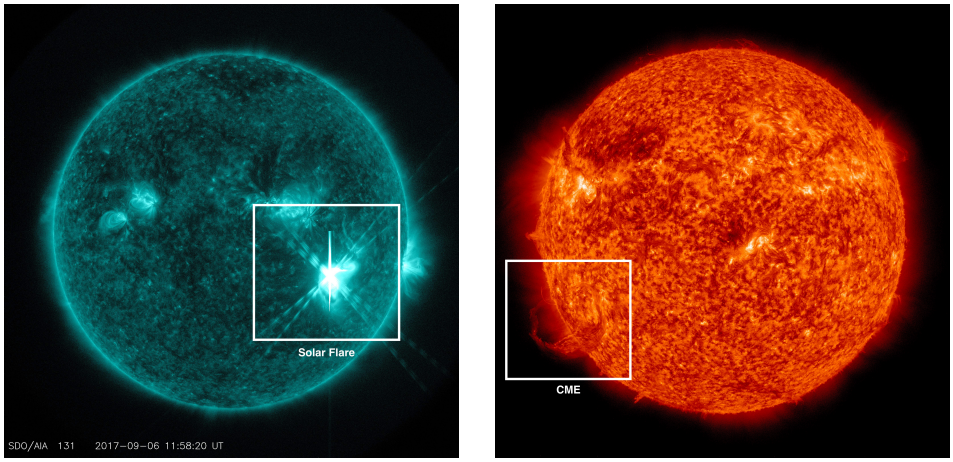


Figure 2.7: Examples of solar activity: The left image shows a strong solar flare (X9.3) captured by the AIA instrument of NASA's Solar Dynamics Observatory on Sept 6, 2017 in the 131 angstrom wavelength. The right image shows an Earth-directed coronal mass ejection on Jan 31, 2013 also taken by the Solar Dynamics Observatory in the 304 angstrom wavelength. Source NASA and NASA

fields. Similarly to CMEs they also have the potential to damage electronics onboard satellites, disrupt communications, and posing radiation hazard to astronauts.

Solar activity and variability have an imminent impact on Earth and are therefore key concerns for our society. Solar flares and coronal mass ejections have the ability to disrupt and damage critical infrastructures by disabling or damaging satellites, causing power grids to fail or disrupting communications. The strongest solar storm ever measured on Earth was the so-called "Carrington Event" in 1859. A large solar flare followed by a massive Coronal Mass Ejection (CME) led to an intensive geomagnetic storm causing failures in North American and European telegraph networks. During this period auroras could be observed as far south as Rome and Hawaii.[29] Today, such a storm would be devastating for satellites, but also electronic installations and equipment on Earth. Furthermore, the smallest change in the sun's irradiance could have a detrimental effect on our climate. This makes the study of the sun and its impact on the heliosphere an important field of research.

Solar activity changes over the course of an approximate 11-year cycle, the *solar cycle*,[30] indicated by the number and intensity of sunspots. During the solar minimum the sun is predominantly "quiet", exposing only a small number of sunpots and little to no activity. During the solar maximum, a high number of sunspots and a lot of strong activity can be observed and the activity is more likely to affect space weather. The solar cycle is driven by the sun's magnetic field, which completely flips approximately every 11 years,[31] meaning that the solar north and south pole switch places. The different activity levels between 2010 and 2020 are illustrated in Figure 2.8. Solar activity is thought to be caused by the differential rotation rate which gives rise to the formation of magnetic fields (solar dynamo.[32]) The variation in rotation speeds causes the magnetic fields to deform and shear, which in turn amplifies the magnetic fields or generates new ones.

[29] https://www.space.com/the-carrington-event
[30] https://www.space.com/solar-cycle-frequency-prediction-facts
[31] https://spaceplace.nasa.gov/solar-cycles/en
[32] https://solarscience.msfc.nasa.gov/dynamo.shtml

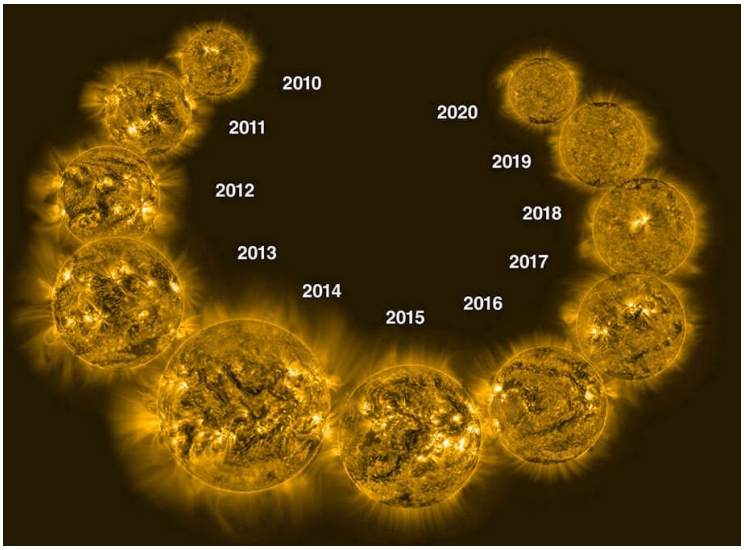


Figure 2.8: Activity levels in extreme ultraviolet light from 2010 through 2020 during solar cycle 24 observed by Europe's PROBA2 spacecraft. Source: NOAA/JPL-Caltech

There is a large array of spacecraft monitoring the sun that is strategically placed throughout the heliosphere and allows to continuously monitor solar activity.[33] The different missions are illustrated in Figure 2.9. In this work, we primarily make use of data from NASA's Solar Dynamics Observatory, which is introduced in the next section.

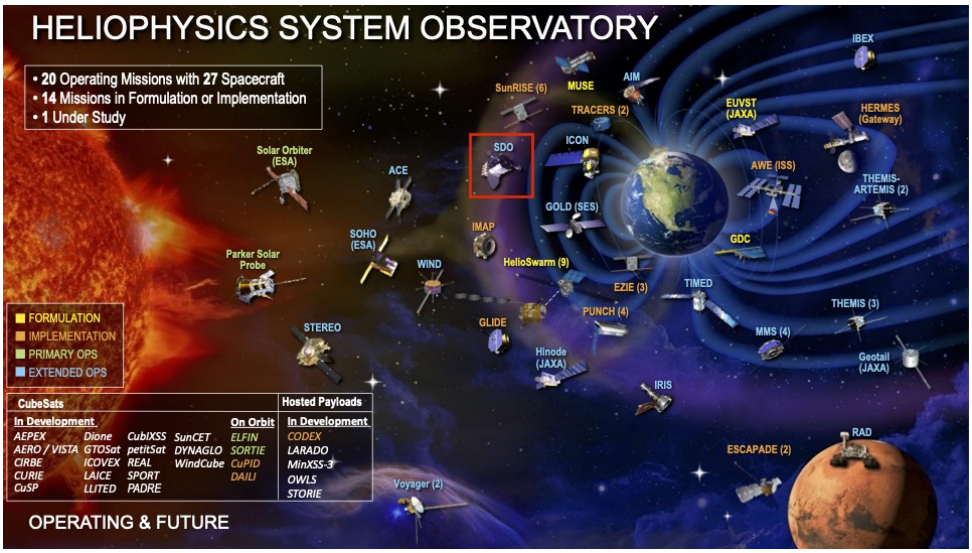


Figure 2.9: Heliospheric mission fleet chart. Source: NASA

[33] https://science.nasa.gov/missions-page?field_division_tid=5&field_phase_tid=All

2.2 The Solar Dynamics Observatory (SDO)

NASA's Solar Dynamics Observatory (SDO) [9] was launched in February 2010 and aims to understand how solar activity is created and how space weather results from that activity. The goals of the mission include fundamental questions like "*What mechanisms drive the quasi-periodic 11-year cycle of solar activity?*" or "*When will activity occur, and is it possible to make accurate and reliable forecasts of space weather and climate?*".[34] One of the main objectives is to develop the ability to reliably predict solar variations that affect life on Earth through a better understanding of the underlying physical processes that lead to solar activity.
An illustration of SDO is shown in Figure 2.10. SDO is in a geosynchronous orbit at a distance of 6.6 Earth radii and has three different instruments mounted onboard:

- **The Helioseismic and Magnetic Imager** (HMI) [10] is capturing information on the magnetic field in the solar photosphere.
- **The Atmospheric Imaging Assembly** (AIA) [11] captures high-definition full-disk images of the sun in eight different wavelength bands and is designed to study the sun's surface and atmosphere. The instrument serves as the main data source for this thesis. More details about the data acquisition process are presented in Chapter 4. AIA images have a 4096 $\times$ 4096 resolution with 0.6 arcsec pixel size (or 440 km on the sun) and are recorded at a cadence of 12 seconds. The wavelength bands can be split into two ultraviolet (UV) bands centered at 1600 and 1700 Å, seven Extreme Ultraviolet (EUV) bands centered at 94, 131, 171, 193, 211, 304 and 335 Å and one visible wavelength band centered at 4500 Å.
- **The EUV Variability Experiment** (EVE) [12] monitors the solar EUV spectral irradiance.

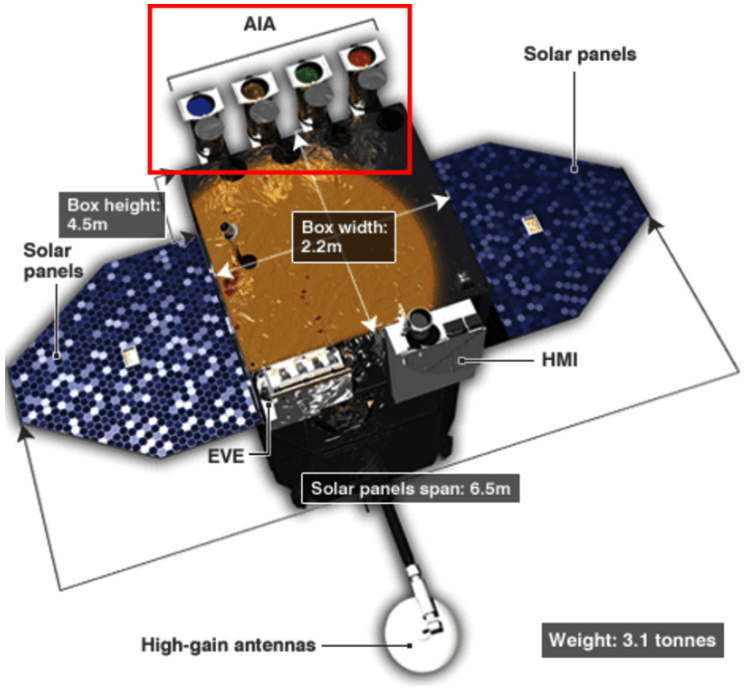


Figure 2.10: The Solar Dynamics Observatory spacecraft with the red box highlighting the AIA instrument. Source: NASA

The different wavelengths observed by SDO were chosen to emphasize specific aspects of the sun's surface or atmosphere.[35] Figure 2.11 displays the different images that are captured by the HMI and

[34] for more information about the mission goals and science refer to: https://sdo.gsfc.nasa.gov/mission/science.php
[35] https://www.nasa.gov/mission_pages/sdo/how-sdo-sees-the-sun

AIA instrument. SDO is observing the sun with an unprecedented spatial and temporal resolution in a range of spectral bands producing 1.5 TB of data per day. For example, EUV images have twice the imaging resolution of STEREO[36] and 4 times the resolution of SOHO[37]; SDO takes an EUV image every second, STEREO at best takes one image every 3 minutes and SOHO takes 1 image every 12 minutes. SDO data is available as soon as it reaches Earth and has undergone basic preprocessing and calibration, which makes it very useful for science and led to numerous publications.[38] Due to the large spatial and temporal resolution of the data, it is especially well-suited for machine-learning methods. Calibrated level 1 data from both AIA and HMI is available via the Joint Science Operations Center at Stanford University[39] (JSOC).

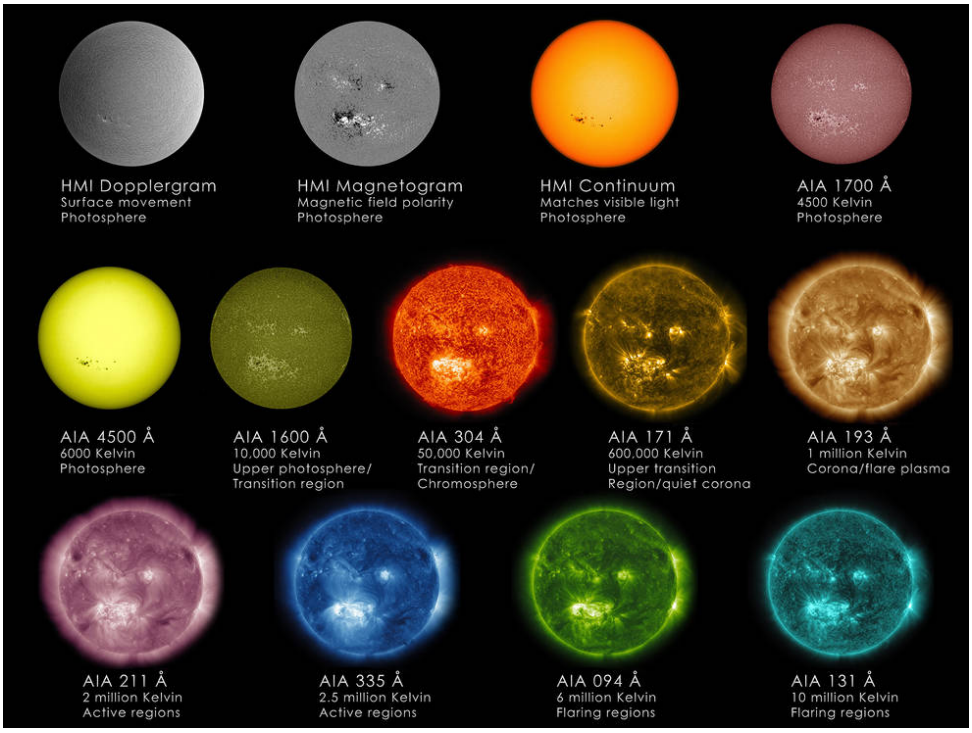


Figure 2.11: Wavelengths observed by SDO. Source: NASA

2.2.1 SDO Event Detection

Several different research groups have developed a comprehensive set of automated event recognition modules to automatically extract features from data captured by SDO. One example is the so-called *SDO Feature Finding Team* (FFT) [13] which is an international consortium of independent groups chosen by NASA that developed software modules to detect, trace and analyse numerous solar phenomena in near real-time (including flares, sigmoids, filaments, coronal dimmings, polarity inversion lines, sunspots, X-ray bright points, active regions, coronal holes, EIT waves, coronal mass ejections (CMEs), coronal oscillations, and jets, but also the emergence and evolution of magnetic elements). The FFT modules are part of the SDO Event Detection System (EDS) operated at Stanford's Joint Science Operations Center (JSOC) and the Lockheed Martin Solar Astrophysics Laboratory (LMSAL), as well as the Harvard-Smithsonian Center for Astrophysics (CfA), and NASA's Goddard Space Flight Center (GSFC). The Event Detection System analyzes the stream of data from the SDO as soon as it has arrived at JSOC and undergone some basic preprocessing and calibration. The detected events are then reported to the

[36] https://www.nasa.gov/mission_pages/stereo/spacecraft/index.html
[37] https://soho.nascom.nasa.gov/
[38] https://sdo.gsfc.nasa.gov/mission/publications/
[39] http://jsoc.stanford.edu/

Heliophysics Event Knowledgebase (HEK) [14]. As part of the FFT, Verbeeck et al. introduced the SPoCA-suite[40] [15], which is a set of algorithms to localize and characterize active regions (AR) and coronal holes (CH) observed by extreme ultraviolet (EUV) imagers such as the SDO AIA instrument. The tool is not restricted to SDO data and is also used for other EUV imagers (SOHO EIT, STEREO EUVI, and PROBA2 SWAP). The SPoCA-suite makes use of a multichannel, unsupervised, fuzzy clustering algorithm that segments EUV images into different regions according to their intensity level and provides a segmentation of active regions, quiet sun and coronal holes.

One of the main sources for solar events is the Heliophysics Event Knowledgebase (HEK), which was introduced to catalogue interesting solar events and features and make it easier for researchers to find relevant data [14]. The HEK combines metadata from different automated feature extraction methods as well as manual observations. Each report of an event contains an event class, the extraction method, information about data and associated parameters as well as a duration and bounding box that describe the event in space and time. Furthermore, each event class can specify additional required and optional attributes. Similar databases for metadata can be found in many fields where image or event metadata and labels are stored. However, compared to other fields the challenge inherent in solar data is the complexity of the different events (e.g., the temporal and spatial dimension). This means that the quality of labels for events highly depends on the extraction methods. In 2016, Schuh et al. [16] conveyed an analysis of the events reported by the FFT modules to HEK since the SDO mission was first launched in 2010. In their report, they discuss the quality and reliability of the different FFT modules and highlight potential data quality issues where the modules were not reporting properly. Their initial analysis shows that the events reported by the FFT modules compose a reasonably clean dataset. However, they note several instances where there is a notable difference between the different detection algorithms, and it is not clear how large the error margin for the different methods is (e.g., false positives and false negatives). For this reason, the events reported in HEK should only be used as a baseline and not as the ground truth for training and evaluating a machine learning model.

2.2.2 Working with SDO Data

Working with SDO data is not trivial and requires heliophysics knowledge and technical understanding of the data products at hand. The *Guide to SDO Data Analysis*[41] provides a detailed description of working with SDO data, showing how to browse, find and download data and perform common AIA data-processing tasks. Importantly, the guide contains details about the AIA data format and explains the despiking and respiking of AIA images. Several software tools have been developed to facilitate the work with solar data products. SolarSoftWare IDL (SSWIDL)[42] [17] is a set of software libraries in IDL developed by the community that can be used to find and download SDO data. Sunpy[43] [18] is a set of tools for solar data analysis written in Python, and *aiapy*[44] [19] is a Python package for analyzing calibrated EUV data from AIA.

A more detailed description of the preparation of SDO data in this study is outlined in Chapter 4.

[40] Spatial Possibilistic Clustering Algorithm
[41] https://hesperia.gsfc.nasa.gov/~bdennis/Folders/Missions/SDO/SDOD0060_N_Guide_to_SDO_Data_Analysis_sdoguide.pdf
[42] http://www.lmsal.com/solarsoft/sswdoc/index_menu.html
[43] https://sunpy.org/
[44] https://aiapy.readthedocs.io/en/latest/

2.3 Machine Learning in Heliophysics

With the increasing popularity and success of machine learning in many research fields, ML methods have in recent years also gained more traction in heliophysics. Given the wealth of data being collected on various missions and the high complexity of the data, the field seems predestined for ML methods. By far the most prominent area for ML applications is *space weather monitoring* due to the sun's imminent impact on Earth.

In this thesis, we do not convey a full survey of the state of the art of machine learning in heliophysics and instead focus on a few relevant works. For a more complete overview, the reader is referred to the books "*Machine Learning, Statistics, and Data Mining for Heliophysics*" by Bobra et al. [20], "*Machine learning techniques for space weather*" by Camporeale et al. [21], and "*Deep Learning in Solar Astronomy*" by Xu et al. [22] which provide a very helpful introduction to the state of machine learning in heliophysics. Bobra et al. have curated a collection of interactive Jupyter notebooks that exhibit a variety of use cases ranging from the prediction of coronal mass ejections to the unsupervised classification of solar wind. Camporeale et al. provide an introduction to space weather forecasting targeted at the data science community that might not be familiar with the field, introduce the most important machine learning principles that are used for space weather applications and provide a detailed overview of several machine learning applications covering a broad range of subdomains. A few examples of the applications include machine learning for solar flare forecasting, the unsupervised classification of magnetospheric particle distribution with self-organizing maps and a comparison of different machine learning methods for coronal hole detection. Xu et al. specifically focus on deep learning applied to heliophysics. In their book, they present the most prominent use cases and models that can be broadly categorized into the following deep learning tasks: classification, generation and prediction/forecasting. The use cases discussed include active region detection, EUV waves detection, solar radio spectrum classification, image deconvolution of a solar radioheliographs and solar flare forecasting as well as forecasting of the solar radio flux at 10.7cm (F10.7).

2.3.1 Machine Learning in the Heliophysics Community

With an increasing number of researchers combining expertise in heliophysics and machine learning, a community has emerged that tries to bridge the gap between the two fields and help the field advance with more automated methods [23]. The conference Machine Learning in Heliophysics (ML Helio)[45] first held in September 2019 marked the first official gathering of this community. Since then, a second version of the conference was held in March 2022, and the conference will occur every two years. With the COVID-19 pandemic preventing most physical interactions, online tools have been increasingly used to bring the community together. One example is the online forum *helionauts.org*[46] that provides a platform for the intersection between solar physics, machine learning and computer science. Furthermore, there have been other initiatives such as SpaceML[47] [24], which is a machine learning toolbox and developer community focused on building open science AI applications for space science funded by NASA's Frontier Development Lab.[48]

[45] https://ml-helio.github.io/
[46] https://helionauts.org/
[47] https://spaceml.org/
[48] https://frontierdevelopmentlab.org/

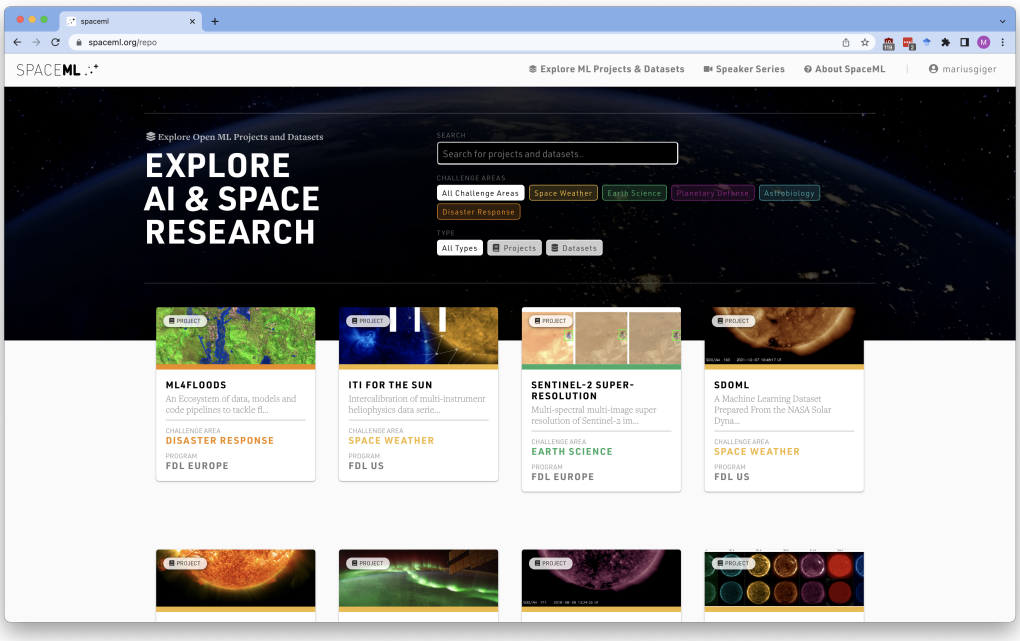


Figure 2.12: Project page of SpaceML: a machine learning toolbox and developer community building open science AI applications. Source spaceml.org

Notably, the solar physics community remains relatively conservative when it comes to the application of machine learning methods, mainly because of scepticism towards the new technologies and because of the lack of comparisons with simpler statistical methods [25]. The scepticism is mainly rooted in the fact that in contrast to machine learning models, physical models are easily interpretable, and there is a good understanding of the assumptions and limitations that come with physical models. Camporeale makes the observation that in reality, physics-based models fail at least as often as empirical models in space weather forecasting because they are usually based on assumptions that can only be checked a posteriori and have several empirical (data-derived) parametrisations [25]. The comparison with more traditional models (as well as other ML-based models) is often difficult to achieve because there is a lack of benchmark datasets, and state-of-the-art results can be obtained by using different data sources or data splits, thus leading to an unfair comparison. This is especially true for the space weather forecasting domain where the amount of published work has significantly increased in the past few years [26]. This has led to higher requirements for publication in journals, which on one hand is a positive development because the quality of published work increases but on the other hand also creates additional hurdles for researchers who do not come from this field. For example, AGU's Space Weather Journal[49] has released the following guidelines for authors focusing on space weather forecasting techniques [26]: the work must correspond to a substantial improvement over the current state-of-the-art and present this comparison; the work should include information about data preparation, including splitting of data between train, validation and test sets; the work must mention a repository with the codebase used to develop the machine learning model; and the work must present a comparison with other methods and discuss the uncertainties of the forecast. The first and last points in particular are usually difficult to achieve due to the fact that it is challenging to reproduce the results of other works without suitable benchmark datasets and therefore very time-consuming to show the improvement over another method.

In the whitepapers [27] and [28], Nita et al. present a set of findings and recommendations for machine learning in heliophysics and space weather forecasting as well as for the solar research cyberinfrastructure needs. Not surprisingly, at the forefront is reproducible research with high-quality benchmark

[49] https://agupubs.onlinelibrary.wiley.com/journal/15427390

datasets including standardized metrics and the availability of source code to facilitate reproducibility. Importantly, Nita et al. note that machine learning is not a competitor of physical modelling but should be seen as a complementary asset that is able to handle the present and future big data requirements. Furthermore, they mention the need for sufficient computational resources for big data processing that includes large-scale data storage and high throughput compute clusters because future missions are expected to produce even more data (e.g., the Daniel K. Inouye Solar Telescope [DKIST] is expected to produce 8-10TB of data per day).

Despite the many works that are published, the community is still searching for groundbreaking advances in the field enabled by machine learning [25].

2.3.2 Event Detection and Space Weather Forecasting

Detecting, tracking and forecasting solar events are key elements for space weather monitoring. The current operational event detection pipeline for SDO data still mainly uses manual or automatic extraction based on traditional image processing techniques (see Section 2.2.1). The extracted events are then reported to the heliophysics event knowledgebase (HEK). To improve the existing algorithms, reduce human intervention and satisfy current and future Big Data requirements, the application of machine learning plays an important role. To this end, several ML-based methods for detecting and tracking solar events have been proposed in recent years:

Reiss et al. [29] demonstrate the use of supervised classification to distinguish coronal holes and filaments in SDO AIA images. They extracted several first- and second-order image statistics and shape measures which were then used in different ML algorithms (support vector machine (SVM), linear support vector machine, decision tree, and random forest). They made use of a relatively small, manually labelled dataset for training and show that the different classifiers can provide good results (TSS of 0.90). Notably, by adding magnetic field information from HMI line-of-sight magnetograms, they were able to improve the performance of the different classifiers. Schuh et al. [30] propose a region-based retrieval system that extracts labelled regions of interest (active regions and coronal holes) from AIA images using 10 general-purpose grid-based image parameters (i.e., entropy, mean, std. deviation, kurtosis and others). AIA images are segmented into a 64×64 grid. For each grid cell, the 10 image parameters are calculated which are then used to classify the cell to be an active region (AR), a coronal hole (CH) or quiet sun using a set of different classification methods (naive Bayes, decision tree, support vector machine, K-nearest neighbour and random forest). They show moderate performance (60-65% accuracy) for the applied ML techniques, which provides a baseline for retrieval performance for the two different event types (AR and CH). Kempton et al. [31] propose a system to track solar phenomena over time by creating spatiotemporal trajectories. Their system uses the same image parameters as Schuh et al. [30] and is able to track active regions and coronal holes by using a statistics-based multiple hypothesis tracking algorithm (MHT) that takes outputs from FFT modules and calculates different trajectory paths, then assigns likelihoods to the different paths and selects the most likely path. By using crowdsourced human labels (ground truth), they are able to evaluate the performance of their model and show that their approach works as well or better than the event-specific SPoCA tracking module for active regions and coronal holes. Although their approach does not directly use machine learning, it demonstrates the difficulty of evaluating a method using annotations from the FFT modules, in this case only made possible by using human-assigned labels.
Notably, these and several other machine learning-based studies have relied on a set of image parameters capturing the distribution of pixel intensities and describing shape parameters but did not make use of deep learning to directly learn a feature representation from the raw data. While these image parameters have proven successful in the solar domain [32], several deep learning-based approaches have shown that they can improve the performance of these algorithms by directly learning a representation from the raw data, but at the expense of interpretability.

Kucuk et al. [33] used different deep convolutional neural network architectures (LeNet-5, CifarNet, AlexNet, and GoogLeNet) to classify five solar event types (active regions, sigmoids, coronal holes, flares and quiet sun). They used event reports from the FFT modules to compose a dataset consisting of patches with an event region and show that the proposed deep learning models provide better

accuracy in each class compared to conventional pattern recognition algorithms as well as ML methods based on statistical image features and also perform better in terms of overall accuracy.
Illarionov et al. propose a U-Net-based deep learning model to segment coronal holes [34] and indicate that their model produces better predictions of coronal holes in contrast to the semi-automatic procedure applied for coronal hole segmentation (SPoCA and CHIMERA).

Armstrong and Fletcher propose a deep learning-based approach for solar image classification trained on data from the Hinode/Solar Optical Telescope (SOT) to categorize solar features with different geometries [35]. They show that their approach is robust with respect to the resolution of images by applying their model to SDO data and achieving similar performance. Furthermore, they present an impressive overall performance for unseen data. Baek et al. [36] investigated the use of object detection methods for coronal hole, sunspot and prominence detection. They trained a single-shot multibox detector (SSD) and a faster region-based convolutional neural network (R-CNN) on a new dataset curated by experts for that purpose. They report good performance for both sunspots and coronal holes but a poorer performance for prominences.

The limiting factor in all cases is the availability of high-quality labels necessary for training sufficiently complex deep learning models. This task is likely to become infeasible for larger datasets due to time and cost constraints.

Due to their imminent impact on Earth, there is a great deal of research dedicated to forecasting solar flares. For example, Park et al. have successfully applied deep convolutional neural networks to forecast solar flares based on full-disk magnetograms and achieved at least similar performance to previous models [37]. In [38] Nishizuka et al. present Deep Flare Net, an operational deep neural network used to predict solar flares for the next 24 hours. They propose using a chronological split of the data for training and testing and show how to evaluate an operational model. Li et al. [39] propose a CNN architecture that makes use of data shuffling and cross-validation based on active region segregation. This leads to more robust and stable predictions compared to previous works and shows the importance of proper sampling of solar events.

Machine learning for space weather monitoring and forecasting constitutes a very active research field with many different research groups looking for methods that could work well and could be used in an operational scenario. One of the actively discussed topics is the use of physics-informed neural networks to include physical models in order to improve interpretability of the machine learning methods (e.g., [40]). Most of the methods introduced above rely on supervised learning and are trained based on annotations from HEK. The next section lists a few works that do not relay on labels for training.

2.4 Unsupervised Machine Learning in Heliophysics

Only a few studies have focused on unsupervised learning in heliophysics. Banda et al. propose a method for finding regions of interest in solar images and provide a content-based image-retrieval system that makes it possible to search for similar events [41]. They made use of K-means to cluster image parameters extracted from image cells. Furthermore, they introduced three evaluation metrics that determine the amount of overlap between the regions of interest and a predefined large set of ground truth labels extracted from HEK that correspond to four types of different solar events. They report an error overlap of 35% in the best case but show that they found a considerable number of new and unlabelled regions in solar images. Innocenti et al. propose an unsupervised classification method for magnetospheric regions using self-organizing maps (SOMs) to avoid relying on a labelled training set [42]. They applied their method to simulations and show that the retrieved classification results match the physical knowledge about the processes in the terrestrial magnetosphere well. Brown et al. [43] trained a sigma-variational autoencoder (VAE) to automatically extract features from SDO images. They made use of the SDO ML v1 dataset and show that a VAE can effectively learn a latent representation of the different channels for both AIA and HMI images by using the extracted features to forecast the F30 index (a proxy for solar ultra-violet irradiance data). Their model was able to reduce the original dataset to 0.19% of its

original size while maintaining good reconstructions of AIA images and still reasonable reconstructions of the z-component of the magnetic field.

At this point, previous research on the use of unsupervised methods in heliophysics is still at an early stage. Further studies are needed to gain a deeper understanding of the various methods and their applicability. Nonetheless, unsupervised or semi-supervised approaches appear to offer a reasonable way forward to overcome the lack of annotations faced by researchers in this field and to realize the full potential of the data.

2.5 Out-of-Distribution Detection Models

Generally, out-of-distribution detection methods can be categorized into three fields: statistical, proximity- and reconstruction-based methods [44]. Statistical anomaly detection methods assume that data is modelled from a certain probability distribution. If the probability of a data point being generated from this distribution is below a certain threshold, it can be marked as anomalous. Statistical methods, among which VAEs can be categorized, have received a great deal of attention because their outputs are theoretically justifiable, and the underlying distribution of the data can be modelled with deep neural networks. Proximity-based methods assume that anomalous data can be isolated from the majority of data. For example, one can apply a clustering algorithm and compute the distance to the closest cluster and the size of that cluster to decide whether a data point should be flagged as an anomaly or by using a distance metric that quantifies the distance to neighbouring data points of a given data point (e.g., K-nearest neighbor distances). Reconstruction-based methods, such as principal component analysis (PCA) or autoencoders, use dimensionality reduction to compress and reconstruct a data point and make use of the reconstruction error to decide whether a data point is anomalous. Data points with high reconstruction errors are treated as anomalous.

Variational autoencoders (VAEs) [45] and their extensions have been successfully applied to extract relevant features from high dimensional data by learning data distributions and thereby enabling anomaly detection [1, 46, 47]. An and Cho introduced the concept of reconstruction probability for anomaly detection [44] and show that using the probabilistic characteristics of a VAE as an anomaly score, can outperform autoencoder and PCA-based methods, which rely solely on the reconstruction error. They argue that including the reconstruction probability and thus considering the variance of the data distribution makes it possible to analyse the underlying cause of the anomaly. A similar approach is proposed by Zimmerer et al., who applied a context-encoding variational autoencoder (ceVAE) for medical imaging to identify and localize abnormal regions in brain MRI images and therefore enable the detection of brain tumours [1]. Zimmerer et al. show that by using the model-internal latent representation deviations, a more expressive anomaly score can be retrieved, that works on both the image and pixel-level [1, 47]. They present promising results for unsupervised anomaly segmentation and show that their model was able to outperform other state-of-the-art methods in the medical domain at the time of publication.

A set of similar studies has been performed in the medical field which have had great success in unsupervised deep learning for the task of anomaly detection [48, 49, 46, 50]. Banda et al. show that there is an accurate transfer of image parameters between the medical and solar domains [51]. This suggests that models that perform well in the medical field might also perform well in heliophysics.

Other works from different domains show similar success when applying unsupervised anomaly detection using deep neural networks. For example, in [52], Kiran et al. compare multiple unsupervised anomaly detection methods for anomaly detection in videos based on Variational Autoencoders (VAE), Generative Adversarial Networks (GANs) and Long Short Term memory networks (LSTMs) that formed the state-of-the-art at the time of the study. They show that VAEs consistently perform well or better than other methods. Wang et al. propose a self-adversarial Variational Autoencoder (adVAE) [53] for anomaly detection in tabular data in which a Gaussian transformer net is trained to synthesize anomalous but near-normal latent variables. This introduces a form of regularisation into a VAE-based outlier detection method. They used well-known tabular anomaly detection datasets to evaluate their approach and show

that their method is able to outperform state-of-the-art methods for tabular data. Li et al. [54] propose a framework called *CutPaste* that uses self-supervised learning for anomaly detection in images and can be trained on normal training data only. Their approach makes use of a data augmentation strategy that is used as a proxy task – in this case, cutting an image patch and pasting it at a random location in a large image and letting the model decide between the augmented image (anomalous) and source image (normal). They achieved new state-of-the-art performance on the MVTec anomaly detection dataset. In order to evaluate unsupervised models, many tasks provide benchmark datasets to validate the models, which means that models are examined for a certain task or domain before being applied to other fields. Due to the high complexity of solar data, it is questionable whether models can be transferred without concern. However, there is still a lack of a comprehensive benchmark dataset in heliophysics, especially for out-of-distribution detection.

2.6 Research Motivation

Current research can be considered an important first step towards a more profound understanding of the application of machine learning in heliophysics. However, most of the proposed methods are not yet ready to be used in an operational scenario (e.g., for space weather monitoring). One limiting factor is the availability of high-quality labels required to train large-scale deep learning models. An interesting research question is therefore *how to overcome the data bottleneck of scarce labels and make use of the full available datasets*. This is necessary in order to let the model learn more about the underlying processes without the need for costly annotations and thereby boost performance. Importantly, the problem of scarce labels is by far not restricted to heliophysics because other research fields (e.g., astrophysics or high energy physics) are facing similar challenges. An approach that works well in heliophysics could therefore likely be applied to other similar problems in related fields and vice-versa. A second question is *how to reduce image search space*, such as when searching for interesting solar phenomena, either between multiple sets of images or within a single image, to speed up algorithms or aid manual review of data. These questions are of central interest because the amount of data collected is constantly growing and new innovative approaches are needed to realize the full potential of the available data and with that help the field to advance.

In the last section, we saw that unsupervised out-of-distribution detection has already been successfully applied in various contexts (especially in the medical field) and can provide answers to missing labels as well as support knowledge discovery by finding anomalous or interesting instances. In order to address the above questions in the heliophysics domain, we transferred an unsupervised approach from the medical domain [1] to automatically learn a feature representation from solar EUV images and find and localize out-of-distribution images and pixels. By doing so, we hope to demonstrate a way to overcome the need for labels and provide a tool which is able to improve space weather monitoring and might help to increase the understanding of the sun by reducing the search space and thereby finding interesting or extreme phenomena for solar physicists. To our knowledge, no prior studies have examined the use of deep learning-based out-of-distribution detection for solar EUV images. This method is introduced in more detail in the next chapter.

Chapter 3

An Out-of-Distribution Detection Method for Solar EUV Images

The high dimensional structure of images is on one hand determined by the pixel resolution (height and width multiplied by the number of channels) and on the other hand by the size and orientation of relevant information within an image. It is therefore necessary to extract a lower-dimensional, invariant feature representation before applying the actual task (e.g., classification or segmentation). In Section 2.3.2, we have seen several works that have extracted statistical and/or structural image descriptors from solar images which are then fed to a machine learning model. This process can provide a useful baseline but is not truly data-driven as it requires a great deal of domain knowledge and fine-tuning and is usually not very generalisable. For these reasons most of the more recent works make use of deep learning.

Deep learning techniques typically embed both the extraction of relevant features and the task at hand into a single end-to-end pipeline, learning a more adequate feature representation and thereby outperforming more traditional manual feature engineering-based methods. However, training DL models requires a large amount of annotated data which is not available in our case because we cannot entirely rely on existing annotations (i.e., HEK annotations). In Section 2.5, we have introduced a series of recent studies that show how to overcome the data bottleneck of scarce annotations by making use of *unsupervised deep learning*. This assumption builds the foundation for this thesis.

In this chapter, we introduce a purely *unsupervised* approach based on a variational autoencoder (VAE) that directly learns a lower-dimensional representation from solar EUV images without requiring any labels. By making use of a generative model, we hope to gain a better understanding of the model-internal latent representations. The method studied in this research is called *context-encoding variational autoencoder* (ceVAE) [1] and is borrowed from the medical domain, where it was successfully applied to detect and localize abnormal regions in medical images (MRI) and thereby identify brain tumours. Similarly, our goal is to extract semantically relevant information from solar EUV images and thereby address the need for a more data-driven approach for detecting "interesting" phenomena and events and at the same time learn a useful representation of the sun. Such a method could potentially be used in downstream applications, such as flare prediction or event tracking. Notably, we are trying to understand whether such an approach could work at all and therefore pursue a bottom-up strategy by making one approach work and leaving a comparison of alternative deep learning paradigms to future work (e.g., Generative Adversarial Networks or Normalizing Flows).

3.1 Learning a Useful Representation

The following sections introduce the different building blocks for the *context-encoding variational autoencoder* and show how to learn a useful representation from images using *unsupervised* deep learning.

3.1.1 Autoencoders

Autoencoders (AE)[1] are neural networks, that attempt to reconstruct an input x from a compressed representation z. Autoencoders consist of two parts, an encoder f and a decoder g. The encoder transforms an input x into a lower-dimensional representation z, and can be expressed as $z = f(x)$. The decoder attempts to reconstruct the input x by decoding the encoded representation z, creating output $\hat{x}$ and can be expressed as $\hat{x} = g(z)$. A schematic illustration of an Autoencoder is shown in Figure 3.1:

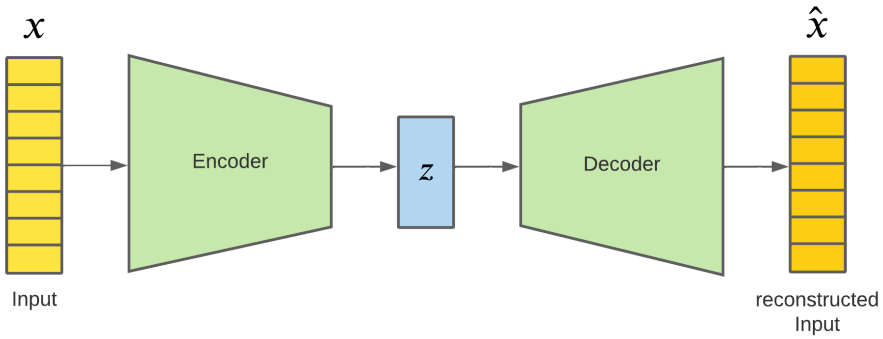


Figure 3.1: Autoencoder (AE)

Autoencoders are an unsupervised learning technique and therefore do not require any target labels at training time. Instead, they make use of the input data to optimize the weights of the encoder and decoder. Autoencoders are designed to enforce a data bottleneck and thus learn a lower-dimensional representation of the input data (given that the input can be compressed). Ballard et al. [55] first introduced autoencoders in 1987 as a pretraining mechanism for artificial neural networks (ANN), but autoencoders have since been used for many different tasks such as dimensionality reduction, compression and feature learning. Autoencoders are trained by minimizing the reconstruction loss, L_{AE}, which is defined as the reconstruction error between input x and its reconstruction $\hat{x}$ (Eq. 3.1). Common choices for the reconstruction error, L_{rec}, are the mean-squared error (MSE), $L_{MSE}(x, \hat{x}) = ||x - \hat{x}||^2$ or the mean-absolute error (MAE), $L_{MAE}(x, \hat{x}) = |\hat{x} - x|$.

$$L_{AE} = L_{rec}(x, g(f(x))) \tag{3.1}$$

Standard autoencoders encode inputs as points, only capturing the variation needed to reconstruct training samples. This does not yet force the hidden representation to capture information.[2] To circumvent this problem, a variety of different autoencoder architectures have been proposed to accomplish a set of different tasks and extend the capabilities of the models beyond just copying inputs to outputs. Sparse autoencoders [56] include a sparsity penalty that forces the autoencoder to be more sparse (i.e., fewer neurons are activated) and thereby focus on statistically relevant features of the source dataset. Contractive autoencoders [57] include a penalty that encourages the derivatives of the encoder to be as

[1] The reader is referred to Chapter 14 of The Deep Learning Book by Goodfellow et al. [6] for a more detailed introduction to autoencoders.

[2] Refer to Chapter 14.6 of The Deep Learning Book about learning manifolds [6].

small as possible, thereby making the encoder less susceptible to small perturbations of the input. This results in more robust encodings and forces the autoencoder to capture information about the training distribution. Denoising autoencoders [58] receive a perturbed data point $\tilde{x}$ that has been subject to noise as input and are trained to reconstruct the original uncorrupted data point as output. This regularization prevents the model from learning a useless identity function and leads to models that are more robust and invariant to noise in the input data. It has been shown that denoising autoencoders actually implicitly learn the structure of $p_{data}(x)$ [59], which is a useful property that emerges as a byproduct from minimizing the reconstruction error. Variational autoencoders (VAE) [45] do not map from input to output deterministically, rather they learn a stochastic mapping by encoding distributions instead of points. This class of autoencoders is introduced in more detail in the next section.

3.1.2 Variational Autoencoders

Variational autoencoders (VAE) are able to capture information about the data-generation process and therefore about the distribution of the data, or as Kingma and Welling explain [60]:

"[The] quest for disentangled, semantically meaningful, statistically independent and causal factors of variation in data is generally known as unsupervised representation learning, and the variational autoencoder (VAE) has been extensively employed for that purpose."

Kingma and Welling first introduced variational autoencoders in 2014 [45] as a method for jointly learning deep latent-variable models (DLVM) and corresponding inference models using stochastic gradient descent (SGD).[3] At the same time Rezende et al. [61] proposed a similar framework that formulates very similar ideas and can be seen as a complementary work.

Variational autoencoders can be understood as an extension of normal autoencoders which encode inputs as distributions rather than points. For this reason, by definition, they fall into the category of *generative models*. As opposed to discriminative models (such as standard classifiers or regressors), for which a predictor is trained given the observations, a generative model tries to solve the more general problem of learning a joint distribution over all the variables and thus simulating how data is generated in the real world [60]. In the case of a *latent variable* model (such as the VAE) in which a latent variable z causes the observation x, the variables cannot be observed directly. Instead, the inference model (also called the encoder or recognition model) approximates the posterior distribution of the latent variables. The generative model (also called a decoder, generator or deep latent-variable model) then samples from the posterior distribution in order to obtain a data point. These two models coupled together form a VAE (illustrated in Figure 3.2).

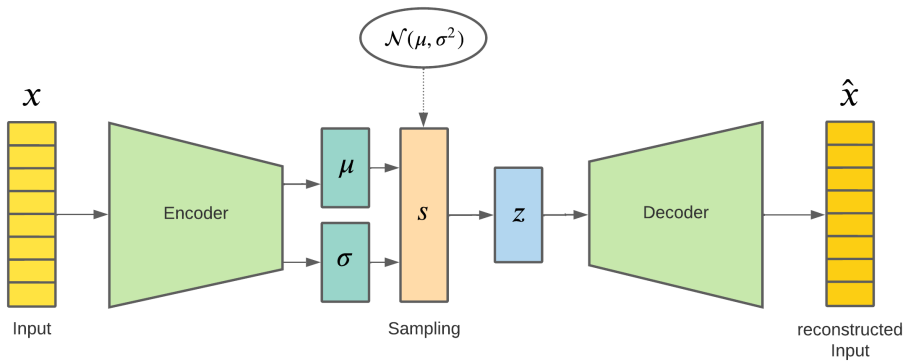


Figure 3.2: Variational autoencoder (VAE)

[3] Also refer to Chapter 20 of "*The Deep Learning Book*" by Goodfellow et al. [6] and "*An Introduction to Variational Autoencoders*" by Kingma and Welling [60].

The encoder and decoder models are independently parametrised but jointly optimized by performing stochastic gradient ascent on the *variational lower bound* $\mathcal{L}$, which is also often referred to as the *evidence lower bound* (ELBO). By applying the so-called *reparameterisation trick*, which is a change of variables so that the model becomes differentiable, the variational lower bound yields an estimator that can be directly optimized using standard stochastic gradient methods.[4] In this way, VAEs are able to approximate the underlying data distribution. The evidence lower bound (or variational lower bound) is defined as follows:

$$\begin{aligned}\mathcal{L}(q) &= \mathbb{E}_{z\sim q(z|x)}[\log p(z,x)] + \mathcal{H}(q(z|x)) \\ &= \mathbb{E}_{z\sim q(z|x)}[\log p(x|z)] - D_{KL}(q(z|x)||p(z)) \\ &\leqslant \log p(x)\end{aligned} \tag{3.2}$$

In this formula, $p(z)$ is the prior distribution of the latent variable z, $q(z|x)$ is the approximate inference model and $p(x|z)$ is the generative model. By maximizing the evidence lower bound $L(q)$, the probability distribution q approximates the true data distribution and can be used as a proxy for the probability estimate (likelihood) of a data sample and therefore also as an anomaly score.

In practice, VAEs use diagonal normal distributions for $q(z|x)$ and $p(x|z)$, where $q(z|x)$ is parametrized by the encoder neural network (f_μ and f_σ) and $p(x|z)$ is parametrized by the decoder neural network g. The encoder encodes input x to a learned feature representation z by computing mean μ and standard deviation σ for this data point and sampling from the learnt distribution q in order to obtain the latent vector z. The decoder attempts to reconstruct the original input from the resulting representation.

Varational Autoencoders are trained by maximizing the evidence lower bound. More specifically, the loss function for training a VAE includes two loss terms. The first is the Kullback-Leibler (KL) divergence loss L_{KL} which quantifies the divergence between the true and approximated posterior w.r.t. to data distribution. The second is the reconstruction loss L_{rec} which captures the difference between model input and output (Eq. 3.3). A common choice for the reconstruction loss is the mean-squared error (MSE).

$$L_{VAE} = L_{KL}(f_\mu(x), f_\sigma(x)^2)) + L_{rec_{VAE}}(x, g(z)) \tag{3.3}$$

where $z \sim \mathcal{N}(f_\mu(x), f_\sigma(x)^2)$ and f_μ, f_σ and g are neural networks.

When applying VAEs to images, both encoder and decoder are composed of convolutional neural networks (CNN). Figure 3.3 illustrates a convolutional variational autoencoder which uses images as input and output.

[4] Refer to Section 2.4 of "An Introduction to Variational Autoencoders" [60]

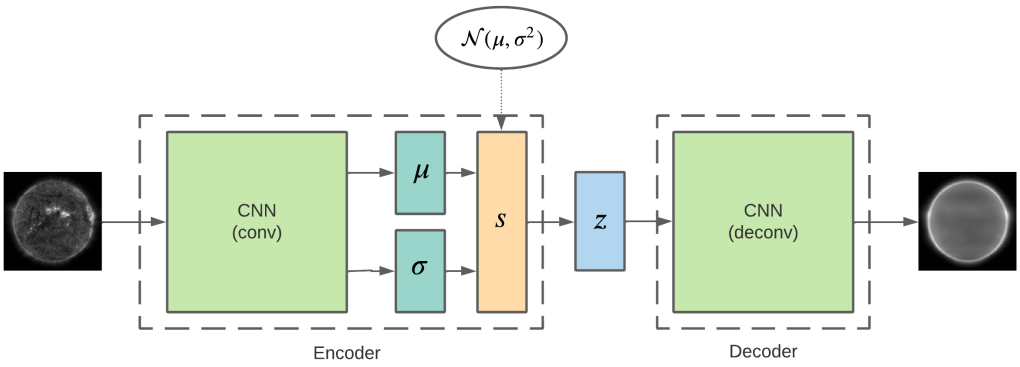


Figure 3.3: Convolutional Variational Auto Encoder (VAE)

VAEs impose an implicit form of regularisation by forcing the model to learn something about the data-generating process and are therefore often able to generalise well. However, one major drawback of VAEs trained on images is that they tend to generate somewhat blurry outputs which could be an intrinsic effect of maximum likelihood [6]. In recent years, other generative modelling paradigms have gained considerable attention with generative adversarial networks (GANs) [62] at the forefront. Compared to VAEs, GANs can generate images of high subjective perceptual quality; however, GANs tend to lack full support over the data (this problem is also known as mode collapse[5]). GANs and other hybrid or alternative architectures are not discussed in this thesis and a comparison to the methods discussed here is left to future work. The next section introduces the main method studied in this thesis.

3.1.3 Context-Encoding Variational Autoencoder

A context encoder (CE) is a special type of denoising autoencoders trained by masking local patches of the inputs and letting the model reconstruct the masked-out part. This is commonly referred to as inpainting, and it has proved to be an effective regularization technique that, in addition to improving generalisability, also captures semantic information from the input by learning the semantics of visual structures [63].

In 2018, Zimmerer et al. introduced a method combining a context encoder and a variational autoencoder, which is the so-called *context-encoding variational autoencoder* (ceVAE) [1]. Zimmerer et al. used the ceVAE model to identify and localise abnormal regions in medical images and thereby detect brain tumours. In addition to proposing a new architecture, the authors made use of the model-internal latent representations to compute anomaly scores, arriving at a more expressive scoring method which includes both reconstruction error and the deviations from the learned distribution (KL term). The different scoring modes are introduced in more detail in the next section.

The ceVAE model consists of two branches: a CE branch and a VAE branch. The model uses two fully convolutional encoders f_μ, f_σ, which share most of their weights and a decoder g which is used for both branches. A schematic illustration of the model is provided in Figure 3.4:

[5]https://developers.google.com/machine-learning/gan/problems#mode-collapse

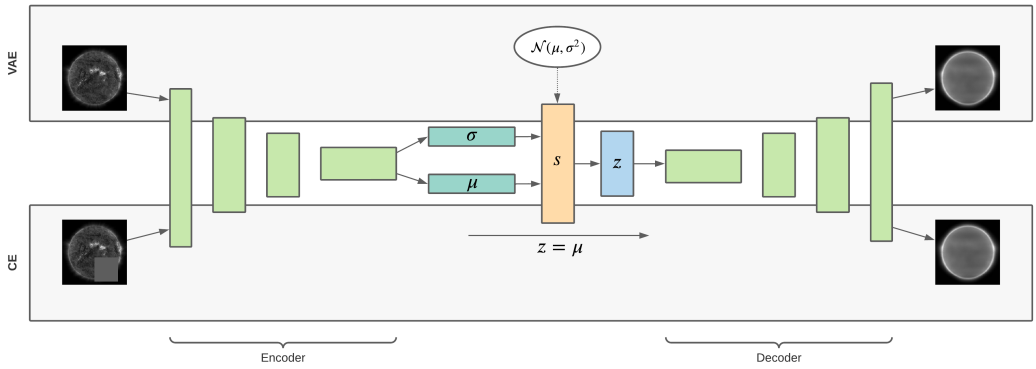


Figure 3.4: Context-Encoding Variational Auto Encoder (ceVAE)

The CE branch only uses the mean encoder f_μ to encode a data point x because it does not involve sampling from a distribution in order to generate the latent code z; therefore, $z = \mu$. Inputs to the CE branch are subject to noise by masking out rectangular patches (randomly sized and positioned with a random noise value within the limits of the input). The CE branch is trained to reconstruct the perturbed inputs $\tilde{x}$ by using f_μ as the encoder and g as the decoder. The loss function of the ceVAE model is defined in eq. 3.4. It extends the VAE loss by an additional regularisation term $L_{rec_{CE}}$ which is the loss of the CE branch computed as the reconstruction error between the perturbed input $\tilde{x}$ and its reconstruction $\hat{x}$.

$$L_{ceVAE} = L_{KL}(f_\mu(x), f_\sigma(x)^2)) + L_{rec_{VAE}}(x, g(z)) + L_{rec_{CE}}(x, g(f_\mu(\tilde{x})) \tag{3.4}$$

As highlighted by Zimmerer et al. [1], the denoising task is expected to lead to reconstruction errors that approximate the local derivative of the log-density with respect to the input. This should lead to more expressive model-internal variations by forcing the model to learn something about the semantics of the visual structures of the input. This is based on a concept for denoising autoencoders introduced by Alain et al. [59]. To assess whether an input deviates from normality, the deviations of the latent representation from its mean are used. This is the task of the VAE part of the model which learns to generate data and therefore incorporates the underlying data distribution. The encoders f_μ and f_σ, the decoder g and a standard Gaussian prior $p(z)$ yield a comparable per-sample-likelihood estimate approximated by the evidence lower bound (ELBO). This estimate is then used as a comparable anomaly score introduced in the next section.

3.2 Anomaly Detection

3.2.1 Out-of-Distribution Detection with the ceVAE Model

The ceVAE model makes it possible to calculate anomaly scores on both the image and pixel-level. Consequently, it is possible to first find images that are considered out-of-distribution and then localise the abnormal parts within those images.

Sample-wise scoring The sample-wise, or in our case image-level, anomaly score of an input x, is calculated by evaluating the evidence lower bound (ELBO; Equation 3.5). Thus, the score quantifies the log-likelihood of input x and is farther away from 0 for inputs that are less likely, or out-of-distribution, and closer to 0 for inputs that are deemed normal.

$$\log p(x) \approx L_{KL}(x) + L_{rec_{VAE}}(x, g(f(x))) \tag{3.5}$$

Pixel-wise scoring Usually, VAE-based anomaly detection models generate an anomaly segmentation map by applying a threshold to the pixel-wise reconstruction error of the model, therefore entirely discarding the KL-term and with that, potentially useful information contained in the latent feature representation. In their subsequent work Zimmerer et al. introduce different scoring-mechanisms that also include the KL-term for anomaly scoring [47]. They show that the anomaly detection performance for medical images can be improved by combining the reconstruction error with the backpropagated KL-term for a VAE-based model.

The different strategies for calculating pixel-wise scores are summarised as follows:

- **Rec-error**: This is the traditional reconstruction-based score, only including the pixel-wise reconstruction error.
- **ELBO-grad**: This score uses the derivative of the ELBO $\mathcal{L}$ with respect to the input, yielding a pixel-wise vector pointing towards a data sample with a lower $\mathcal{L}$ and builds on the assumption that the variational lower bound allows for a sufficient approximation of the true data distribution, which is presumably always the case if the model has enough expressiveness.
- **KL-grad**: This score is computed by differentiating the KL-term with respect to the input.
- **Rec-Grad**: This score is computed by differentiating the reconstruction-term of $\mathcal{L}$ with respect to the input.
- **Combi**: This score is computed as the combination of the reconstruction error and the KL-grad score and is expected to be less prone to noise artifacts. The two scores are combined by multiplication because they differ by several orders of magnitude.

It is possible to localise anomalous regions in an image x by calculating the pixel-wise anomaly score. When using the "combi" strategy, this is formulated as the combination of the density-based and reconstruction-based anomaly score. The reconstruction-based score is given by the reconstruction error, which according to the authors, is expected to tend towards the derivative of the log-density with respect to the input due to the denoising task of the CE-branch. When the mean absolute error (MAE) is used as the reconstruction error, the reconstruction-based score is calculated as the absolute pixel-wise difference. The density-based score is derived by differentiating the KL-term with respect to the input and thus computing the latent variable deviations from the prior. Notably, this is slightly different from the initially proposed score-mode in [1] which would propagate the ELBO back onto the input. The combination of the scores can be achieved by an element-wise function h (e.g., a pixel-wise multiplication as in Equation (3.6)):

$$h\left(|x - g(f(x))|, \frac{\delta(L_{KL}(x))}{\delta x}\right) \tag{3.6}$$

3.2.2 Out-of-Distribution Detection with a Naive Baseline Model

To understand whether the studied model architecture learns anything about solar phenomena in EUV images and is able to effectively label out-of-distribution regions, we compare the results with a naive baseline model that is purely threshold-based and does not involve any machine learning. We would like to prevent the ML-based model from learning a mean-pixel intensity and labelling all bright regions as out-of-distribution. For this reason, the baseline model was constructed to label regions with high intensity as anomalous.

Sample-wise scoring The sample-wise anomaly score for an image is defined by the mean pixel intensity:

$$score_{sample} = mean(x) \tag{3.7}$$

Pixel-wise scoring For the pixel-wise anomaly score, we define a threshold as the mean pixel intensity plus 2 times the standard deviation of the pixel intensity. The pixel-score is defined as 1 if the pixel value is greater than the threshold and 0 otherwise.

$$\begin{aligned} threshold &= mean(x) + 2 * std(x) \\ score_{pixel} &= \begin{cases} 1 & \text{if pixel value} > threshold \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{3.8}$$

In Chapter 6, we analyse the different outputs of the models.

Chapter 4

Preparing Solar Data for Machine Learning

The Solar Dynamics Observatory has enabled a completely new approach to solar data analysis by providing a high temporal and spatial resolution of structured scientific data poised for automated data analysis and machine learning. Prior missions have reported lower resolution data (e.g., SOHO and STEREO) or relied on so-called *field-of-view (FOV)* observations. These observations are based on a planned schedule which would monitor small parts of the solar disk for a certain period of time and would often allow a specific set of configurations for an observation (e.g., the sampling algorithm for IRIS). This makes observations inherently different from each other and therefore significantly increases the effort necessary for automated analyses.[1]

While calibrated level 1 data[2] from the AIA and HMI instruments are easily accessible from the Joint Science Operations Center (JSOC) at Stanford University, the preprocessing of this data for scientific analysis often requires specialized heliophysics and instrument-specific knowledge before it can be used in a machine learning model. To reduce the barriers of entry for nonheliophysics machine learners, several research groups have published curated datasets that take care of domain-specific nuances (e.g., that images must be spatially and temporally aligned) [66, 67]. In the following sections, we will briefly highlight the challenges observed with solar data, introduce the data sources that are used throughout this study and highlight the different preprocessing steps that we have applied.

4.1 Data Challenges

Solar data is inherently complex, which makes working with the data a challenging task, especially for nondomain experts. A few factors contributing to the complexity are summarized as follows:

- **Spatiotemporality:** Most solar activity is local to a small part of the solar disk and evolves over time as the sun rotates. However, the rotation of the sun is not constant because the rotation velocity varies depending on the distance from the solar equator (differential rotation). Additionally, the sun has a spherical shape which results in shearing at the limb, introducing a squeezing effect for observations closer to the limb.
- **Dynamics at different timescales:** Solar phenomena occur at significantly different timescales. Short-lived phenomena, such as UV bursts, solar flares, or coronal mass ejections (CME) occur within minutes to hours. Longer-term phenomena such as coronal holes or active regions have a lifespan of days up to several weeks [68].

[1] An example of the difficulties observed when working with observations from IRIS can be found in [64] and [65].
[2] http://jsoc.stanford.edu/jsocwiki/Processing

- **Solar cycle:** Throughout the solar cycle, the activity levels of the sun change significantly, leading to nonstationary data. This requires careful selection when sampling data for machine learning models.
- **Data availability:** The available data is often limited by the existence of the recording instruments. For example, SDO has gathered data for more than 11 years at the time of writing, which is slightly more than one solar cycle. This limits the capability to extrapolate to future solar cycles unless additional data sources from past solar cycles are taken into consideration (e.g., data from SOHO, the predecessor of SDO).
- **Large data volume:** The data volume gathered by the different missions observing the sun is massive. For instance, SDO reports roughly 1.5 TB every day, constituting a total of over 18 PB of data gathered throughout the mission at the time of writing this thesis. Furthermore, the data usage per data point is high (i.e., different images at different wavelengths for a given point in time). Depending on the use case, a great deal of the data is not interesting which requires appropriate strategies for downsampling and filtering the data. Future missions will report even larger volumes of data (e.g. the Daniel K. Inouye Solar Telescope (DKIST) is expected to produce 8-10TB of data per day).
- **Scarce labels:** Much of the data is not or inconsistently labelled (e.g., labels per pixel, spectra, active region, full solar disk), which makes supervised training challenging and requires large efforts to produce clean datasets.
- **Rare events:** Interesting solar phenomena, such as strong solar flares or large coronal mass ejections, are rare and therefore induce a natural bias when training machine learning models. Certain use cases, such as space weather prediction specifically flare prediction, therefore require special attention to account for heavily unbalanced data.
- **Instrument zoo:** There is a multitude of instruments with different research goals monitoring different characteristics of the sun (see Figure 2.9). Knowing which data is relevant, aligning the data between different instruments and thereby improving the predictive qualities of machine learning models is nontrivial.
- **Instrument technicalities:** Understanding the different instruments and possible manipulations required to make the data useful for research and machine learning is difficult. For example, AIA observations have heterogeneous exposure times which requires adjustment of the observations for the exposure time. Similarly, the raw data is processed on the ground in such a way that spikes arising from energetic particles hitting the instrument are filtered out. When considering short-lived phenomena, respiking the data is advised to restore the original intensities [69]. A third example is the heterogeneity caused by SDO's orbit around the sun, which makes it necessary to rescale the image based on the distance from SDO to the sun at a given point in time.
- **Instrument noise:** Instruments can always produce artefacts due to manoeuvres, calibration, malfunctioning or due to the fact that they cannot be protected against energetic particles hitting the electronics in space. For example, SDO level 1.5 data retrieved from JSOC contains corrupted images, that are not intended for scientific analysis and need to be removed from the dataset. Corrupt image data usually stems from data that is reported during calibration manoeuvres, eclipse periods, or occasional instrument anomalies. Such data is flagged with a nonzero value of the QUALITY keyword in the FITS header for both the AIA and HMI instruments.
- **Instrument degradation:** Instruments such as EUV imagers suffer from degradation of sensitivity over time, which must be compensated for with calibration. A set of images illustrating the instrument degradation affecting the different AIA channels is shown in Figure 4.1.

This list summarizes some of the main challenges that are observed when working with solar data and, depending on the task, the list might even be extended. Importantly, before applying models, machine learning practitioners should be aware that these factors might influence and skew the results. For example, not compensating for the instrument degradation might have a negative impact on the model

performance, and models will likely even learn to emulate the nonphysical properties caused by the instruments restricting predictive capabilities.

To reduce the time required for data preparation and compensate for missing instrument and domain knowledge, we used the *SDO Machine Learning Dataset v2*, which is introduced in the next section. For our research, the main challenge was the scarcity of labels to evaluate the proposed approach. This made it difficult to compare different baselines and make a statement about the effectiveness of different model architectures. We propose strategies for still obtaining an understanding of how the model performs in Chapter 6. Furthermore, the different activity levels throughout the solar cycle play a significant role for out-of-distribution models because the model should learn the underlying distribution of the data. For this reason, it is important that the training set contains data ranging from the solar minimum to the solar maximum. To ensure this, we introduce a data splitting strategy that is presented in Section 4.3 of this chapter.

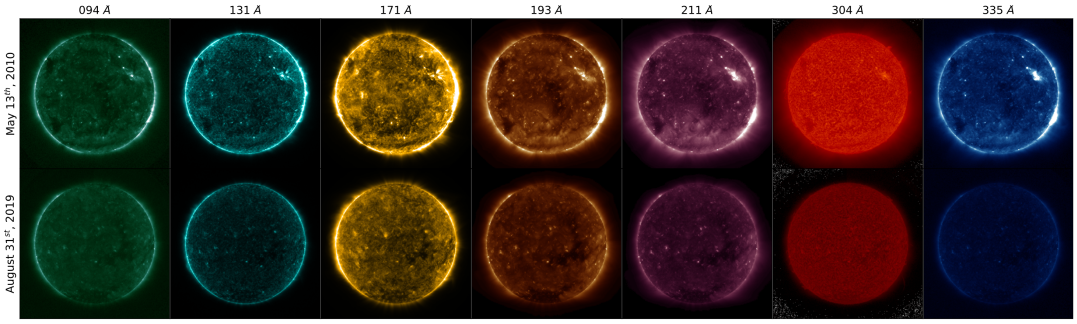


Figure 4.1: Degradation affecting AIA channels. AIA channels from left to right: 94, 131, 171, 193, 211, 304, and 335 Å. Top row: images from 13 May 2010. Bottom row: images from 31 August 2019, without correction for degradation. The 304 Å channel images are in log-scale because the degradation is severe. Source [70]

4.2 Data Sources

In this study, we made use of the following data sources: We used the *SDO Machine Learning Dataset* [67] to train, validate and test different models. Additionally, we used *GOES X-ray sensor* measurements and *HEK event recordings* as part of the validation pipeline to interpret the model predictions.

SDO Machine Learning (SDO ML) Dataset The SDO ML dataset (v1) was originally published by Galvez et al. [67] and is hosted on the Stanford Digital Repository[3] in Numpy's compressed array format (.npz). The authors have applied various instrumental corrections, removed corrupt samples, down-sampled the data to manageable spatial and temporal resolutions and synchronized the observations of the different instruments spatially and temporally. For more information, refer to Table 4.1 for the full preprocessing steps. The resulting dataset covers 2011 through 2018, totalling 6.5 TB of data. In late 2021, the authors released a second version of the dataset (v2)[4] in which the full dataset has been converted to the cloud-friendly Zarr format. Furthermore, SDO/AIA data was updated to account for a change in calibration after 2019. The new dataset now also includes FITS header keyword information, such as observation time and exposure time, as well as a process for continually updating the data until the present day. The second version of the SDO ML dataset is 6.3TB in size the slightly smaller size stems from the compression used to store Zarr arrays (in this case, zstd level 5 compression[5]).

[3] https://purl.stanford.edu/nk828sc2920
[4] https://sdoml.github.io/
[5] https://github.com/SDOML/SDOMLv2/blob/8bd1175f3850af6b1c945656ac7da76db77221ca/aia_fits_to_zarr.py#L76

Curated Image Parameter Dataset Ahmadzadeh et al. [66] published the Curated Image Parameter Dataset in 2019. It consists of images from the SDO/AIA instrument for the period of January 2011 through the current date. The images have a cadence of 6 minutes and include the nine wavelength channels (94, 131, 171, 193, 211, 304, 335, 1600, and 1700 Å). In their work, Ahmadzadeh et al. show that using the JPEG2000 file format results in a significant reduction in size of the level 1.5 FITS data without a significant loss of information contained within the data. We used this dataset for fast random access to images which is not possible with the SDO ML v2 dataset.

Step	Description
1	The raw images are rotated and resized onto a common grid such that the pixel size is 0.600 arcsec and the solar disk is aligned with the solar west and north directions.
2	Images are rebinned by averaging neighbouring 4×4 pixel blocks such that the resulting image has a size of 1024 × 1024 pixels. The resulting images are processed at a 2-min cadence.
3	The images normalised by exposure time, corrected for instrument degradation and corrections for elliptical orbital variation are applied.
4	Images are downsampled by summing in local blocks which emulates the expected observation of a lower-resolution instrument. The final interpolated images have a resolution of 512×512 pixels with a pixel size of 4.8 arcsec.

Table 4.1: AIA preprocessing steps for the SDO ML dataset, refer to [67]

GOES X-Ray Sensor (XRS) measurements A series of GOES satellites have captured soft X-ray measurements in two energy ranges since 1975 (XRSA 0.5-4 and XRSB 1-8 Å). The two satellites GOES 16 and 17 are the latest in line. The flux levels in the GOES 1-8 angstrom channel are used to report flares and determine their magnitude. Science-quality XRS datasets are produced by NOAA's National Center for Environmental Information (NCEI) and are available via Sunpy.[6] Notably, this data has been reprocessed from the start of the mission to the present date and incorporates retrospective fixes for issues and outages.[7] For this study, we downloaded GOES data for 2010 through 2020 and stored the resulting time series as a Dask dataset in the columnar file format Parquet.[8] The data is illustrated in Figure 4.2.

[6] https://docs.sunpy.org/en/v4.0.1/generated/gallery/acquiring_data/goes_xrs_example.html

[7] https://data.ngdc.noaa.gov/platforms/solar-space-observing-satellites/goes/goes16/l2/docs/GOES-R_XRS_L2_Data_Users_Guide.pdf

[8] https://docs.dask.org/en/stable/dataframe-parquet.html

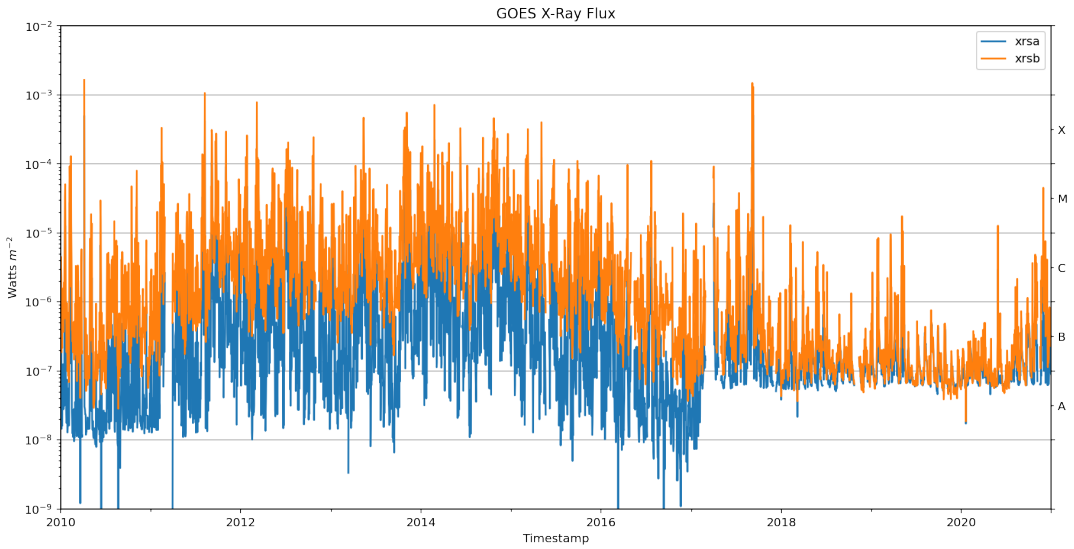


Figure 4.2: The GOES XRS time series for the years 2010 to 2020

HEK Events The Heliophysics Event Knowledgebase (HEK) was introduced to help researchers to efficiently find data relevant to their topics of interest [14]. It is the main catalogue for "interesting" solar events and features and helps to efficiently retrieve metadata about events and their observations. The events are reported using semi-automatic feature extraction described in Section 2.2.1. Despite the limitations of HEK, especially in terms of data quality, HEK events are the easiest way to obtain some form of annotations for SDO data. In this study, HEK annotations are cross-correlated with the model outputs of the unsupervised approach to obtain a better understanding of the model predictions (see Chapter 6).

An example of active region events recorded in HEK in July 2014 is illustrated in Figure 4.3. The process of aligning polygons that define bounding boxes with images is not trivial and requires a few preprocessing steps: polygons need to be rescaled to the correct size according to the distance from the observatory to the sun (i.e. the distance of SDO to the sun); polygons need to be realigned relative to the event-time based on the time the image was taken; and polygons need to be repositioned according to where the sun is centred in the image. Lastly, not all HEK events contain the same information. For example, some have exact bounding boxes whereas others only have rectangular ones. This largely depends on the event extraction method.

When working with the SDO ML v2 dataset, the same preprocessing steps as in the creation of the dataset have to be applied to rescale and reposition the bounding boxes to match the image (i.e., rotation and resizing onto a common grid and rescaling to match the target image size).

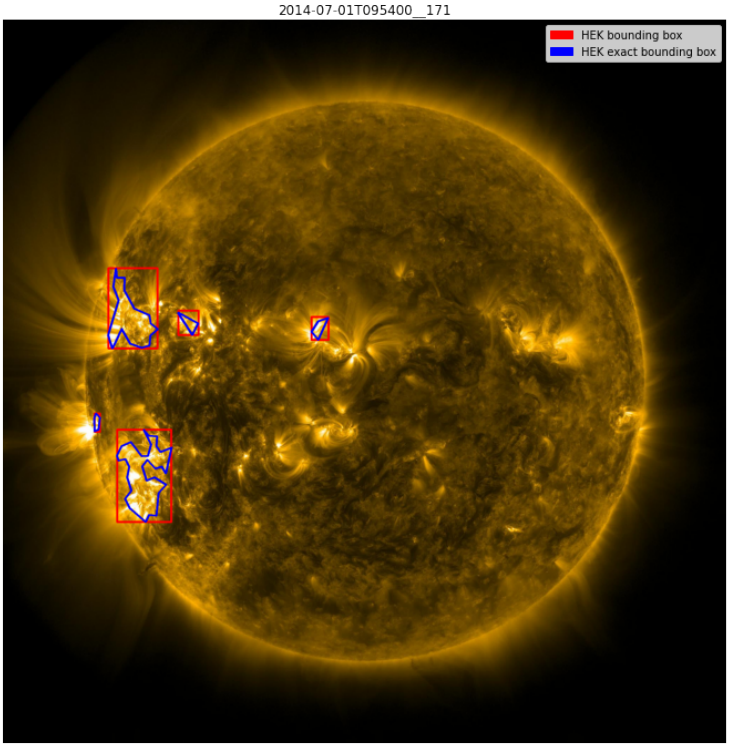


Figure 4.3: Example of active region events recorded in HEK by SPoCA on 2014-07-01 at 09:54AM

4.3 Data Preparation

In our research, we primarily focused on AIA images in the 171 Å band from the SDO ML v2 dataset. This band was chosen because a broad range of solar activity is visible. In addition, the band is used by different event tracking algorithms in the SDO event tracking pipeline. Once proven to be working, the extension of the model to other wavelength bands should be a straight forward task. The 171 Å band captures the upper transition region and quiet corona and is emitted by iron-9 (Fe IX) at approximately 600'000 K. The channel is well suited to study coronal loops, which are arcs extending off the sun where plasma moves along magnetic field lines. Other phenomena such as filaments, flares, and active regions are also clearly visible. Images from the 171 Å channel are typically colorized in gold.[9]

4.3.1 Data Splitting

Throughout the solar cycle, data is highly correlated temporally because large structures are often visible for several days and may even be present after a full rotation of the sun (∼28 days). A random split of the dataset might therefore produce overly optimistic estimates as information from the training set is leaked to the validation and test set. For this reason, an adequate split into train, validation and test set must be found that retains the underlying properties of the dataset and does not lead to validation/test set contamination.

A simple temporal splitting strategy consists of splitting the data into train and test sets on a yearly basis, such that the different solar activity levels are equally represented. This can be achieved by splitting the available years of data based on the integrated GOES flux (indicator for solar activity) and then splitting the data based on smaller temporal blocks for train and validation set within the larger temporal split. Optionally, a gap period, e.g. of 15 days, can be implemented between train and test set which should prevent that the same structures from appearing in both sets . For train-validation split,

[9]How SDO sees the sun: https://www.nasa.gov/mission_pages/sdo/how-sdo-sees-the-sun

a random split based on a smaller temporal unit can be implemented, e.g., a monthly split, for which a larger portion of the months is used for training and a smaller portion for validation.

We used roughly 70% of the data for the training set and 30% of the data for the test set. For the training and validation set a random split into chunks of 14 days (half a solar rotation) is used. Such that 80% of the remaining data was in the training set and 20% in the validation set (refer to Figure 4.4). Notably, data used for training will still have leaked into the validation/test sets at the borders of the temporal chunks. This could be prevented by introducing a gap period during which the data is discarded or could be minimized by choosing a large chunk size resulting in fewer neighbouring chunks. Because our use case is in the unsupervised regime, we disregarded this fact and did not discard any data.

We ended up with a train set size of approximately 500'000 observations and a validation set size of approximately 50'000 observations whereas the test set contains 320'000 observations.

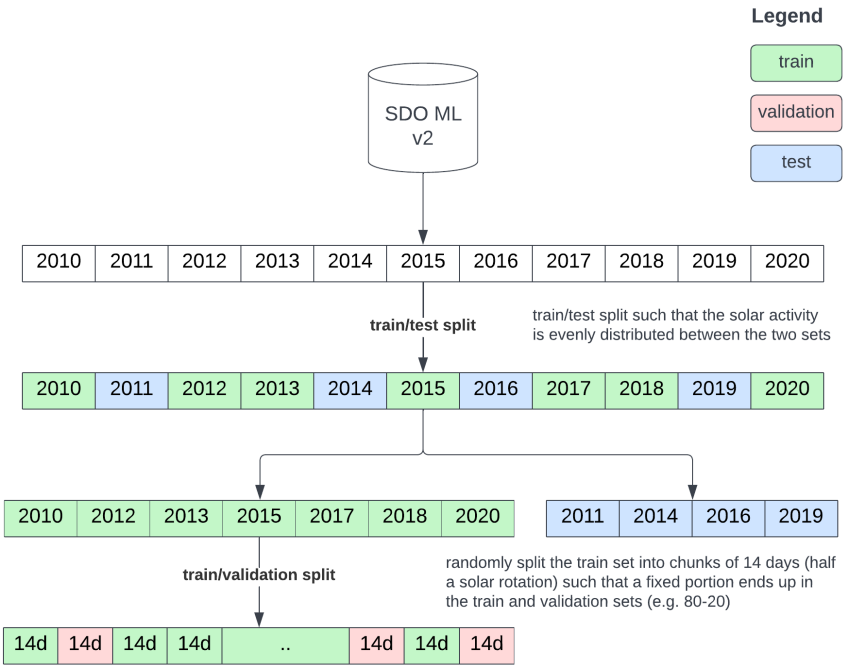


Figure 4.4: Data splitting strategy based on temporal splitting

For other applications, such as flare prediction, it might make sense to impose further constraints to properly sample rare events (e.g., ensuring that the number of C-, M-, and X-class flares is proportional throughout the different sets).

4.3.2 Preprocessing

We applied multiple preprocessing steps to prepare the data for use in a machine learning model. First, we resized the images of the SDO ML v2 dataset to 256 x 256 pixels. Second, based on the channel, we clamped the values between a minimum and maximum value and applied linear, square root or log10 scaling. Finally, we normalised the data between mean and standard deviation for the respective channel and rescaled it to -1 and 1. This is in accordance with the preprocessing steps that were applied in

solarnet.[10] The clamping and scaling factors are the same as in the SDO Benchmark Dataset.[11] Notably, in Helioviewer[12] a similar set of preprocessing is conducted, which was slightly modified by the authors of the SDO Benchmark Dataset (e.g., minimum values and scaling factors are chosen differently for certain channels).
We did not conduct further investigations regarding the different scaling strategies. It is left to subsequent studies to investigate whether there are scaling strategies for solar data which are better suited to machine learning.

For interacting with the SDO ML dataset, we have implemented a PyTorch DataLoader[13] that makes it possible to filter the data based on channel, date and GOES X-ray Flux; removes corrupt samples and provides temporal downsampling. Additionally, it implements the splitting strategy that was introduced in the previous section. Using this DataLoader it is easy to retrieve a customisable subset of the data on the fly.

Because the SDO ML dataset uses the Zarr format, it is split into chunks per channel. Each chunk contains 120 images which all need to be loaded at once (similar to a block on the hard disk). When sampling between chunks, the data loading process slows down which makes training inefficient. To alleviate this, we have implemented a chunk-sampling strategy that loads sequences of elements sequentially but randomizes sequences in a chunk and the chunks in the dataset. When training machine learning models, it is usually better if batches are uncorrelated. This is partially implemented by scrambling sequences within chunks and the chunks within the dataset; however, because the chunk only contains 120 images, covering roughly 12 hours, there is still a strong temporal correlation between the elements of a sequence. This might result in nonoptimal training which could take longer because weights are adjusted nonoptimally and is a trade-off between training speed and well-mixed batches.

[10] https://gitlab.com/jdonzallaz/solarnet/-/blob/36c8e8accf27ae40f649ba71bd58fdd3ce1a33a2/solarnet/data/transforms.py
[11] https://github.com/i4Ds/SDOBenchmark/blob/a4882916cf1f8c1a93fcab7617b6443d74bbfe2e/dataset/data/load.py#L363
[12] https://github.com/Helioviewer-Project/jp2gen/blob/9bd8f50e9044f95bf5b725cd3f77cf534e8dbdb2/idl/sdo/aia/hvs_default_aia.pro
[13] https://github.com/i4Ds/sdo-cli/blob/main/src/sdo/sood/data/sdo_ml_v2_dataset.py

Chapter 5

Applying the ceVAE Model

Based on the work of Zimmerer et al. [1], which is available as part of the Medical Out-of-Distribution Analysis Challenge 2020 repository,[1] we have implemented the algorithms for the context-encoding variational autoencoder. The following sections introduce the model configurations and provide details about the experimental setup.

5.1 Model Configuration

Architecture We have implemented a similar model architecture as proposed by Zimmerer et al. [1]: The inputs for the model are batches of AIA 171 Å full-disk grayscale images from the SDO ML v2 dataset that were preprocessed and split into train, validation and test set according to the specifications in Chapter 4. Due to the size of the dataset, we purely relied on network-based file storage (NAS) and tuned the data loading process in a way that would offer reasonable GPU-utilization. The encoder and decoder of the ceVAE model were implemented as fully convolutional neural networks, which are symmetrically composed of five 2D-Conv-Layers and 2D-Transposed-Conv-Layers. The feature maps of the convolutional layers were chosen to be of size 16, 64, 256 and 1024 with kernel size 4 and stride 2. Each layer is followed by a Leaky-ReLU activation. The encoders for the VAE as well as the CE branch have shared weights. The last layer has two heads, one predicting the mean and the other predicting the log-standard deviation that are then used to sample from a standard normal distribution to generate the reconstruction. The CE-branch is not sampling from the learnt distribution but rather directly forwarding the encoded representation of the mean to the decoder. Consistent with the original model, we also chose the L1 loss as the reconstruction loss L_{rec} because the authors have reported visually slightly better results using the L1 loss instead of another loss, such as loss MSE.
The model was implemented with PyTorch Lightning[2] which is a deep learning framework on top of PyTorch. PyTorch Lightning defines different hooks for the training, validation and testing phases and glues together the different parts of the code thereby removing boilerplate code. Further, it provides useful utilities, such as early stopping, different loggers, checkpointing and automatic device management which is useful for GPU-based and distributed training. To track the experiments, we made use of Weights & Biases[3], a cloud-based tool for experiment tracking, model registry and hyperparameter search. We ran the training on the FHNW infrastructure (4x NVIDIA GeForce RTX 2080 Ti, 126GB RAM, AMD EPYC 7551P 32-Core Processor) and used approximately 180 hours compute time to train the final models using a single GPU.

Hyperparameter Tuning The most relevant hyperparameters are the dimension of the latent space (z_dim), the input size and target size of the reconstructed output, the feature map sizes of the

[1] https://github.com/MIC-DKFZ/mood
[2] https://pytorch-lightning.readthedocs.io/en/stable/
[3] https://wandb.ai/site

convolutional encoder and decoder, the CE-factor which is the amount to which the context-encoder contributes to the model (between 0 which means standard VAE and 1 which means purely CE), the β-factor, which is a weighting factor for the KL loss to influence the loss, the number of training epochs, the batch size and the learning rate.

We have trained several different models, using different configurations, which are summarised in the following table:

Parameter/Model name	default-128	default-256	limb-masking	standard VAE
Latent Dimension	128	256	256	128
Input/Target Size	256			
Feature Map Sizes	16, 64, 256 and 1024			
CE-factor	0.5	0.5	0.5	0
β-factor	0.01			
# Epochs	5	5	5	5
Batch Size	8			
Learning Rate	0.00001			
# Trainable Params	107 mil.	208 mil.	208 mil.	107 mil.

Table 5.1: Hyperparameters of the different models

In all cases except the Standard VAE model, both VAE and CE branches of the model are weighted equally (CE-factor of 0.5). The small batch size of 8 was chosen as a counter measure to the sequence in chunk sampling (performance tuning for the Zarr-based SDO ML v2 dataset, see Chapter 4) which results in poor variation within a batch and leads to nonoptimal training if the batch size is larger. Furthermore, we have used a small learning rate of 0.00001 which led to a bit more stable training. A latent space size of 128 turned out to yield reasonable reconstructions. Further information can be found in Chapter 6. The source code of the model is available as part of the *sdo-cli* Github repository.[4]

To assess whether the model overfits the training set, we made use of the validation set and used early stopping to either manually or automatically (with PyTorch Lightning callbacks) stop the training. The training was terminated if the validation loss would no longer improve or even increase again. The best model checkpoints in terms of validation loss were stored and used for the subsequent analysis of the models presented in the next chapter (6).

5.2 Helpful Utilities

We have implemented several procedures to make training models on SDO data more straight-forward. Most of these are not restricted to heliophysics and can be applied to other areas as well:

Exposing functionality as a CLI To emphasise reproducible results, we have implemented *sdo-cli*, a library and command-line interface (CLI) that wraps common tasks such as training/validation of models when working with SDO data. The source code is available as a Github repository.[5] We have also published the library as a PyPI package[6] to allow easy installation for potential users. An example command output is illustrated in Figure 5.1. Further information can be retrieved in Appendix A.

[4] https://github.com/i4Ds/sdo-cli/blob/main/src/sdo/sood/algorithms/ce_vae.py
[5] https://github.com/i4Ds/sdo-cli/
[6] https://pypi.org/project/sdo-cli/

```
(.venv) ➜  sdo-cli git:(main) ✗ sdo-cli sood ce_vae --help
Usage: sdo-cli sood ce_vae [OPTIONS] COMMAND [ARGS]...

Options:
  --help  Show this message and exit.

Commands:
  generate  Generate a set of images with the CE-VAE model (requires a
            pretrained model)
  predict   Predicts anomaly scores using a CE-VAE model (requires a
            pretrained model)
  train     Trains a CE-VAE model
```

Figure 5.1: sdo-cli: a small helper toolkit for machine learning with SDO data

File-based config As the amount of configuration parameters quickly increased, we needed to implement a way to efficiently and reproducibly manage configuration. For this reason, we used a YAML-based configuration format which allows defining a default set of parameters (defaults.yaml) that can be overridden in an individual run configuration.[7] The run configuration is set when invoking the sdo-cli command for training the model. See Figure 5.2 for an example of the default configuration.

config > ce-vae > ! defaults.yaml

```yaml
model:
  target_size:
    value: 256
    desc: "Target size of the reconstructed output"
  z_dim:
    value: 128
    desc: "Dimension of the latent space"
  fmap_sizes:
    value: [16, 64, 256, 1024]
    desc: "Feature map sizes for the CNN"
  ce_factor:
    value: 0.5
    desc: "Amount to which the context-encoder contributes to the model (between 0 only VAE and 1 only CE)"
  load_path:
    value: null
    desc: "Path to a pretrained model"
data:
  batch_size:
    value: 16
    desc: "How many samples per batch to load"
  channel:
    value: "171A"
    desc: "Channel name that should be used. If None all available channels will be used."
  data_dir:
    value: ./data
    desc: "Path to the root directory of the dataset"
  dataset:
    value: SDOMLDatasetV2
    desc: "Which dataset to use (CuratedImageParameterDataset, SDOMLDatasetV1 or SDOMLDatasetV2)"
  num_data_loader_workers:
    value: 0
    desc: "How many subprocesses to use for data loading. 0 means that the data will be loaded in the main process."
```

Figure 5.2: Extract of the default configuration

Powerful data loader To accommodate different scenarios, handle a large dataset (several TeraBytes) and compose a new data-subset on the fly, we have implemented a powerful PyTorch DataLoader[8] for the SDO ML v2 dataset.[9] It allows to load, filter, preprocess and split data from either a local file system, a network file storage or a Google Cloud Storage bucket. The SDO ML v2 data loader allows to filter data by channel, date range or GOES irradiance, downsample data temporally and includes several measures to correct data quality (e.g., dropping invalid samples and attributes). Further, it allows to compose a train/validation/test split by exposing a PyTorch Lightning DataModule.[10] One particular trick to make the data loading more efficient was the implementation of a sequence-in-chunk sampler. The sampler loads consecutive images belonging to the same Zarr chunk and thus reduces the number of files that needs to be loaded for a single batch.

[7] e.g., https://github.com/i4Ds/sdo-cli/blob/main/config/ce-vae/run-fhnw-full-2-256.yaml
[8] https://pytorch.org/tutorials/beginner/basics/data_tutorial.html
[9] https://github.com/i4Ds/sdo-cli/blob/main/src/sdo/sood/data/sdo_ml_v2_dataset.py
[10] https://pytorch-lightning.readthedocs.io/en/stable/extensions/datamodules.html

GOES timeseries as a Distributed Data Frame To gain efficient access to the GOES X-ray flux timeseries, we have implemented a caching mechanism based on a distributed data frame (DDF) using the Python library *dask*.[11] Using dask, we have split the timeseries into weekly chunks that can be efficiently indexed, filtered and converted to a pandas DataFrame.[12]

Experiment Tracking with Weights & Biases Tracking experiments with the cloud-based tool Weights & Biases turned out to be extremely valuable to compare different training runs (e.g., monitoring training metrics, gradients and system performance). Also see Figure 5.3 for an example. The reports feature turned out to be useful to compare and share a specific set of runs. Additionally, we have also used the tool as the model registry providing model checkpoints for the different training runs and thereby enabling the use of pretrained models.

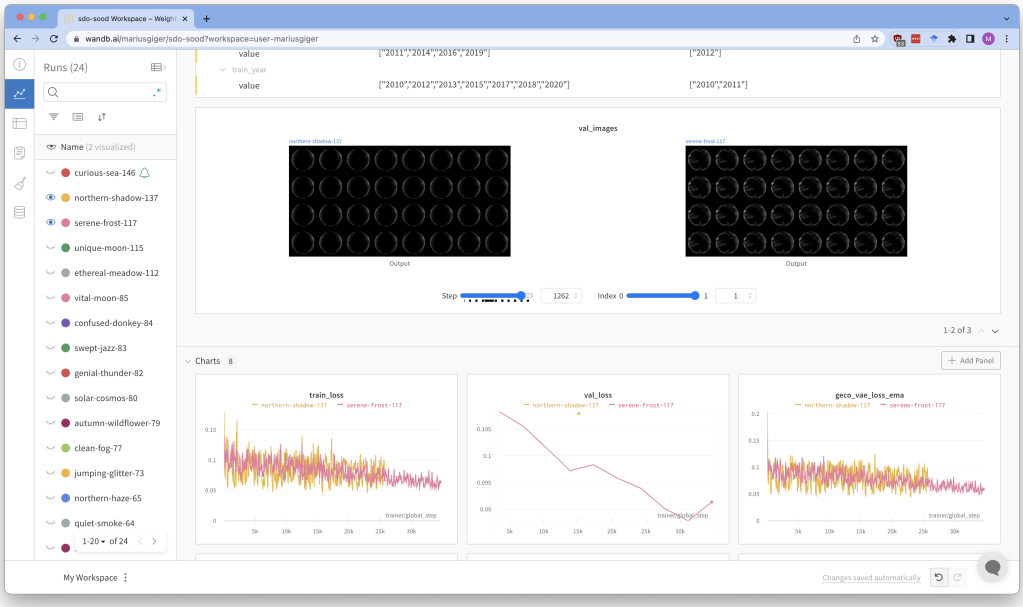
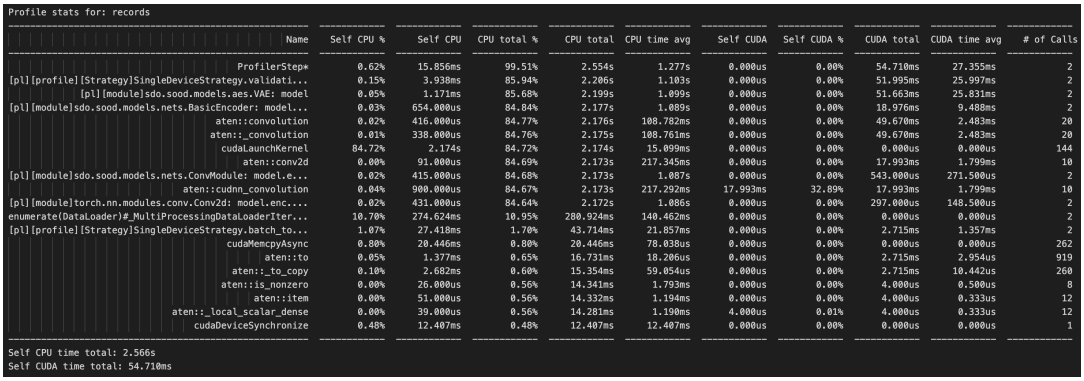


Figure 5.3: Experiment Tracking with Weights & Biases

Debugging To debug the model, we made use of the PyTorch profiler[13][14] and thereby gaining valuable insights to optimize the training (e.g., see Figure 5.4). For example, identifying the runtime of an individual method or whether the GPU is used for a certain calculation. Furthermore, we used the following tools to monitor training runs locally on the FHNW compute infrastructure: glances,[15] nvtop[16] and nvidia-smi.[17]

[11] https://docs.dask.org/en/stable/dataframe.html
[12] https://pandas.pydata.org/docs/reference/api/pandas.DataFrame.html
[13] https://pytorch-lightning.readthedocs.io/en/stable/advanced/profiler.html
[14] https://pytorch.org/tutorials/recipes/recipes/profiler_recipe.html
[15] https://nicolargo.github.io/glances/
[16] https://github.com/Syllo/nvtop
[17] https://developer.nvidia.com/nvidia-system-management-interface



Profile stats for: records

Name	Self CPU %	Self CPU	CPU total %	CPU total	CPU time avg	Self CUDA	Self CUDA %	CUDA total	CUDA time avg	# of Calls
ProfilerStep*	0.62%	15.856ms	99.51%	2.554s	1.277s	0.000us	0.00%	54.710ms	27.355ms	2
[pl][profile][Strategy]SingleDeviceStrategy.validati...	0.15%	3.938ms	85.94%	2.206s	1.103s	0.000us	0.00%	51.995ms	25.997ms	2
[pl][module]sdo.sood.models.aes.VAE: model	0.05%	1.171ms	85.68%	2.199s	1.099s	0.000us	0.00%	51.663ms	25.831ms	2
[pl][module]sdo.sood.models.nets.BasicEncoder: model...	0.03%	654.000us	84.84%	2.177s	1.089s	0.000us	0.00%	18.976ms	9.488ms	2
aten::convolution	0.02%	416.000us	84.77%	2.176s	108.782ms	0.000us	0.00%	49.670ms	2.483ms	20
aten::_convolution	0.01%	338.000us	84.76%	2.175s	108.761ms	0.000us	0.00%	49.670ms	2.483ms	20
cudaLaunchKernel	84.72%	2.174s	84.72%	2.174s	15.099ms	0.000us	0.00%	0.000us	0.000us	144
aten::conv2d	0.00%	91.000us	84.69%	2.173s	217.345ms	0.000us	0.00%	17.993ms	1.799ms	10
[pl][module]sdo.sood.models.nets.ConvModule: model.e...	0.02%	415.000us	84.68%	2.173s	1.087s	0.000us	0.00%	543.000us	271.500us	2
aten::cudnn_convolution	0.04%	900.000us	84.67%	2.173s	217.292ms	17.993ms	32.89%	17.993ms	1.799ms	10
[pl][module]torch.nn.modules.conv.Conv2d: model.enc....	0.02%	431.000us	84.64%	2.172s	1.086s	0.000us	0.00%	297.000us	148.500us	2
enumerate(DataLoader)#_MultiProcessingDataLoaderIter...	10.70%	274.624ms	10.95%	280.924ms	140.462ms	0.000us	0.00%	0.000us	0.000us	2
[pl][profile][Strategy]SingleDeviceStrategy.batch_to...	1.07%	27.418ms	1.70%	43.714ms	21.857ms	0.000us	0.00%	2.715ms	1.357ms	2
cudaMemcpyAsync	0.80%	20.446ms	0.80%	20.446ms	78.038us	0.000us	0.00%	0.000us	0.000us	262
aten::to	0.05%	1.377ms	0.65%	16.731ms	18.206us	0.000us	0.00%	2.715ms	2.954us	919
aten::_to_copy	0.10%	2.682ms	0.60%	15.354ms	59.054us	0.000us	0.00%	2.715ms	10.442us	260
aten::is_nonzero	0.00%	26.000us	0.56%	14.341ms	1.793ms	0.000us	0.00%	4.000us	0.500us	8
aten::item	0.00%	51.000us	0.56%	14.332ms	1.194ms	0.000us	0.00%	4.000us	0.333us	12
aten::_local_scalar_dense	0.00%	39.000us	0.56%	14.281ms	1.190ms	4.000us	0.01%	4.000us	0.333us	12
cudaDeviceSynchronize	0.48%	12.407ms	0.48%	12.407ms	12.407ms	0.000us	0.00%	0.000us	0.000us	1

Self CPU time total: 2.566s
Self CUDA time total: 54.710ms

Figure 5.4: Profiling output of the PyTorch profiler

The next chapter introduces and analyses the results that were obtained with the different models.

Chapter 6

Out-of-Distribution Analysis

For each of the four models introduced in the last chapter as well as for the baseline model, we ran a set of analyses on the test set (years 2011, 2014, 2016 and 2019). First, we investigated the reconstruction quality as a measure of how well the models can reproduce a given image. Second, we studied the anomaly predictions of the models and compared the image-wise scores to the GOES X-ray flux and the baseline scores. We then analysed the pixel-wise scores for the highest and lowest ranking images with respect to the image-level anomaly score. Finally, we looked at the latent space and tried to reason about the model-internal representations by generating new images and interpolating within the latent space. This chapter summarizes the results of the different analyses.

6.1 Inputs and Reconstructions

To assess how well the models can represent a given input, it is useful to consider the resulting reconstructions. The reconstructions are obtained by first encoding an image with the encoder and then decoding the encoded representation with the decoder of the model. Figure 6.1 shows a set of inputs and reconstructions for the four models. All four models were able to capture most of the activity in the input image. However, the reconstructions are rather blurry, which lies in the nature of variational autoencoders. To better understand the reconstruction quality, we calculated the structural similarity index measure (SSIM) [71], the mean squared error (MSE) and the mean absolute error (MAE) for the test set shown in Table 6.1. The SSIM is commonly used to quantify the similarity between two images and incorporates structural features instead of just the pixel-wise errors. A value of 1 indicates that two images are very similar while a value of 0 means that the two images are very different when using the implementation in PyTorch metrics.[1]

[1] https://torchmetrics.readthedocs.io/en/stable/image/structural_similarity.html

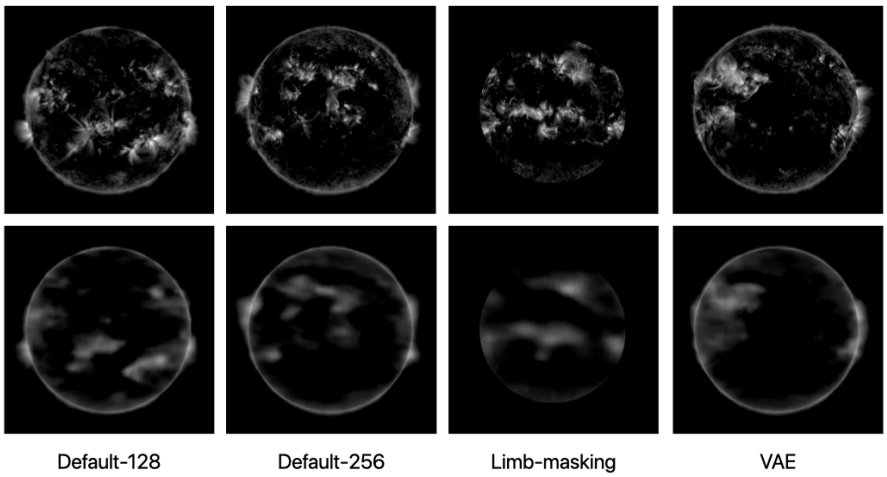


Figure 6.1: Inputs and reconstructions for the different models.

The reconstruction metrics provide a relative comparison between the different models and help to understand which model is able to better reproduce the structure of solar images and will thereby better capture solar activity. Notably, the comparison is not valid for the limb-masking model because the input images are masked, and hence the reconstruction is inherently easier for this model. Both visually and when considering the structural similarity as well as the MSE and MAE, the default-256 model seems to yield the best reconstructions and is therefore analysed in more detail alongside the limb-masking model.

Metric/Model name	default-128	default-256	limb-masking	standard VAE
MSE	0.010 $\pm$ 0.004	**0.009** $\pm$ 0.004	(0.005 ± 0.002)	0.012 ± 0.006
MAE	0.066 $\pm$ 0.017	**0.064** $\pm$ 0.016	(0.037 ± 0.009)	0.073 ± 0.02
SSIM	0.558 $\pm$ 0.043	**0.568** $\pm$ 0.041	(0.689 ± 0.021)	0.542 ± 0.047

Table 6.1: Reconstruction errors for the different models over the test set.

6.2 Out-of-Distribution Detection

Because we did not apply any filter to the training data (e.g., only quiet sun data), the models operate virtually assumption free and therefore consider unusual activity anomalous (or out of distribution). Depending on the application it might make sense to further restrict the training data and exclude extreme events or even train on quiet sun data only, which could be a direction for future research. To detect anomalous activity, we first retrieved the sample-level anomaly scores for the full test dataset, which are calculated by evaluating the evidence lower bound for each image (see Section 3.2.1). Subsequently, we computed the pixel-level scores for images with very high and very low scores, for which we made use of the *combi* score mode which is the combination of the density- and reconstruction-based pixel scores. Additionally, we compared the different score modes for the default-256 model.

6.2.1 Image-level Anomaly Scores

Before any further analysis, the image-level scores were normalised (rescaled between 0 and 1) to get a more intuitive representation. Scores closer to 1 are more out-of-distribution or anomalous, while scores closer to 0 are considered normal. Figure 6.2 shows the distribution of normalised anomaly scores for the different models over the test set. For the default-256, the default-128, and the VAE model, the

histogram shows three peaks and for the limb-masking model two. The shape of the distributions can be attributed to the nonstationary solar cycle and the data selection process for the test set; the left-most peak, consisting of very low scores, can be assigned to the solar minimum in 2019; scores in the centre represent the time in between solar minimum and maximum (2011 and 2016); and the right-most peak with a long tail represents the very active periods during the solar maximum. This is confirmed when studying the scores over time shown in Figure 6.3.

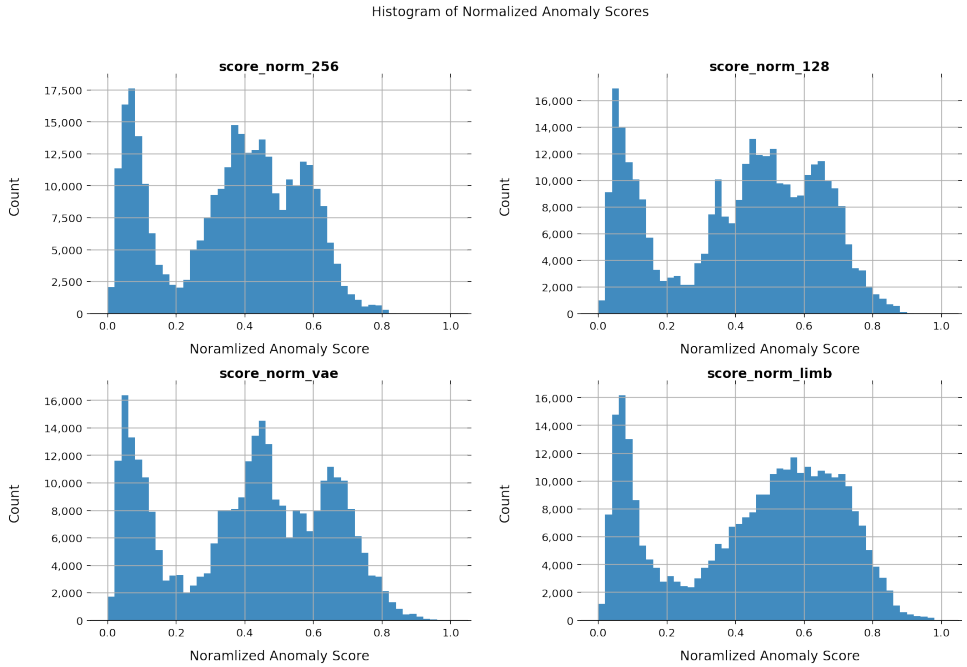


Figure 6.2: Histograms of normalised anomaly scores for the different models obtained for the test set (top left default-256 model, top right default-128 model, bottom left VAE model and bottom right limb-masking model)

To understand whether the obtained anomaly scores align with solar activity, we compared the image-wise scores with the GOES X-ray flux (xrsb). We expected that GOES X-ray flux and anomaly scores would approximately match because the GOES X-ray flux is used as an indicator of solar activity, and we expected active periods to contain more anomalous observations. Figure 6.3 shows this comparison. It is evident that during the solar maximum (2014), the anomaly scores were generally high. During the solar minimum, in 2019, the anomaly scores were low, and in 2011 and 2016, they were somewhere in between. Generally, the two time series match visually quite well. However, obtaining a time series correlation between GOES flux and anomaly score is not trivial because metrics like the Pearson correlation coefficient cannot easily be used because of the highly different scales and the nonstationary solar cycle. Therefore, we settled for a visual comparison. Figure 6.4 shows this comparison for the solar maximum and Figure 6.5 for the solar minimum. Notably, the anomaly scores of the different models align quite well among themselves, especially during the solar minimum, with the limb-masking model resulting in somewhat higher peaks. Furthermore, during the solar minimum, a periodicity of the anomaly scores can be observed that matches the solar rotation and can likely be attributed to solar activity (e.g., several active regions that stayed during this time and was caught by the model every time the region came into view).

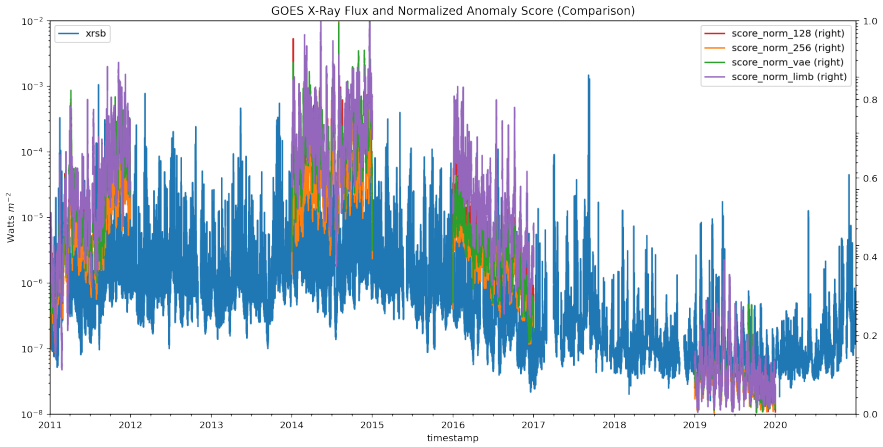


Figure 6.3: Alignment of Goes X-ray flux and normalised anomaly score for the different models.

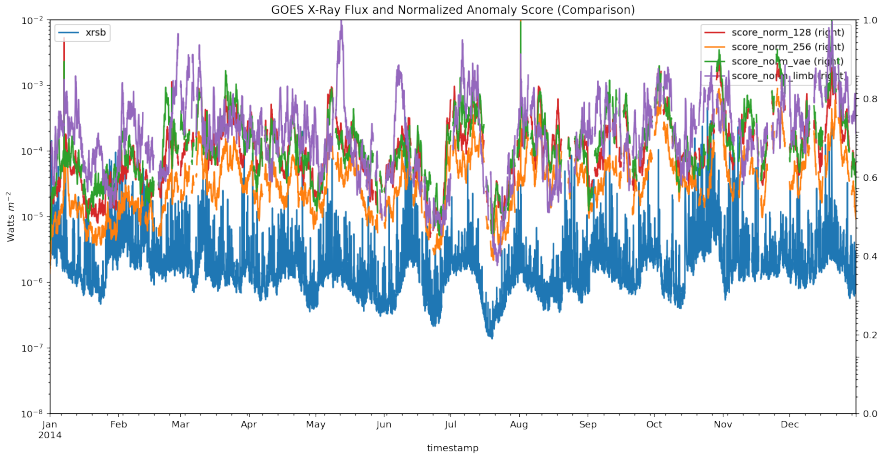


Figure 6.4: Alignment of Goes X-ray flux and normalised anomaly scores for the different models during the solar maximum (2014).

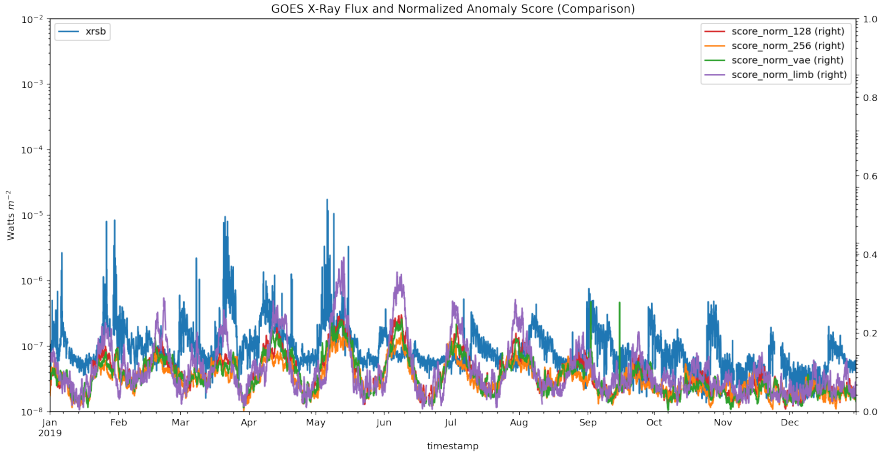


Figure 6.5: Alignment of Goes X-ray flux and normalised anomaly scores for the different models during the solar minimum (2019).

A comparison of the image-wise anomaly scores and the baseline scores for the solar maximum is shown in Figure 6.6 and for the solar minimum in Figure 6.7. There is some alignment between the machine learning-based ceVAE model and the naive baseline model which hints that at least some of the variations of the scores of the ceVAE model can be traced back to the mean pixel intensity. During the solar minimum the alignment is less evident but also the baseline scores expose a certain periodicity.

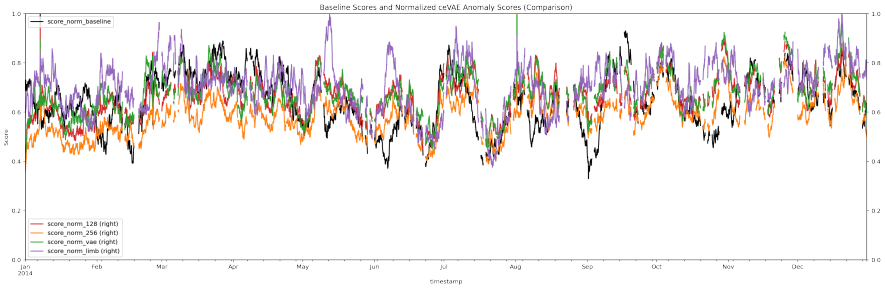


Figure 6.6: Alignment of the baseline scores and normalised anomaly scores for the different models during the solar maximum (2014).

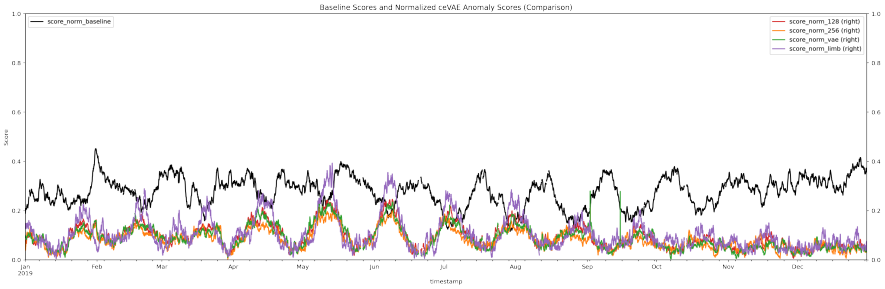


Figure 6.7: Alignment of the baseline scores and normalised anomaly scores for the different models during the solar minimum (2019).

6.2.2 Pixel-level Anomaly Scores

The highest-scoring images for the default-256 model, the related pixel-scores and the scores for the baseline model are shown in Figures 6.8, 6.9 and 6.10. The lowest-scoring images are shown in Figures 6.11 and 6.12. Similarly, images and scores for the limb-masking model are shown in Figures 6.13, 6.9, 6.15, 6.16, and 6.17, respectively. The baseline pixel-level predictions are left out for the low-scoring images because they do not contain any anomalous pixels. Results from the other models can be found in Appendix B.

The default-256 model assigns high image-level anomaly scores to images with much activity and low scores to images that have little activity and show the quiet sun. Both top-scoring images contain a flare, but do also contain artefacts from overexposure, which might have an impact on the high score. The other high-scoring images contain a great deal of activity and show strong magnetic activity that reaches far into space. The pixel-level scores hold interesting information about the regions that the model considers anomalous. We have normalised and inverted the pixel-level anomaly scores to make it visually easier to distinguish out-of-distribution regions. Therefore, the darker areas in the pixel-level anomaly maps are considered to be more anomalous. In comparison to the baseline scores, which are purely based on the pixel intensity, the default-256 model marks regions with unusual activity as out of distribution regardless of the intensity. These regions are not necessarily brighter in the input image but can also be darker (e.g., related to filaments or coronal holes). Nevertheless, regions with strong activity are consistently marked as out of distribution. Furthermore, the model predictions can be traced back to the location (even beyond the solar limb) and structure of the appearing phenomenon (e.g., filament in the top right image of Figure 6.9). This clearly indicates that the model has learned something about the structure of the sun and solar activity. To better understand what the model had learned, we analysed the model-internal representation shown in Section 6.3. Importantly, the pixel-level scores for the low-scoring images also seem to carry interesting information with coronal streamers and other less obvious activities being highlighted.

The limb-masking model was trained by masking input images, reducing the analysed disk size to 90% of its original size and thereby removing the solar limb. For this reason, all activity that is reaching out into space is masked, and the shearing effect at the edge of the solar disk is reduced. The assumption was to let the model focus on what is happening on the disk and remove some of the noise induced by limb shearing and activity reaching beyond the solar limb. This might have a negative impact on the model's ability to learn the physical properties of the sun, which are greatly altered (e.g., by cutting coronal loops that would otherwise connect). However, the model could be able to monitor certain phenomena more robustly on the disk.

Similar to the other models, the limb-masking model identifies images with a large amount of activity during the solar maximum as out of distribution, and the pixel-level scores highlight regions with unusual activity. Images with a low score contain the quiet sun in the solar minimum. It is less obvious why certain regions are chosen to be out of distribution because some of the resulting pixel-level scores can be attributed to active regions moving in or out of the view and therefore gain more emphasis by the model than phenomena in the centre of the disk. Surprisingly, the model had more difficulties learning a lower-dimensional representation of the sun when masking the limb (see the model reconstructions and the latent space analysis in the next section). Another masking approach with different boundaries possibly might produce better outputs.

A comparison of the different pixel-level score modes, introduced in Section 3.2.1, for highest and lowest ranking images of the default-256 model is illustrated in Figure 6.18 and Figure 6.19. The Rec scores, which are purely based on the reconstruction errors, are most easily interpreted as the pixel-wise difference between input and reconstruction. Both Rec-grad and ELBO-grad (combination of KL-grad and Rec-grad) scores contain artefacts outside the solar limb which do not seem to carry meaningful information and are likely caused by the gradients of the reconstruction error as the KL-grad scores show more expressiveness. The KL-grad scores exhibit interesting patterns that can be traced back to the model-internal deviations from normality. The combination of Rec and KL-grad scores therefore seems

to be a reasonable choice. However, depending on the application it might make sense to consider the Rec scores as a more conservative anomaly score.

Highest-Scoring Images (default-256 Model)

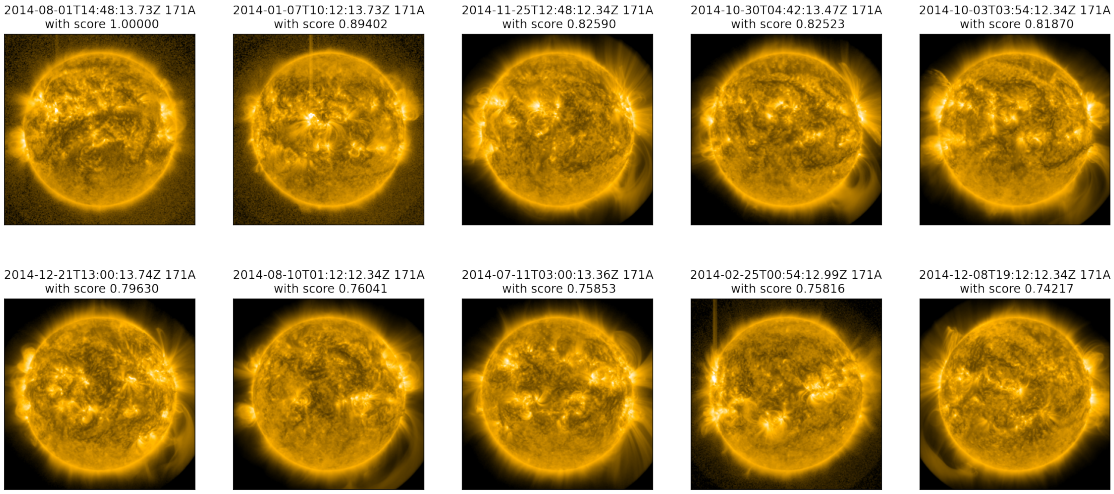


Figure 6.8: Top-scoring sample-level predictions for the default-256 model chosen to be at least one week apart from each other.

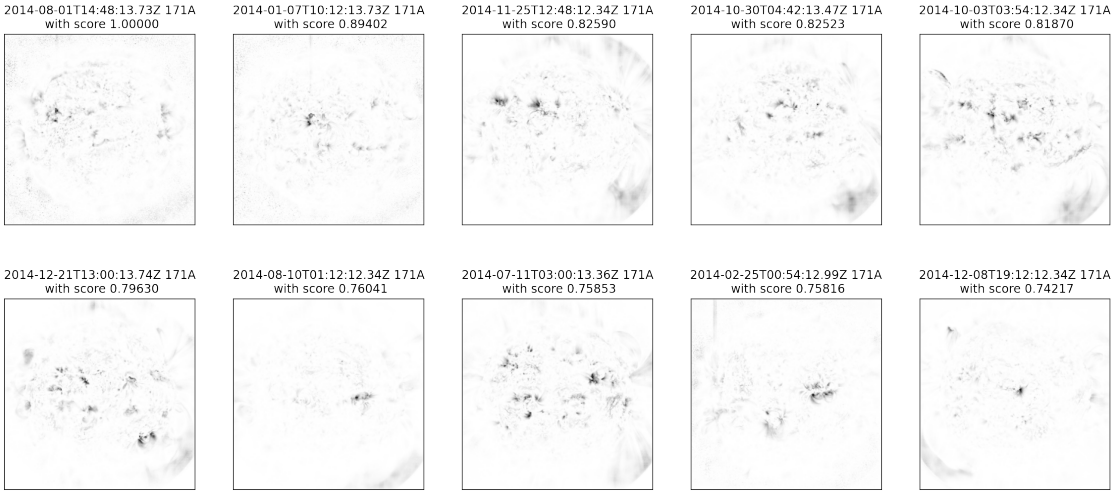


Figure 6.9: Pixel-level predictions for the top-scoring sample-level predictions of the default-256 model.

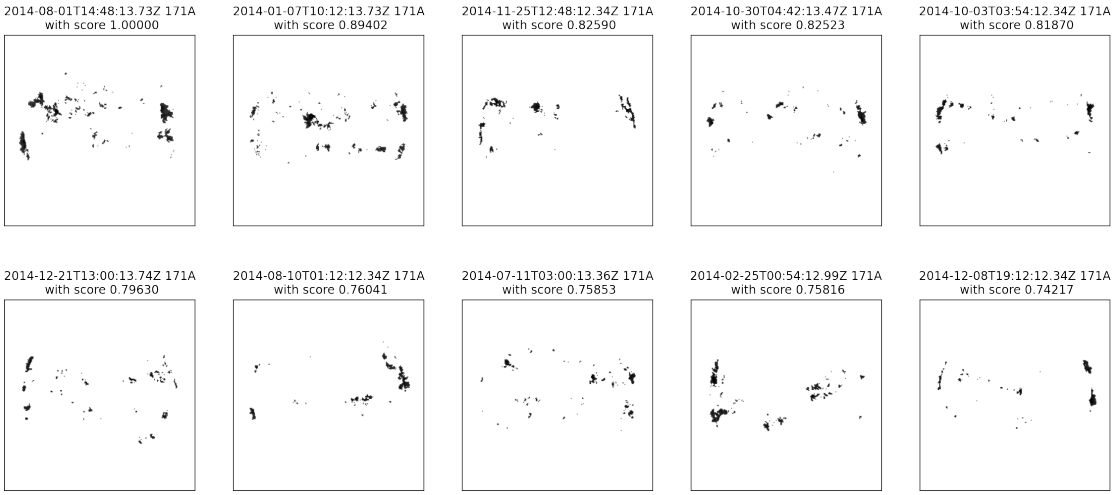


Figure 6.10: Baseline predictions for the top-scoring sample-level predictions of the default-256 model.

Lowest-Scoring Images (default-256 Model)

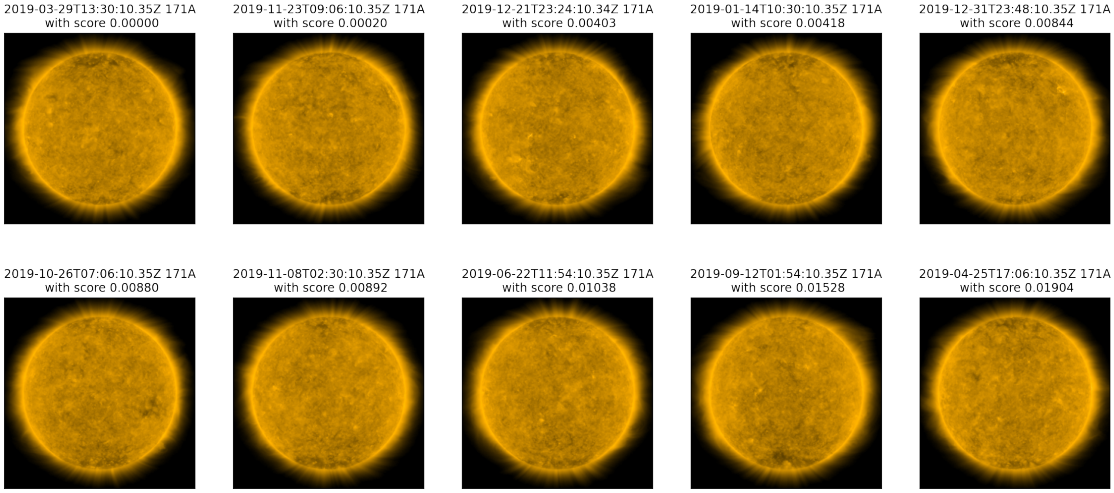


Figure 6.11: Lowest-scoring sample-level predictions for the default-256 model chosen to be at least one week apart from each other.

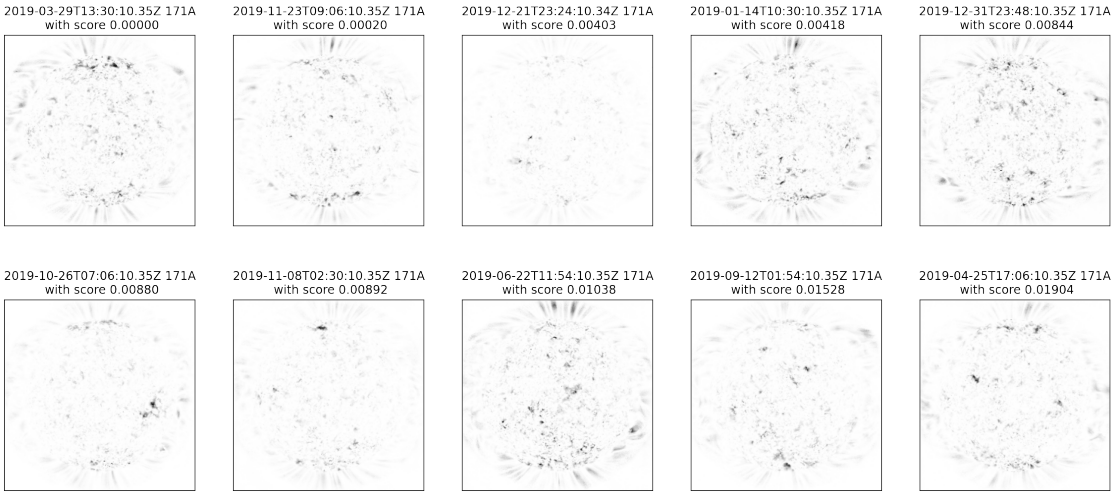


Figure 6.12: Pixel-level predictions for the lowest-scoring sample-level predictions of the default-256 model.

Highest-Scoring Images (limb-masking Model)

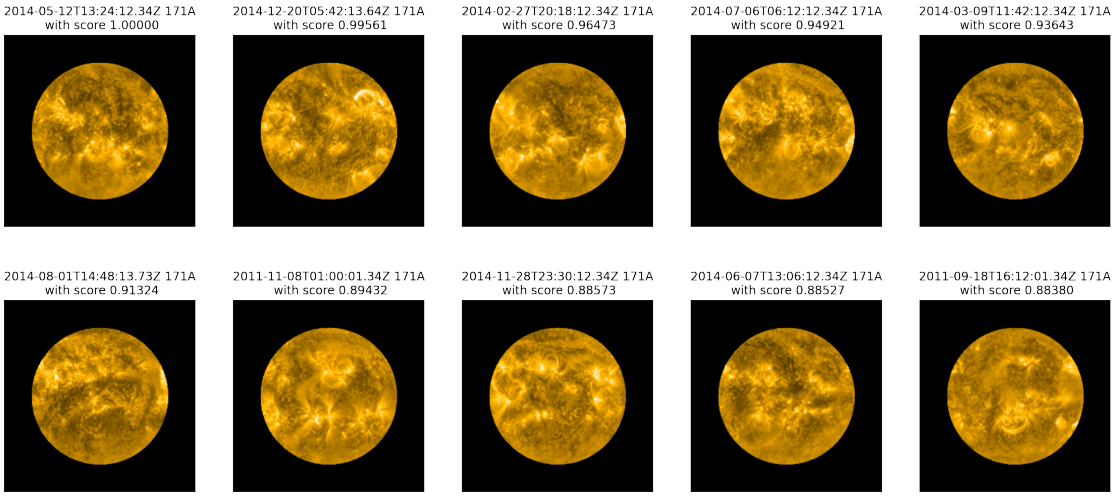


Figure 6.13: Top-scoring sample-level predictions for the limb-masking model chosen to be at least one week apart from each other.

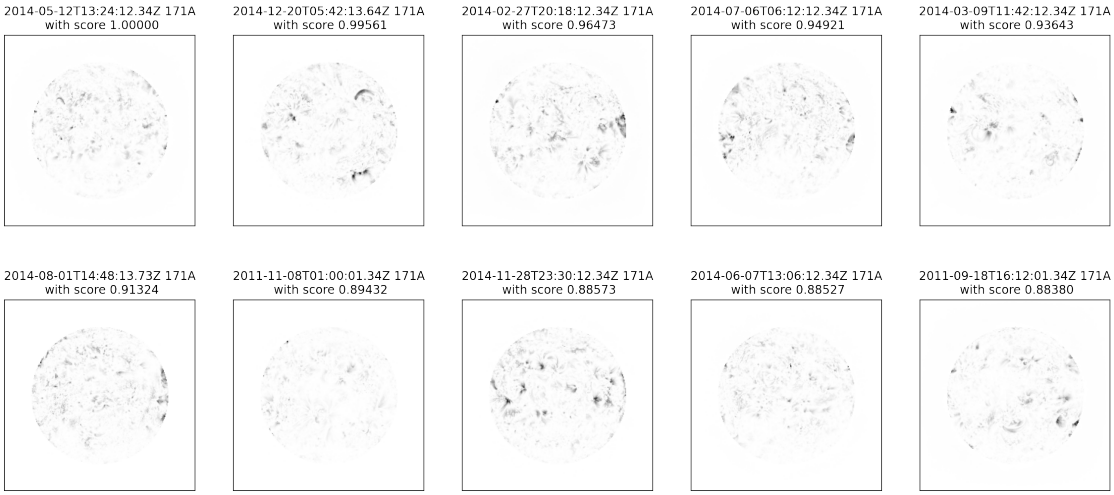


Figure 6.14: Pixel-level predictions for the top-scoring sample-level predictions of the limb-masking model.

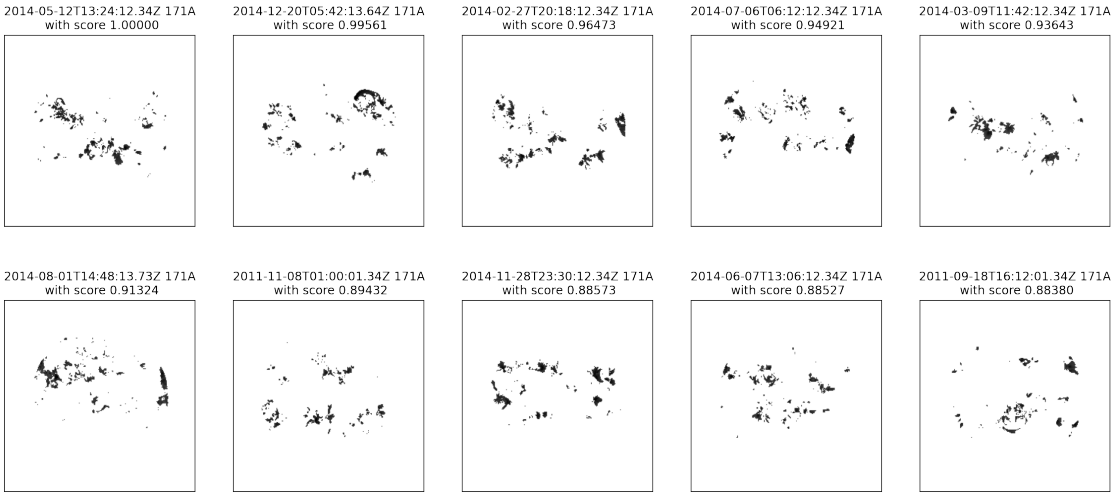


Figure 6.15: Baseline predictions for the top-scoring sample-level predictions of the limb-masking model.

Lowest-Scoring Images (limb-masking Model)

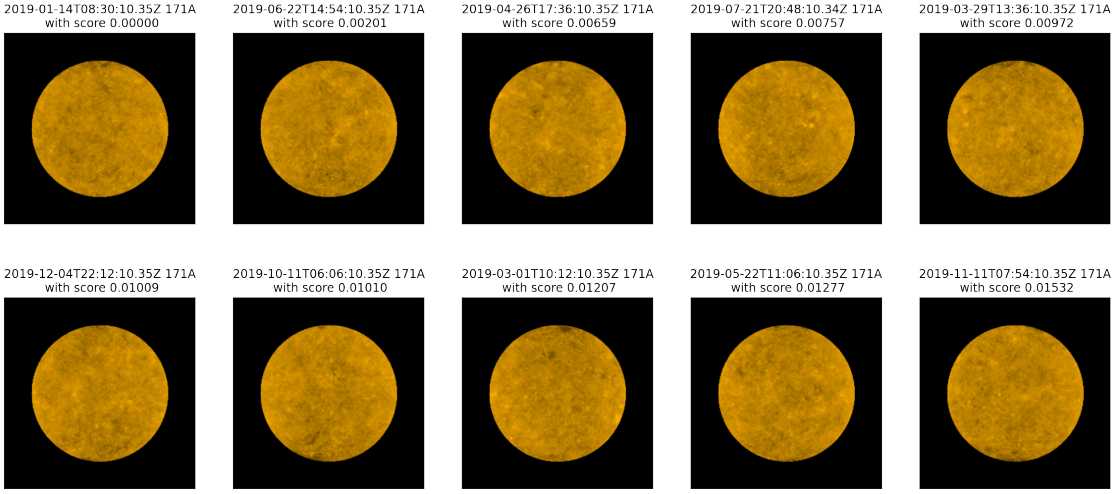


Figure 6.16: Lowest-scoring sample-level predictions for the limb-masking model chosen to be at least one week apart from each other.

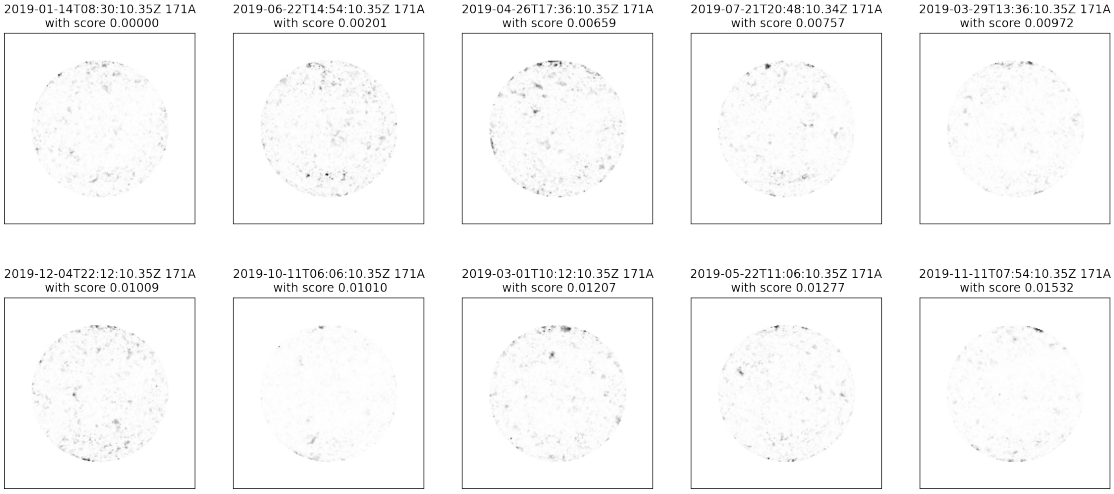


Figure 6.17: Pixel-level predictions for the lowest-scoring sample-level predictions of the limb-masking model.

Comparison of Different Score Modes (default-256 Model)

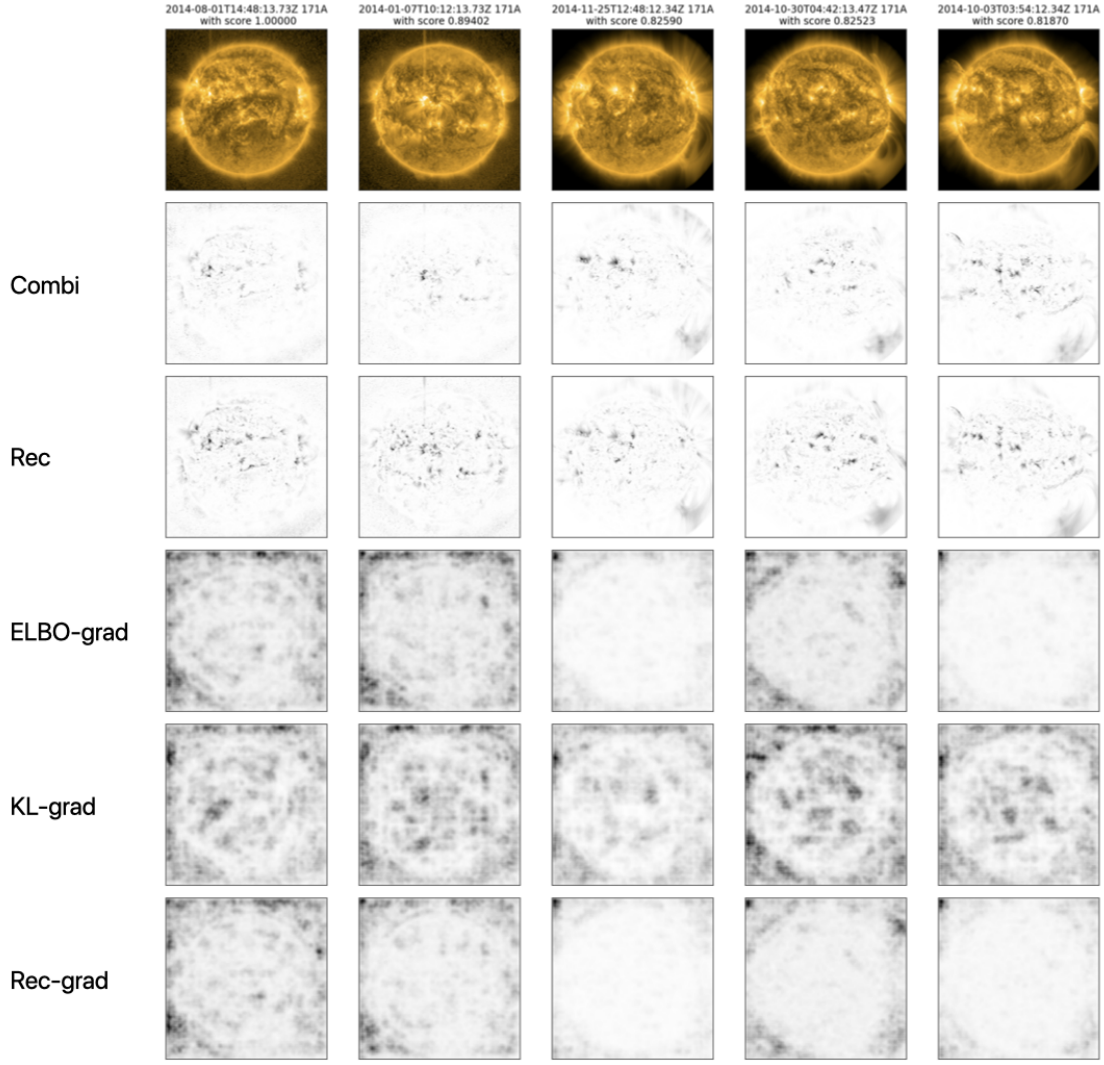


Figure 6.18: Pixel-level predictions for the different score modes of the top-scoring sample-level predictions of the default-256 model.

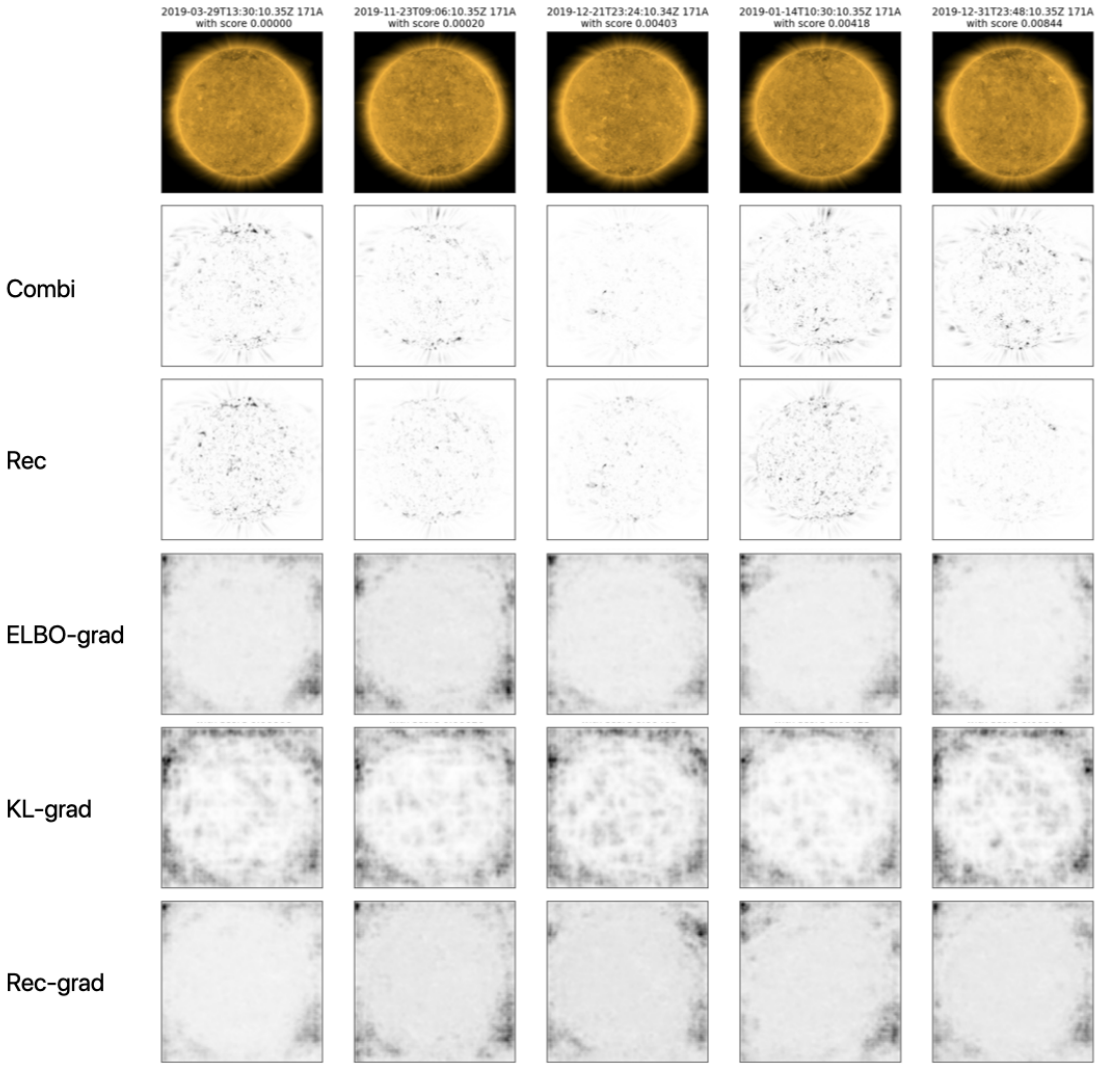


Figure 6.19: Pixel-level predictions for the different score modes of the top-scoring sample-level predictions of the default-256 model.

6.3 Understanding What the Model Has Learned

To understand what the ceVAE model has learned, new images can be generated, leveraging the generative capabilities of the variational autoencoder. A set of generated images using the default-256 model (left) and the limb-masking model (right) are illustrated in Figure 6.20. It is clearly visible that both models have learned to represent the spherical shape of the sun and also expose some activity patterns that can be attributed to coronal holes, coronal loops and active regions. Notably, the default-256 model seems to generate more realistic images with fine-grained details, while the limb-masking model shows many large-scale patterns.

For the VAE model, many of the generated images did not carry much information, suggesting that when using a standard VAE, only a few dimensions of the latent space z are actively used (this is consistent with theory [6, p. 694]). Using a context encoder alongside a VAE at training time seems to lead to more robust encodings and consequently has a positive impact on model performance.

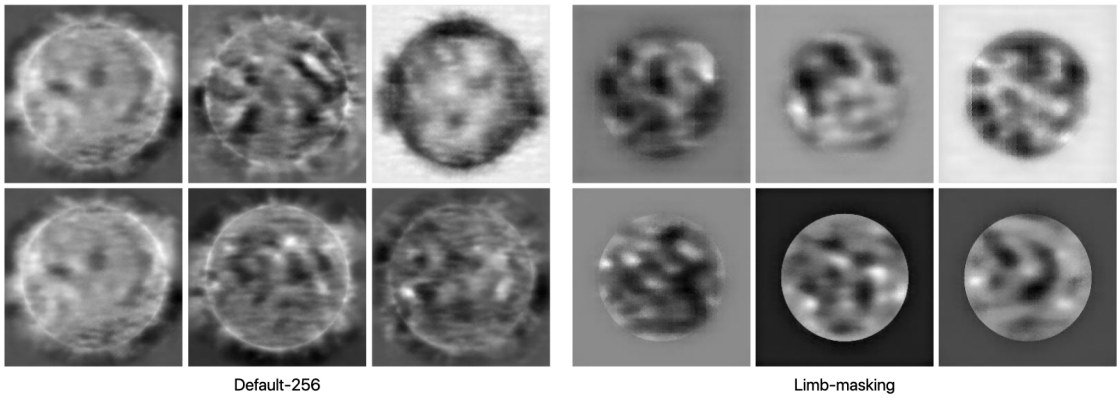


Figure 6.20: Generated images using the default-256 model (left) and the limb-masking model (right).

To determine whether the latent space is well structured, we tried to interpolate between different representations. For that purpose, we encoded two images during the solar maximum with the default-256 model obtaining μ_0 and σ_0 for the first image and μ_n and σ_n for the second image. We used linear interpolation to linearly interpolate between the two representations, obtaining μ_k and σ_k (6.1), which were then used to generate the latent representation z_k by sampling from a normal distribution. The obtained representations were decoded with the decoder and are shown in Figure 6.21. All of the representations seem reasonable and capture the transition between the two points in time. Although a linear interpolation might not properly capture the temporal domain, this shows that the latent representations for this example are reasonably robust.

$$
\begin{aligned}
t &= \frac{k}{n} \\
\mu_k &= \mu_0 + t(\mu_n - \mu_0) \\
\sigma_k &= \sigma_0 + t(\sigma_n - \sigma_0)
\end{aligned}
\tag{6.1}
$$

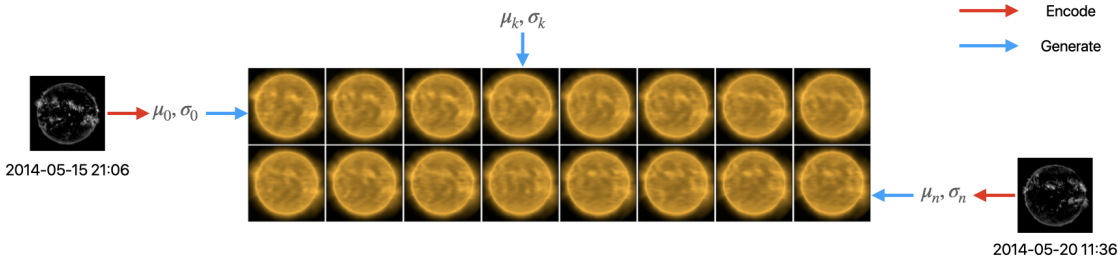


Figure 6.21: Latent space interpolations for a 5-day period with the default-256 model during the solar maximum.

Based on the analysis in this chapter, the default-256 model seems to provide the most reliable outputs and can be effectively used in a downstream application. Several applications are presented in the next chapter. Notably, the model performs very well with a comparatively small latent space (z dimensionality of 256). An even larger latent space might lead to even better representations of the input but will result in almost twice as many trainable parameters.

Chapter 7

Applications

In the context of heliophysics, the proposed architecture can be understood as a new, purely data-driven paradigm for knowledge discovery that enables a variety of use cases. Depending on the data used for training, we envision several applications for which the presented ceVAE model could provide benefits:

- **Search space reduction:** The model can efficiently filter large amounts of data and search for uncommon patterns. The model could therefore be used by solar physicists and space weather researchers to retrieve a set of relevant images that are collected by using the sample-wise anomaly score. For example, by using a two- or three-sigma rule (all images with an anomaly score twice the standard deviation above the mean are considered anomalous).[1] Similarly, the latent encodings could be used to find a set of similar images by finding images whose encoding is close to a given image. Furthermore, the pixel-wise scores can be used to find regions of interest. In this way, researchers can be directed to anomalous patterns within a given observation, even for images that are not considered anomalous by themselves. This could improve efficiency, significantly reduce the time needed to search the dataset and potentially uncover phenomena that would not have been marked as interesting by traditional algorithms or human analysis.
- **Observational quality control:** The model could be used to find potential data quality issues in existing labels (e.g., HEK annotations). This could be achieved by comparing model outputs to existing labels and pointing out the differences that are found. A similar pattern as in [41] could be used where an image is clustered in grid cells and then the intersection over union metric (IoU) is used to quantify overlap with existing bounding boxes. Alternatively, a bounding box extraction algorithm can be used, such as the traditional Teh-Chin chain approximation algorithm [72] that can be found in opencv[2] or a deep learning-based approach such as SSD [73] or YOLO [74]. The resulting differences could then be analysed by human experts to tune existing algorithms or the proposed ceVAE model.
- **Downstream ML applications:** The model can significantly compress input images. By using a latent space size of 128, a 256 x 256 pixel input image can be compressed to 128 x 2, with one latent dimension for the mean and one for the log standard deviation, which results in a compression of the input to 0.39% of its original size and for a latent space size of 256 to 0.78% of the original size. The resulting representation holds relevant information about the features observed in the source image which are potentially invariant of orientation and location and can be used in a downstream application such as a flare prediction or event detection model. Alternatively, the resulting pixel-wise anomaly segmentation map could be used to drive the attention of another model to certain parts within an image.

[1] Commonly, a three-sigma rule is used to detect outliers; however, as seen in Chapter 6, the anomaly scores are not normally distributed likely due to the nonstationary solar cycle and the data split for the test set. Therefore, using an adjusted rule of thumb is advised here.

[2] Refer to findContours

- **Anomaly detection for space weather monitoring:** The model could be used as an (almost) assumption-free agent in a space weather monitoring pipeline alongside the FFT modules providing an anomaly score that could notify space weather monitors if a certain threshold is exceeded. Again, a two-sigma could be used, or an alternative approach (e.g., an approach based on the Mahalanobis distance [75]) might provide a better threshold.
- **Smart downlink control:** A more visionary application could involve smart downlink control. Given the restricted downlink bandwidth of satellites, the model could be used for a spacecraft to decide which images or data are worth transmitting back to Earth. Notably, the power-consumption of the model might prevent the deployment to constrained devices (the same is true for other deep learning models).

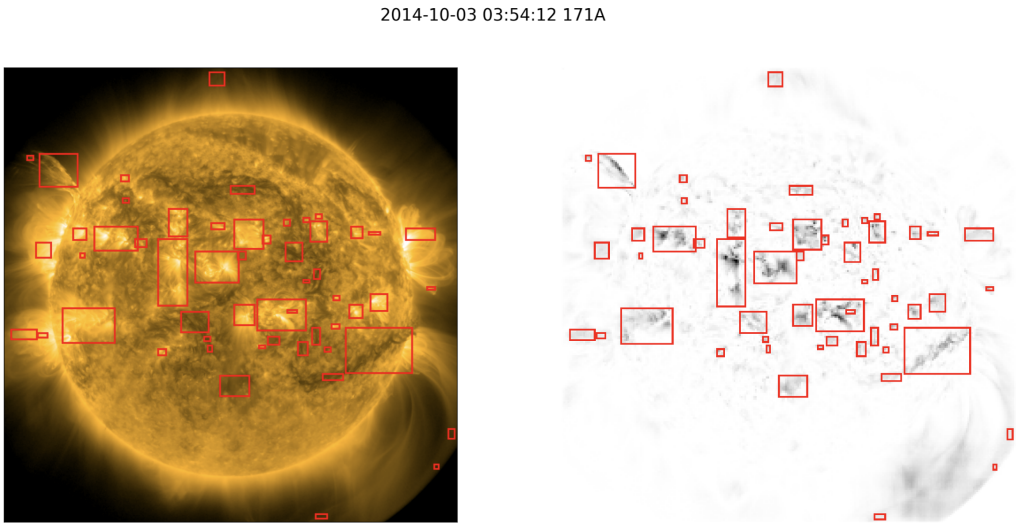


Figure 7.1: Example of a possible model application pointing out regions of interest. The image on the left is showing the source observation taken on Oct 3, 2014. The image on the right is showing pixel-level anomaly scores from the default-256 model. Both images are overlayed with bounding boxes extracted from the binarised anomaly map (Otsu's binarisation) using the Teh-Chin chain approximation algorithm from opencv.

These are just some examples for which this unsupervised approach could be applied. Notably, by training the model on different types of data (e.g., image patches instead of full-disk images), the applications differ and could even span use cases that are not listed above.

Chapter 8

Discussion

This chapter summarizes the main findings, limitations, and future directions of this thesis.

8.1 Findings

In our results, we demonstrated two things. First, we show that it is possible to extract information about solar activity with a purely unsupervised approach, which does not require any labels during training and is therefore able to overcome the need for annotations. Second, we show that the use of a hybrid model, combining a generative model (VAE) and a deterministic model (CE) at training time, leads to more robust latent representations for solar EUV images. From the results, it is clear that the data-driven ceVAE model is able to offer more insights into solar activity than a naive baseline model and is able to learn features beyond pure intensities. The *default-256* model performed best, providing suitable reconstructions and anomaly scores. Masking the limb did not seem to work as well and might introduce undesirable nonphysical artefacts and therefore requires further investigations.

Overall, the image-level anomaly scores for the ceVAE model align well with solar activity, which is shown by the comparison of the anomaly scores with the GOES X-ray flux (see Figure 6.3). The image-level scores were high during the solar maximum and low during the solar minimum, with a few exceptions that are mostly caused by instrument anomalies. When more closely examining specific years, the time series comparison between anomaly scores and GOES X-ray flux shows certain lags (see Figure 6.5) which should be more closely investigated to determine whether the deviations have a physical explanation or are caused by model assumptions. If there is a physical justification this could be useful to solar physicists studying the solar dynamo. Furthermore, anomalies identified as instrument artefacts might be excluded or handled with special care in future datasets to prevent ML models from learning nonphysical properties. The images for the highest scores, shown in Figure 6.8, expose a great deal of unusual activity which could also be very interesting. In particular, the parts of the activity that are not yet well understood or documented could form the basis for further studies by heliophysicists. Notably, the ceVAE model is able to extract information that might not be apparent to a human by just looking at the observations. One example is the periodicity of the anomaly score during quiet periods (long-term trends); another example is the solar activity reaching into space which might easily be overlooked (unusual local activity). Including multiple AIA channels in the model might even increase the utility of an exploratory search for interesting phenomena. To determine which images are marked as out of distribution, a manually-assigned anomaly threshold can be defined (e.g., all scores above the mean of the score). Alternatively, the mean and variance during training can be stored and compared with means and variances at prediction time in order to define a dynamic threshold that determines which scores are to be treated as anomalous (e.g., using the Mahalanobis distance [75]). By using a different dataset to train the model, the "normal" state of the system is shifted, and other images are flagged as anomalous, which allows different kinds of use cases for the model.

The pixel-level anomaly scores (anomaly segmentation map) make it possible to localize anomalies within an image and help to explain why the model considers a given image anomalous. Making use of the model-internal variations by combining reconstruction and density-based scores seems to work well for AIA images and theoretically offers more meaningful anomaly scores than scores purely based on reconstruction errors. To effectively extract solar activity that is not directly related to intensity (such as magnetic activity that reaches out into space), a purely threshold-based method such as the proposed baseline model seems to be insufficient. In contrast, the ceVAE model seems to be well suited for such a purpose and is able to capture unusual activities regardless of the intensity.

We asked ourselves whether the model is able to detect solar events. The answer is likely no because we cannot guarantee that the ceVAE model is actually finding events. Instead, the model is able to detect regions of interest that coincide with solar activity. However, this is a first important step towards unsupervised event detection. A similar conclusion was reached by Banda et al. [41], who showed a significant overlap with events reported in HEK. Notably, the comparison with events from HEK requires a careful selection of the event types because the events that should be included to compute the evaluation score highly depend on the wavelength, observatory and algorithm that were used to detect the event. Ending up with a representative score is not trivial because multiple overlapping event observations (space, time, and wavelength) and the spatiotemporal alignment of the events with the preprocessed images need to be taken into consideration to calculate overlaps.[1] For these reasons, the comparison of the identified out-of-distribution events with existing annotations in HEK is left to future work and should likely be conducted in collaboration with a domain expert.

Deep learning seems to be an effective way to learn a suitable feature representation from solar EUV images without relying on hand-crafted features. This comes at the cost of computational power and a lack of interpretability because the extracted features are no longer easily understood by humans. This makes it generally difficult to assess what the model has learned. Generating new images and interpolating in the latent space proved to be useful debugging tools for understanding the model-internal latent representations.

In summary, our findings indicate that an unsupervised approach to learn a lower-dimensional representation of the sun, detect out-of-distribution samples, and localize out-of-distribution regions using the ceVAE model produces useful outputs and can be used to extract semantic information from SDO AIA images. This is consistent with the findings by Zimmerer et al. who originally applied the ceVAE model to medical imaging [1, 47]. However, there are some limitations to this study that must be considered.

[1] For example, when using the SDO ML v2 dataset, the HEK events need to be rescaled and repositioned based on the preprocessing steps used to create the dataset.

8.2 Limitations

A major limitation of this study is the lack of a benchmarking dataset that would make it possible to provide more formal guarantees for the identified anomalies. Compared to other domains such as medical imaging where several benchmarking datasets exist,[2] the heliophysics domain does not yet have any comparable datasets. This can be explained by the facts that there are many subdisciplines in heliophysics targeting different areas of solar physics and there are a large number of different data sources used for very specific purposes. Finding a common definition for "anomaly" and therefore a common dataset is likely not possible nor purposeful. Unsurprisingly, this makes unsupervised approaches that can be used as a generic framework and easily applied to different areas, ever more powerful and necessary.

In this study, we made use of the SDO machine learning dataset v2 (SDO ML v2), which is the second iteration of a dataset prepared by Galvez et al. [67]. The dataset already applies several important corrections and does a temporal and spatial downsampling of the raw data, which simplifies the preprocessing substantially. However, in order to make use of the data in a deep learning setting, we still needed to do a great deal of engineering and apply several preprocessing steps. We hope that future researchers will be able to benefit from the ideas and source code that were created as part of this study (refer to Appendix A.2). Notably, relying on curated datasets or the level 1.5 data available via JSOC can also bear the risk of missing short-lived phenomena as shown in a recently published study by Young et al. [69] which analyses spikes in AIA data. Problems mainly occur if the instrument correction pipelines (in this case, the AIA despiking algorithm and the preprocessing steps for the SDO ML dataset) remove real solar features. Young et al. advise that researchers always restore the original intensities in AIA data when studying short-lived or rapidly evolving features (this is called *respiking*). This has to be taken into consideration before applying machine learning models such as the ceVAE model to an operational scenario. However, this task cannot be outsourced to the data scientists because it requires a lot of knowledge about the instruments and the data-processing pipelines before the final data product and therefore requires close collaboration with domain experts. Similarly, other preprocessing steps, such as the clamping, rescaling, and normalization applied in this study could lead to a physically incorrect representation of the sun being learned and should be closely reviewed before physical knowledge is inferred.

We were able to show the successful transfer of a VAE-based model from the medical domain; however, there is still little evidence that this is the best-performing approach. Usually, generative models such as VAEs are worse at discriminating tasks (e.g., classification). Because there is a lack of discriminative information (i.e., lack of annotated data), one cannot rely on a discriminative model. A model like the ceVAE, which captures information about the data generation process and therefore the underlying data distribution, seems to be theoretically well justified. Still, there might be other model architectures that offer similar guarantees which might work better in the solar domain (e.g., GAN-based models as discussed in [52], a transformer-based model as seen in [50] or a self-supervised approach as proposed in [54]). Investigating different approaches is therefore a logical next step. Similarly, the model assumptions of the ceVAE model might require additional checking. For example, we used the Mean Absolute Error (MAE) as the reconstruction loss function for training the model; however, other loss functions might perform better (e.g., by using a feature perceptual loss [76]), and in the best case, the loss function is physically informed and incorporates physical knowledge. Furthermore, in our analysis, we showed that the reconstruction quality of the ceVAE model is not optimal, mostly due to the blurriness induced by the variational autoencoder. Having better reconstructions would also automatically improve the quality of the pixel-wise anomaly scores because the reconstruction error still plays a fundamental role when using the "combi" score mode. For this reason, measures to improve reconstruction quality should be taken into consideration (e.g., by applying deep feature consistent learning and generative adversarial training as proposed by Hou et al. [77]).

[2] For example, the Medical Out-of-Distribution Analysis Challenge which was part of the original ceVAE evaluation; http://medicalood.dkfz.de/web/

8.3 Future Research

Future research is necessary to confirm the types of conclusions that can be drawn from this study. In particular, the overall model performance should be validated in more detail, and the extent to which the model outputs align with existing knowledge should be examined (e.g., by conducting more investigations with existing HEK annotations or by domain expert validation). Moreover, it should be verified whether using a larger latent space size can yield better findings and whether the prior assumptions for the VAE, $\mathcal{N}(0, 1)$, are suitable for solar data. Additionally, it should be determined how to incorporate the model outputs in downstream applications.

Furthermore, the learned feature representation should be analysed in more detail, such as by using grad-cam [78] to better understand the process of representing solar features by the convolutional encoder (CNN). Additionally, the model-internal latent representations should be understood better, ideally, by clustering the model variations in a way that will make it possible to classify events seen in an image (e.g., active regions or coronal holes). Labels are scarce, but they still exist. A semi-supervised approach might be employed to guide the model and make use of the existing knowledge, possibly leading to a more understandable structure in the latent space and thereby increasing interpretability (feature consistency).

Future studies should aim to replicate results for different EUV wavelengths, possibly even combining the wavelength bands (multi-channel model), and explore the application of the approach to HMI magnetograms. Furthermore, the temporal domain could be included by adjusting the model architecture to work on a series of images rather than just a single point in time. Apart from investigating full-disk images, future research should also target specific event types such as active regions. An intriguing question for active regions that could be studied with the ceVAE model is *In what way are active regions that produce a flare within the next 24 hours anomalous?*. This study could be carried out by training on active regions that are not flaring and then applying the model to flaring active regions. In this way, the model could even be used for flare prediction. Notably, this will require a dataset analogous to SDO ML v2 with SHARPS (Spaceweather HMI Active Region Patch [SHARP])[3] and might also require better data sampling strategies that do not only depend on a temporal splitting of the data but also take the distribution of flaring events into consideration. Similar questions could be studied for other event types as well. Future research should also more carefully consider the potential effects of preprocessing and data normalization both during training and at prediction time that might limit the capability to draw conclusions about the physical processes on the sun.

[3] Ideally, the dataset should use the same preprocessing steps as the SDO ML v2 dataset to yield a fair comparison.

Chapter 9

Conclusion

In this study, we applied a purely unsupervised deep learning approach based on a variational autoencoder (VAE) to solar extreme ultraviolet (EUV) images and show that such a model can aid knowledge discovery with a minimal set of assumptions. More concretely, we transferred a method from medical imaging to heliophysics that combines a VAE and a context encoder (CE; the so-called context-encoding variational autoencoder [ceVAE]), to learn a lower-dimensional representation from solar EUV images observed by the Atmospheric Imaging Assembly (AIA) onboard NASA's Solar Dynamics Observatory (SDO). The resulting model is able to find out-of-distribution images and pixels, allowing it to find anomalous and "interesting" solar activity. Thus, the model opens a wide range of use cases. The most obvious use case is to narrow the search space and discover potentially overlooked phenomena and correlations in the vast amounts of solar data accumulated to date (novelty and anomaly detection), thereby assisting space weather researchers and solar physicists in the process of knowledge discovery. A second use case is to provide a tool for observational quality control and hence help to improve existing algorithms (e.g., modules in the SDO Event Detection System). Finally, the model could serve as a potential input for downstream machine learning applications offering both a lower-dimensional representation of the sun as well as an anomaly segmentation map that could be used to guide model attention.

The findings suggest that the combination of a VAE and a CE works well in heliophysics and yields good results on both the image and pixel levels while offering a robust lower-dimensional encoding of inputs. The model is able to extract interesting solar images and regions without requiring any labels during training and is therefore able to overcome the need for costly annotations for certain use cases. With ever-increasing amounts of data collected by current and future missions, this might become even more important. Notably, by using this purely data-driven approach, the whole dataset can be leveraged, and less-studied periods can also be included. This could also be beneficial for missions, for which existing algorithms have not yet been fine tuned. The findings of this thesis once again confirm that the transfer from the medical to the solar domain generally works well and should be taken into consideration for segmentation and anomaly detection tasks.

It is rather difficult to compete with more traditional methods such as the FFT modules in the SDO Event Detection System, because of the number of assumptions and amount of domain knowledge that have been put into the tuning of these algorithms. A major blocker for using machine learning-based approaches in operational scenarios is the complexity of the evaluation and the lack of expressive metrics (especially in the absence of reliable ground truth labels). This is also true for the approach proposed in this thesis. Nevertheless, the basic finding that the ceVAE model works well for extracting regions of interest by only making use of a limited set of assumptions makes it an attractive alternative that is likely more robust and generalisable than a more traditional, less data-driven approach. We strongly believe that using unsupervised methods will become more important in solar physics as the amount of data grows and because these methods provide an elegant way of overcoming a lack of annotated

data. Collaborating with domain experts to include meaningful data into such models and increase the interpretability of their output will be crucial for the success of these purely data-driven approaches.

This thesis serves as an important step towards unsupervised event detection in heliophysics and lays the basis for further studies to realize the full potential of the available data.

List of Figures

Appendix A

Software

A.1 awesome-helio: A Curated List of Datasets, Tools and Papers for Machine Learning in Heliophysics

There are a lot of great resources contributing to the state of the art in heliophysics but they are often hard to find. For this reason, we introduced *awesome-helio*[1]. *awesome-helio* is a curated list of datasets, tools and papers for machine learning in heliophysics and should help researchers to quickly find and navigate to the different solar physics resources on the web.

A.2 sdo-cli: A Practitioner's Utility for Working with SDO Data

The code for *sdo-cli* including a solar out-of-distribution model based on the context-encoding variational autoencoder is implemented in the following repository: https://github.com/i4Ds/sdo-cli.

A full end-to-end anomaly detection pipeline example can be found here: https://github.com/i4Ds/sdo-cli/blob/master/notebooks/ce-vae__e2e-pipeline.ipynb

A.2.1 sdo-cli Commands

Data Loading

We downloaded the SDO ML v2 dataset from Google Cloud to the FHNW infrastructure (NAS) using gsutil. An example for downloading the small version of the dataset is shown in Listing A.1.

```
gsutil -m cp -r gs://fdl-sdoml-v2/sdomlv2_small.zarr/ /astrodata01/sdomlv2_small
```

Listing A.1: Example for downloading SDO ML v2 data

To make use of the data in a machine learning model, we have implemented a PyTorch DataLoader with various configuration and filtering options. The source code can be found in the *sdo-cli* repository under src/sdo/sood/data/sdo_ml_v2_dataset.py. An example for using the loader is shown in Listing A.2.

[1] https://github.com/i4Ds/awesome-helio

```python
from sdo.sood.data.sdo_ml_v2_dataset import SDOMLv2DataModule

batch_size = 8
target_size = 256
input_shape = (batch_size, 1, target_size, target_size)

data_module = SDOMLv2DataModule(storage_root="/mnt/nas05/astrodata01/astroml_data/sdomlv2_full/ sdomlv2.zarr",
                                storage_driver="fs",
                                num_workers=16,
                                pin_memory=False,
                                target_size=input_shape[2],
                                batch_size=batch_size,
                                prefetch_factor=8,
                                channel="171A",
                                train_year=["2010", "2012", "2013",
                                            "2015", "2017", "2018", "2020"],
                                test_year=["2011", "2014", "2016", "2019"],
                                mask_limb=False,
                                reduce_memory=True)
```

Listing A.2: Example for using the SDOMLv2DataModule

In *sdo-cli*, we provide a command to download GOES data. We make use of Dask[2] (distributed data frames) using the Parquet file format to allow efficient access to the time series. Since Sunpy was lacking some of the functionality, we implemented it locally and submitted two pull requests #6260 and #6247.

```
sdo-cli goes download \
    --start=2010-01-01T00:00:00 \
    --end=2020-12-31T23:59:59 \
    --output=./output/goes
```

Listing A.3: Example for downloading GOES data

A key success factor for efficiently working with HEK events, despite the useful APIs in Sunpy, is to cache the events locally in order to gain fast access. This can be achieved with the *sdo-cli events get* command:

```
# first start the database
docker-compose up -d
sdo-cli events get \
    --start=2010-01-01T00:00:00 \
    --end=2020-12-31T23:59:59 \
    --event-type="AR" \
    --db-connection-string="postgresql://sdouser:password@localhost:5432/postgres"
```

Listing A.4: Example for downloading HEK events

Training

To train a ceVAE model, the *sdo-cli sood ce_vae train* command can be used. The configuration is passed as a file and different examples including the ones used to retrieve the results in this work can be found in the config directory in the root of the sdo-cli project.

```
sdo-cli sood ce_vae train \
    --config-file="./sdo-cli/config/ce-vae/run-fhnw-full-2-256.yaml"
```

Listing A.5: Example for training a ceVAE model

[2] https://www.dask.org/

Prediction

Predicting image-level and pixel-level scores is possible with the *sdo-cli sood ce_vae predict* command. Notably, this requires a pretrained model.

```
sdo-cli sood ce_vae predict \
    --config-file="./sdo-cli/config/ce-vae/run-fhnw-full-2-256-predict.yaml" \
    --predict-mode=sample
```

Listing A.6: Example for predicting image-level scores with the ceVAE model

```
sdo-cli sood ce_vae predict \
    --config-file="./sdo-cli/config/ce-vae/run-fhnw-full-2-256-predict.yaml" \
    --predict-mode=pixel
```

Listing A.7: Example for predicting pixel-level scores with the ceVAE model

Image Generation

Generating new images with the ceVAE model is possible with the *sdo-cli sood ce_vae generate* command. Notably, this also requires a pretrained model.

```
sdo-cli sood ce_vae generate \
    --config-file="./sdo-cli/config/ce-vae/run-fhnw-full-2-256-predict.yaml"
```

Listing A.8: Example for generating new images with the ceVAE model

A.2.2 Model Checkpoints

A set of model checkpoints used to produce the results in this work can be found under the following links:

- **default-128** (1.3GB): https://drive.google.com/file/d/1g4VP3AvYXUjG1nMn-GbuMCjtfruQVoke/view
- **default-256** (2.5GB): https://drive.google.com/file/d/16RIjrL6SkxzvDSU74OITCxyROuY4JEHX/view
- **limb-masking** (2.5GB): https://drive.google.com/file/d/1gAKirNqRXhEzmfRgYYW9aPLzWsMomphF/view
- **standard vae** (1.3GB): https://drive.google.com/file/d/1gDL8h0Te1V5sxhAGTFxbyd0qiHOWMywB/view

A.2.3 Jupyter Notebooks

Several Jupyter notebooks were created to allow reproducing the results of this thesis. The notebooks can be found under *notebooks/sdo_ml_v2*[3] in the sdo-cli repository. Most of the outputs were cleared to save space but are available on request from the author. Two examples are *ceVAE - Anomaly Analysis run-fhnw-full-2-256.ipynb* to reproduce the analysis of the default-256 model and *Generate Interpolations.ipynb* to analyze the latent space through interpolation between different encodings.

[3] https://github.com/i4Ds/sdo-cli/tree/main/notebooks/sdo_ml_v2

Appendix B

Additional Results

B.1 Additional Results from the default-256 Model

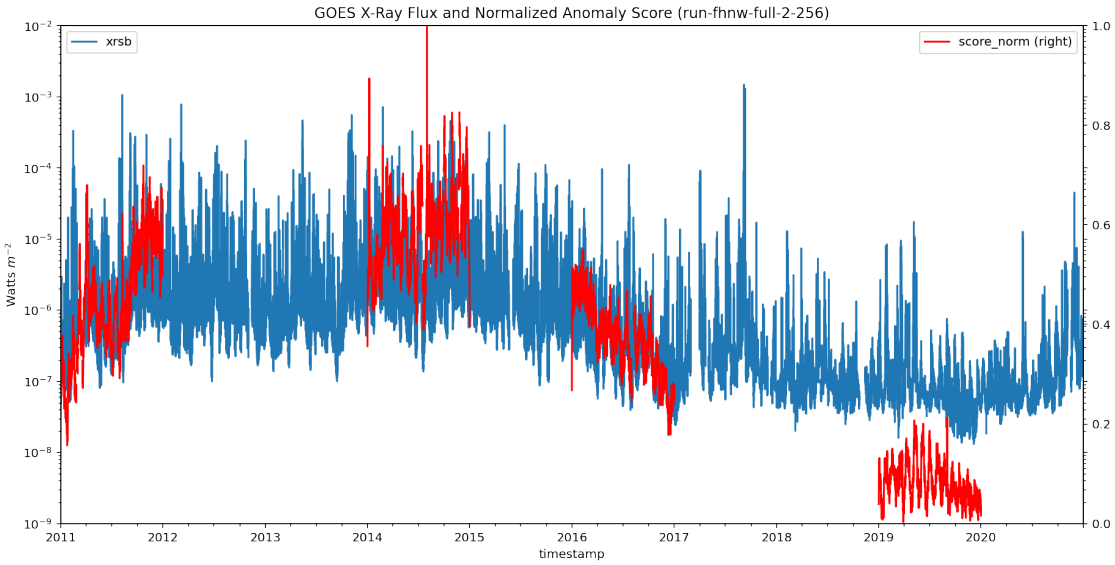


Figure B.1: Alignment of Goes X-ray flux and normalised anomaly score for the default-256 model.

B.2 Additional Results from the limb-masking Model

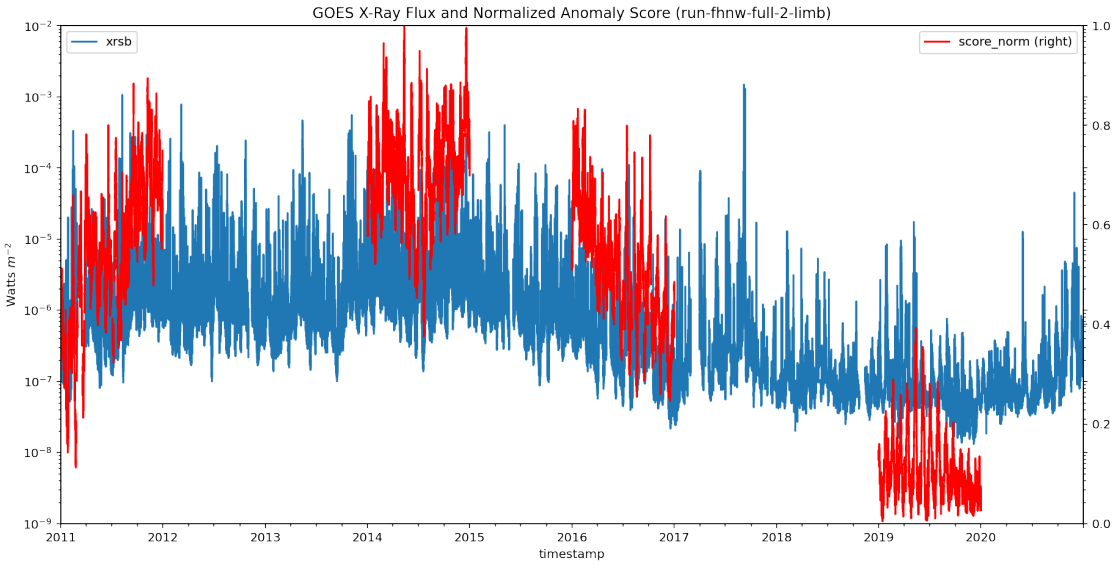


Figure B.2: Alignment of Goes X-ray flux and normalised anomaly score for the limb-masking model.

B.3 Results from the Baseline Model

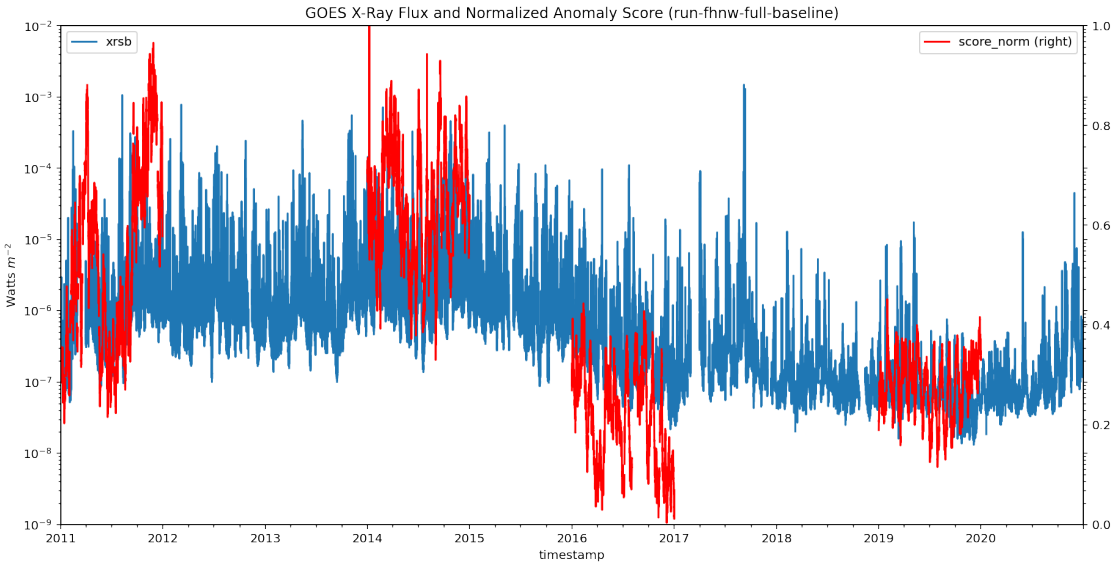


Figure B.3: Alignment of Goes X-ray flux and normalised anomaly score for the baseline model.

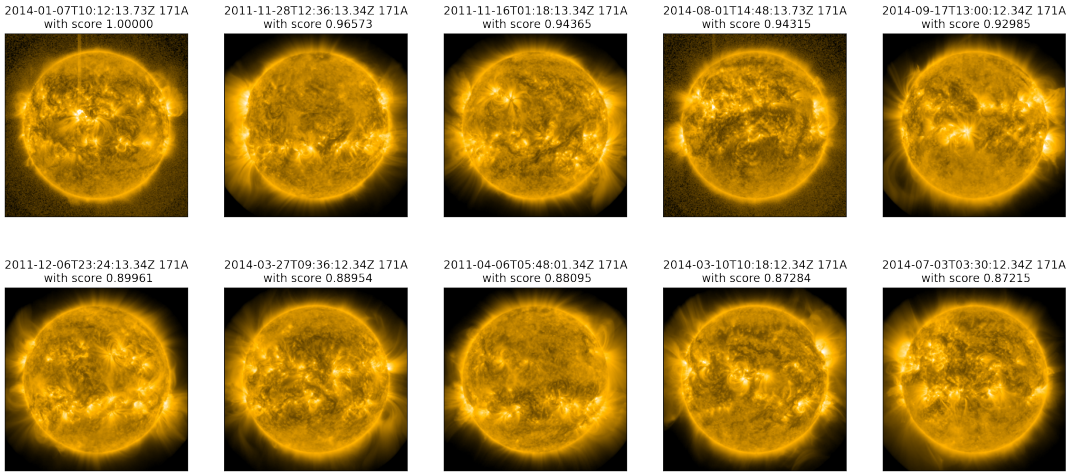


Figure B.4: Top-scoring image-level predictions for the baseline model.

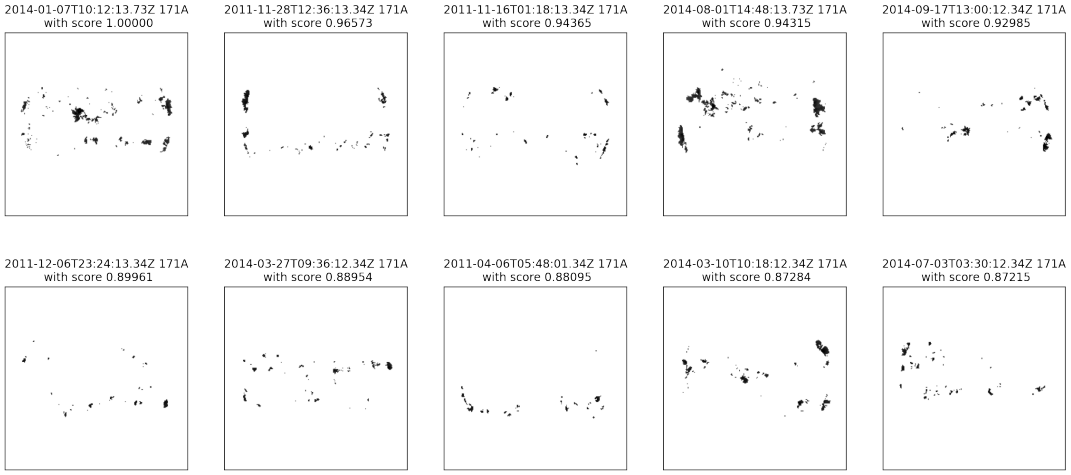


Figure B.5: Pixel-level predictions for high-scoring image-level scores for the baseline model.

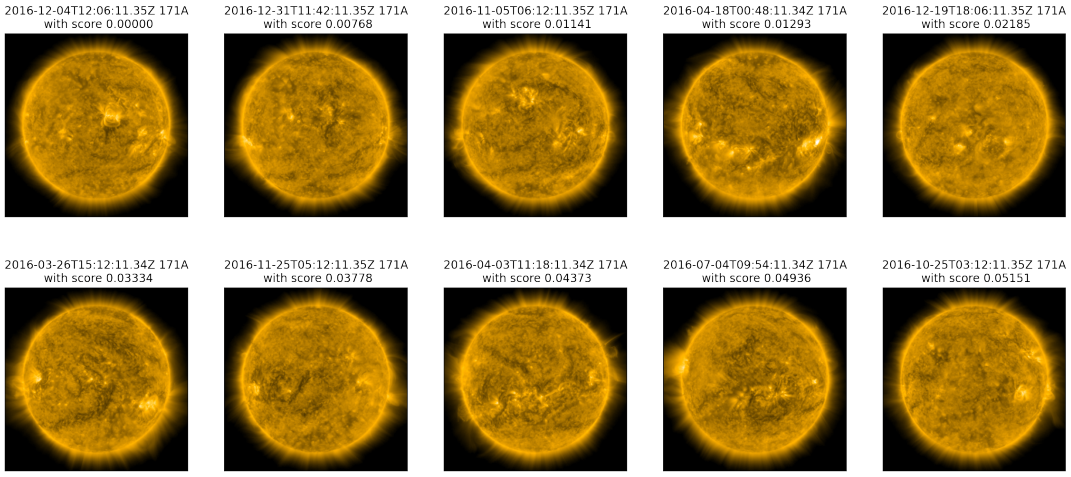


Figure B.6: Low-scoring image-level predictions for the baseline model.

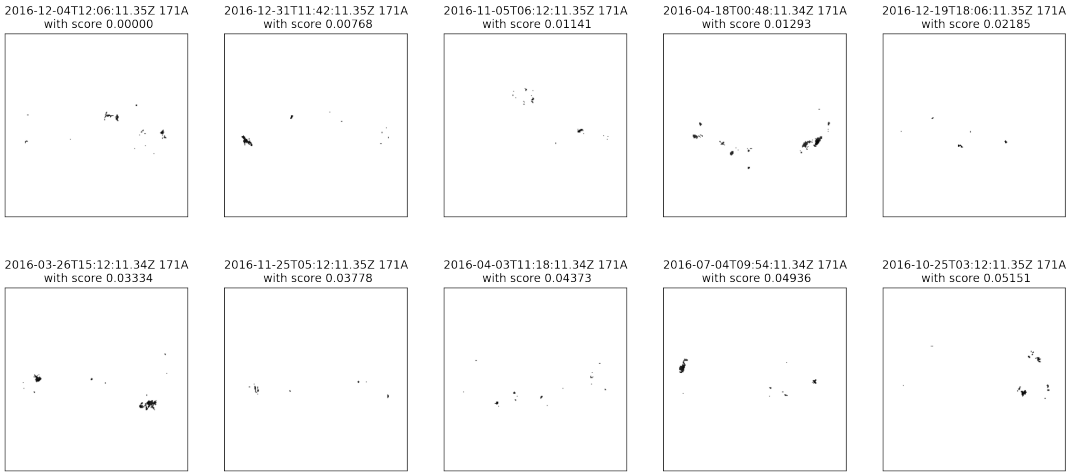


Figure B.7: Pixel-level predictions for low-scoring image-level scores for the baseline model.

B.4 Results from the default-128 Model

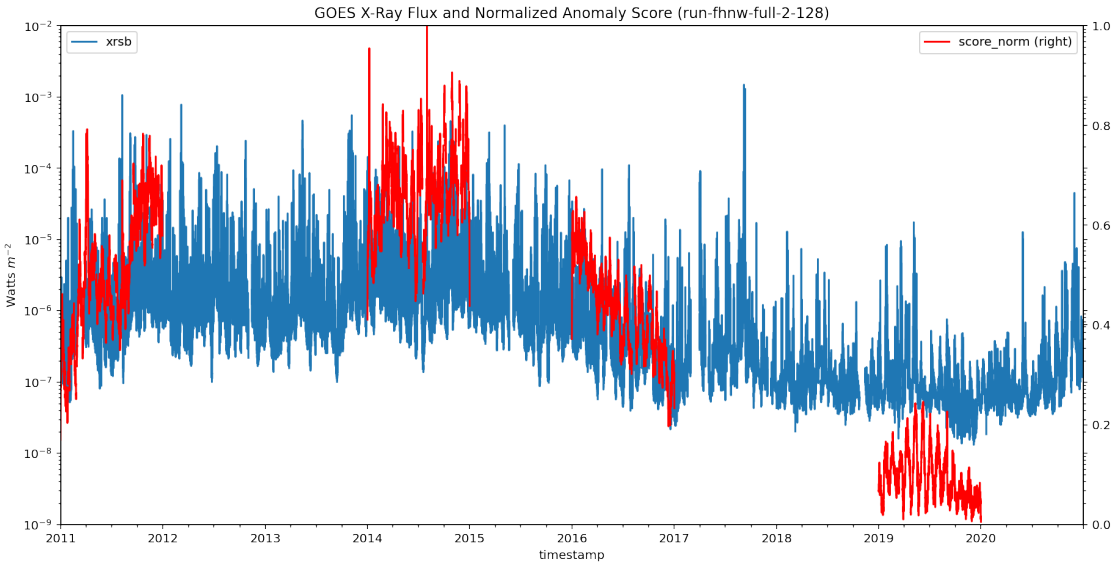


Figure B.8: Alignment of Goes X-ray flux and normalised anomaly score for the default-128 model.

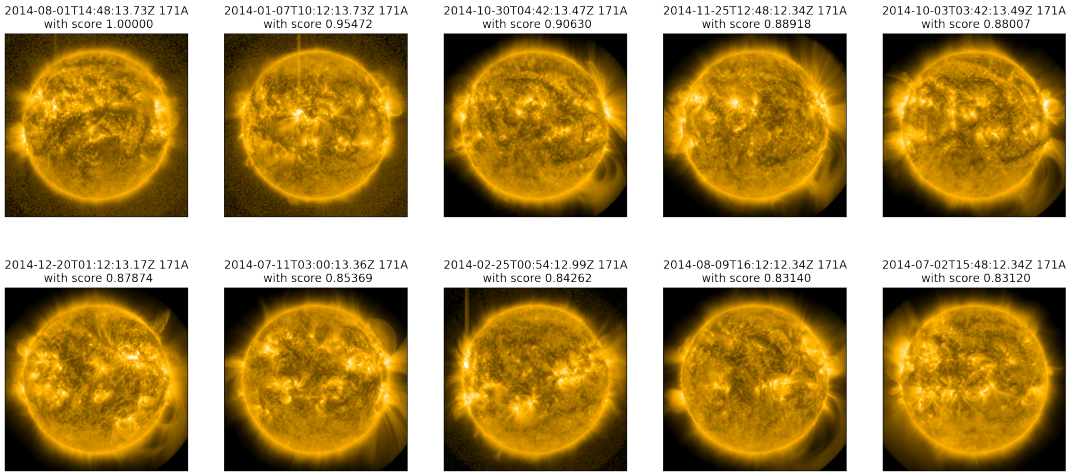


Figure B.9: Top-scoring image-level predictions for the default model (at least 1 week between different observations).

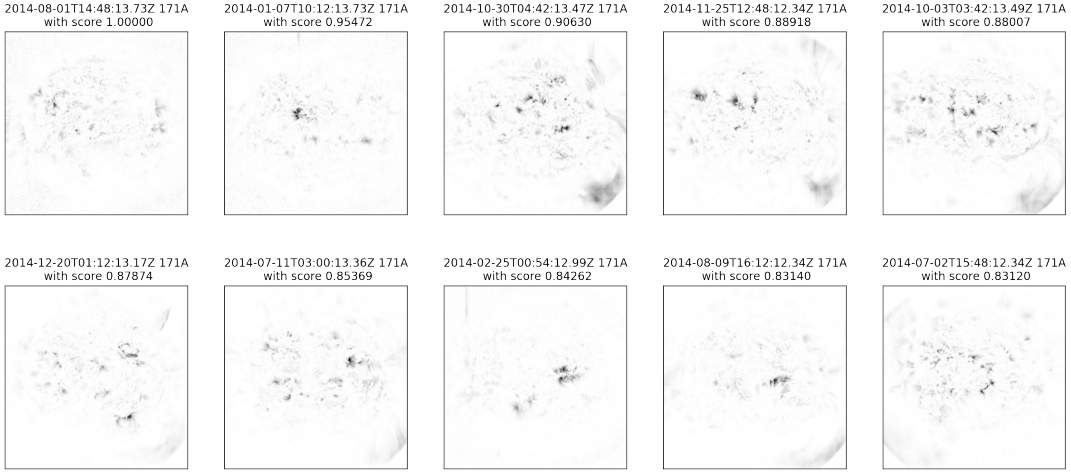


Figure B.10: Pixel-level predictions for high-scoring image-level scores from the default model.

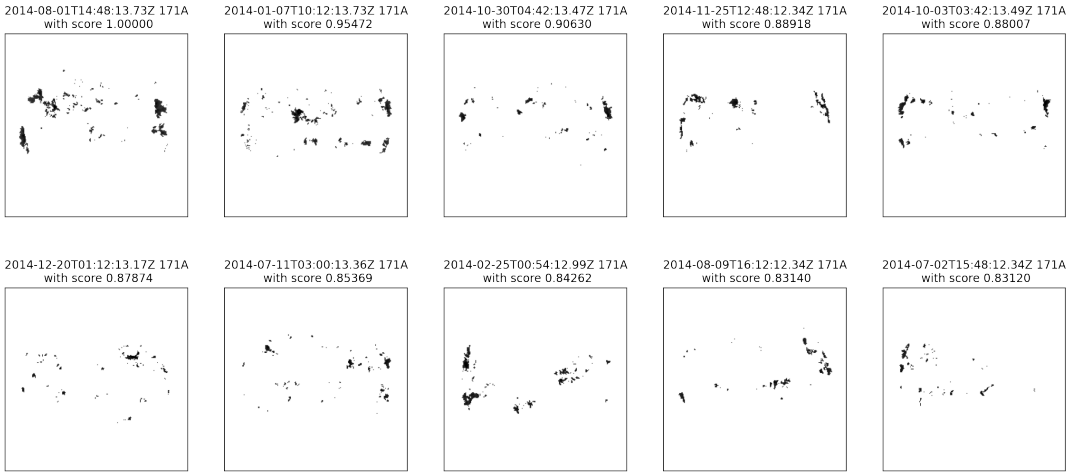


Figure B.11: Pixel-level predictions for high-scoring images-level scores for the baseline model.

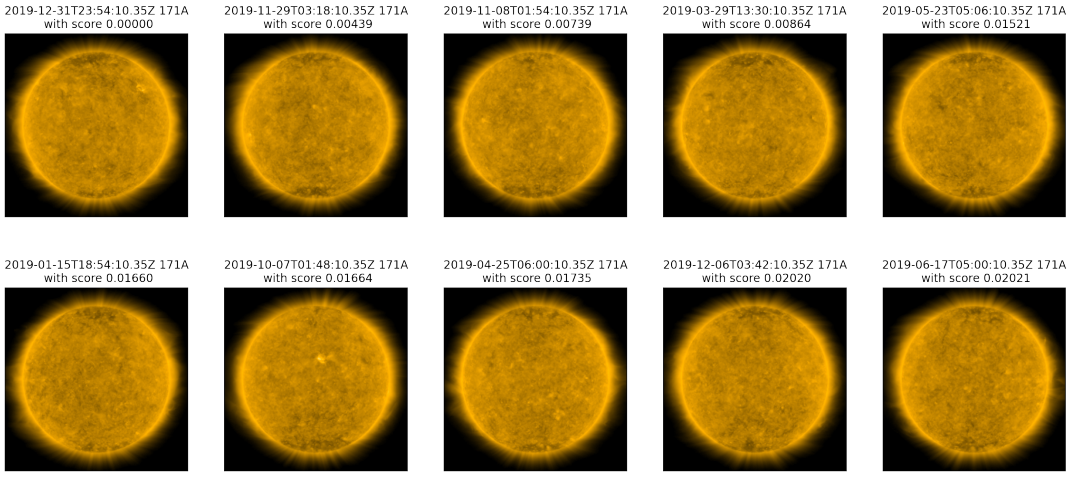


Figure B.12: Low-scoring image-level predictions for the default model (at least 1 week between different observations).

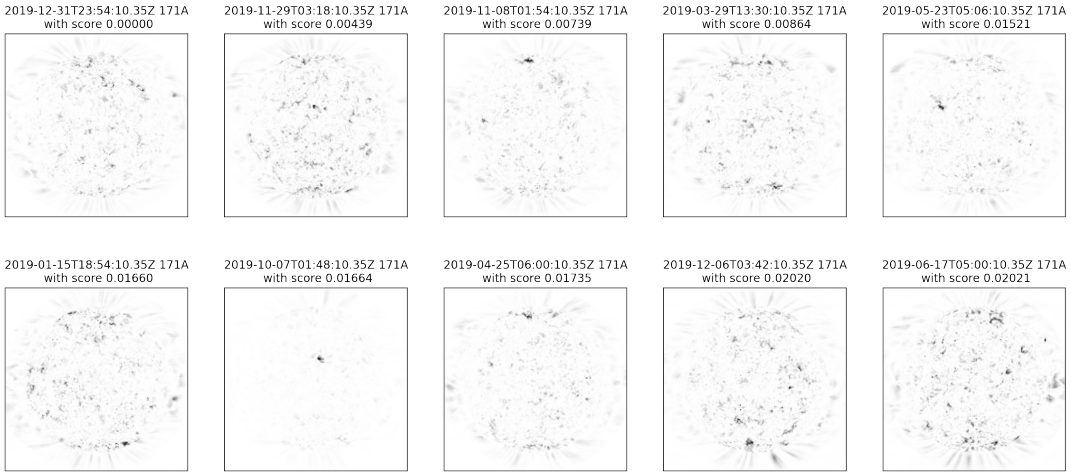


Figure B.13: Pixel-level predictions for low-scoring image-level scores from the default model.

B.5 Results from the VAE Model

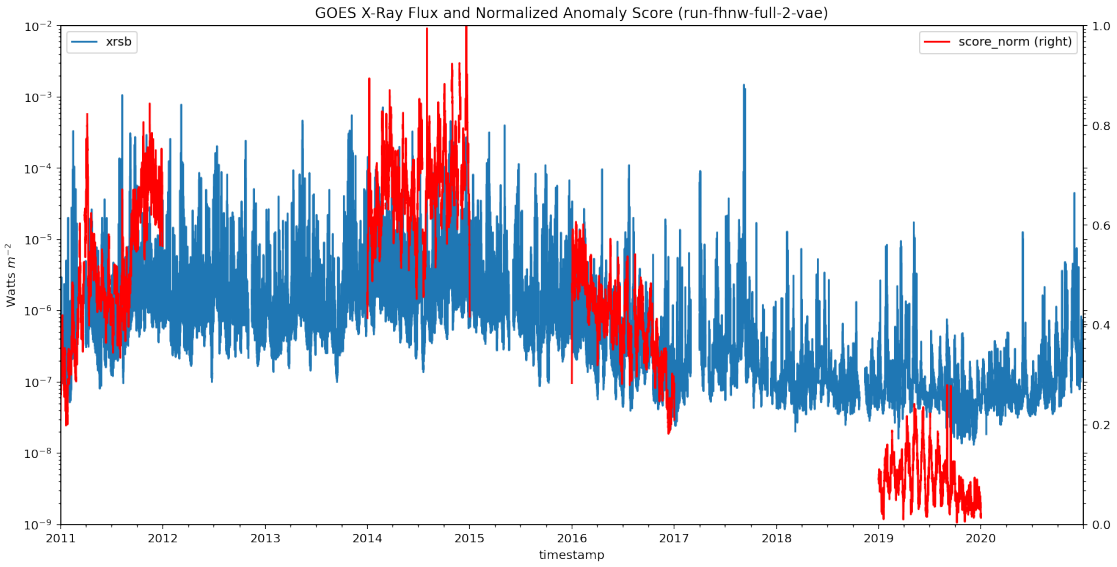


Figure B.14: Alignment of Goes X-ray flux and normalised anomaly score for the standard vae model.

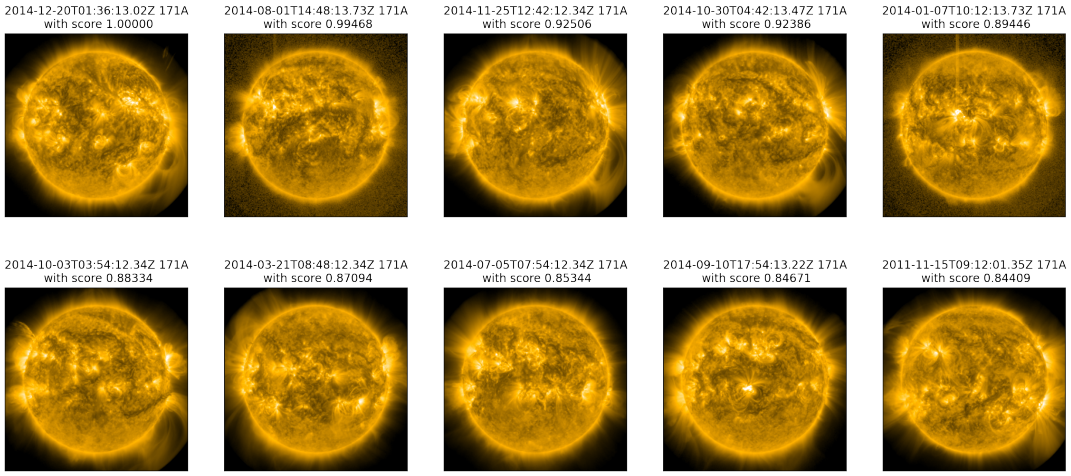


Figure B.15: Top-scoring image-level predictions for the standard vae model (at least 1 week between different observations).

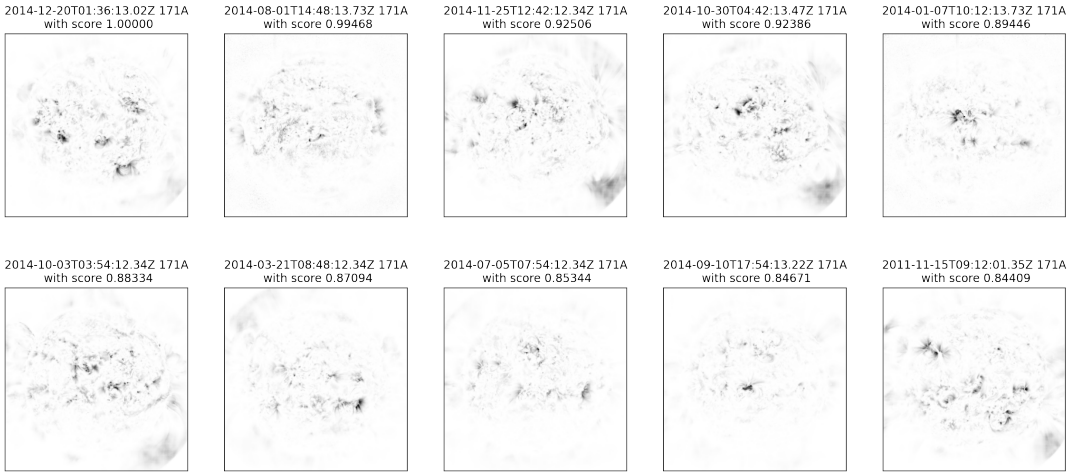


Figure B.16: Pixel-level predictions for high-scoring image-level scores from the standard vae model.

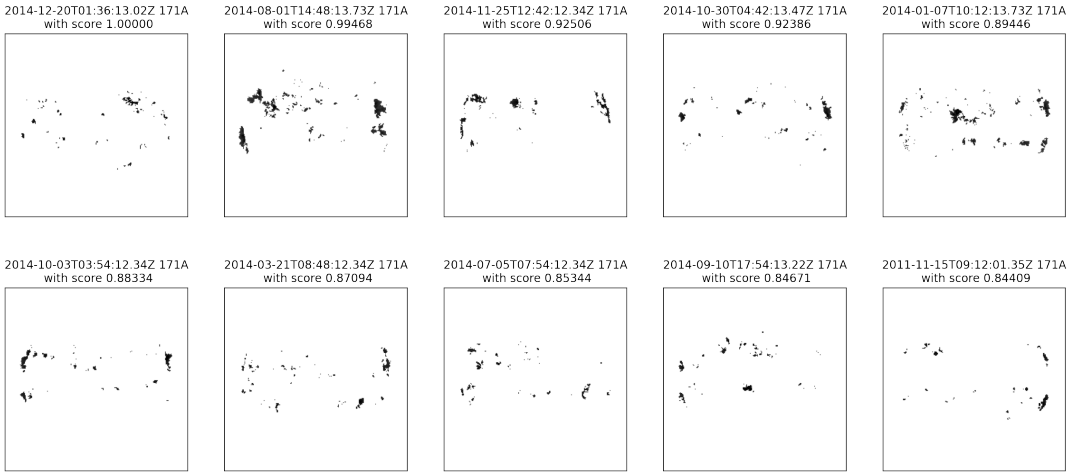


Figure B.17: Pixel-level predictions for high-scoring image-level scores for the baseline model.

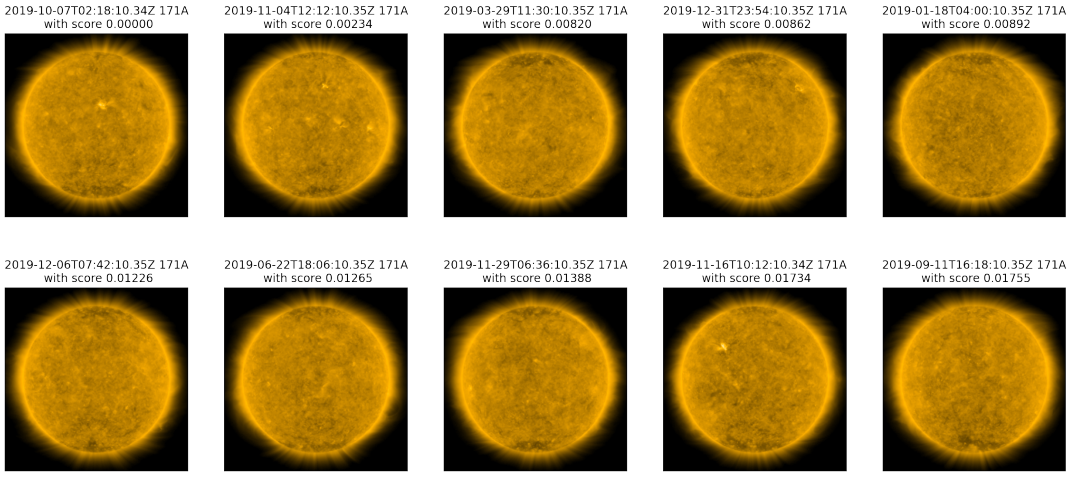


Figure B.18: Low-scoring image-level predictions for the standard vae model (at least 1 week between different observations).

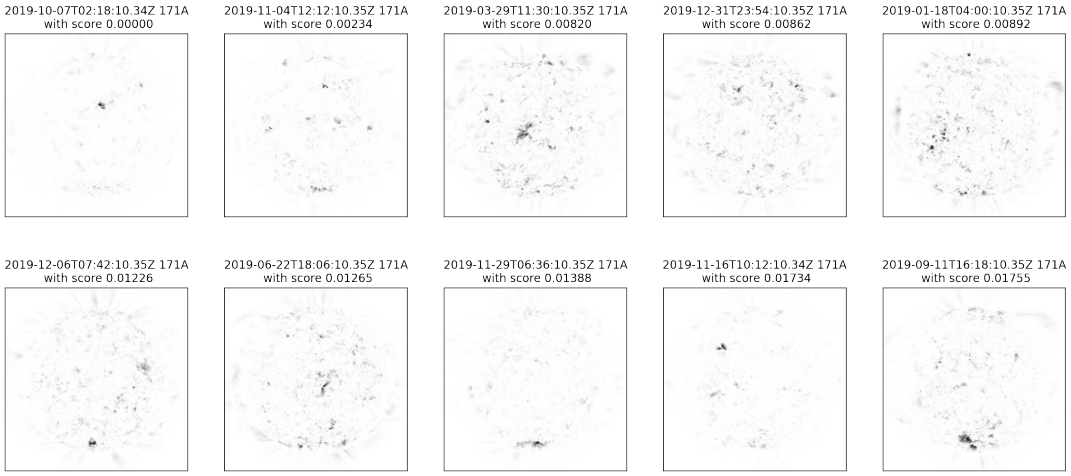


Figure B.19: Pixel-level predictions for low-scoring image-level scores from the standard vae model.

Bibliography

[1] D. Zimmerer, S. A. Kohl, J. Petersen, F. Isensee, and K. H. Maier-Hein, "Context-encoding variational autoencoder for unsupervised anomaly detection," *arXiv:1812.05941*, 2018.

[2] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu, "Coca: Contrastive captioners are image-text foundation models," *arXiv preprint arXiv:2205.01917*, 2022.

[3] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[4] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel *et al.*, "Mastering chess and shogi by self-play with a general reinforcement learning algorithm," *arXiv preprint arXiv:1712.01815*, 2017.

[5] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[6] I. Goodfellow, Y. Bengio, and A. Courville, *Deep learning*. MIT press, 2016.

[7] H. C. Arp, "The hertzsprung-russell diagram," in *Astrophysics II: Stellar Structure/Astrophysik II: Sternaufbau*. Springer, 1958, pp. 75–133.

[8] X. Zhentao, "The basic forms of ancient chinese sunspot records," *Chinese Science*, vol. 9, pp. 19–28, 1989.

[9] W. D. Pesnell, B. J. Thompson, and P. Chamberlin, "The solar dynamics observatory (sdo)," in *The solar dynamics observatory*. Springer, 2011, pp. 3–15.

[10] P. H. Scherrer, J. Schou, R. Bush, A. Kosovichev, R. Bogart, J. Hoeksema, Y. Liu, T. Duvall, J. Zhao, C. Schrijver *et al.*, "The helioseismic and magnetic imager (hmi) investigation for the solar dynamics observatory (sdo)," *Solar Physics*, vol. 275, no. 1, pp. 207–227, 2012.

[11] J. R. Lemen, D. J. Akin, P. F. Boerner, C. Chou, J. F. Drake, D. W. Duncan, C. G. Edwards, F. M. Friedlaender, G. F. Heyman, N. E. Hurlburt *et al.*, "The atmospheric imaging assembly (aia) on the solar dynamics observatory (sdo)," in *The solar dynamics observatory*. Springer, 2011, pp. 17–40.

[12] T. Woods, F. Eparvier, R. Hock, A. Jones, D. Woodraska, D. Judge, L. Didkovsky, J. Lean, J. Mariska, H. Warren *et al.*, "Extreme ultraviolet variability experiment (eve) on the solar dynamics observatory (sdo): Overview of science objectives, instrument design, data products, and model developments," *The solar dynamics observatory*, pp. 115–143, 2010.

[13] P. Martens, G. Attrill, A. Davey, A. Engell, S. Farid, P. Grigis, J. Kasper, K. Korreck, S. Saar, A. Savcheva *et al.*, "Computer vision for the solar dynamics observatory (sdo)," *Solar Physics*, vol. 275, no. 1, pp. 79–113, 2012.

[14] N. Hurlburt, M. Cheung, C. Schrijver, L. Chang, S. Freeland, S. Green, C. Heck, A. Jaffey, A. Kobashi, D. Schiff, J. Serafin, R. Seguin, G. Slater, A. Somani, and R. Timmons, "Helio-

physics event knowledgebase for the solar dynamics observatory (sdo) and beyond," *Solar Physics*, vol. 275, pp. 67–78, 2012.

[15] C. Verbeeck, V. Delouille, B. Mampaey, and R. De Visscher, "The spoca-suite: Software for extraction, characterization, and tracking of active regions and coronal holes on euv images," *Astronomy & Astrophysics*, vol. 561, p. A29, 2014.

[16] M. A. Schuh, R. A. Angryk, and P. C. Martens, "A large-scale dataset of solar event reports from automated feature recognition modules," *Journal of Space Weather and Space Climate*, vol. 6, p. A22, 2016.

[17] S. Freeland and B. Handy, "Data analysis with the solarsoft system," *Solar Physics*, vol. 182, no. 2, pp. 497–500, 1998.

[18] S. J. Mumford, S. Christe, D. Pérez-Suárez, J. Ireland, A. Y. Shih, A. R. Inglis, S. Liedtke, R. J. Hewett, F. Mayer, K. Hughitt *et al.*, "Sunpy—python for solar physics," *Computational Science & Discovery*, vol. 8, no. 1, p. 014009, 2015.

[19] W. Barnes, M. C. Cheung, M. G. Bobra, P. Boerner, G. Chintzoglou, D. Leonard, S. Mumford, N. Padmanabhan, A. Shih, N. Shirman *et al.*, "aiapy: A python package for analyzing solar euv image data from aia," *Journal of Open Source Software (JOSS)*, vol. 5, no. 55, pp. 2801–2801, 2020.

[20] M. Bobra, J. Mason, C. Holdgraf, tfelipe, tbloch1, R. P. de Lima, Brandon, P. Wright, colinrsmall, R. McGranaghan, A. Rokem, M. Craig, and C. J. D. Baso, "Helioml/helioml: Helioml 0.4.0 (2021-02-08)," Feb. 2021. [Online]. Available: https://doi.org/10.5281/zenodo.4521380

[21] E. Camporeale, S. Wing, and J. Johnson, *Machine learning techniques for space weather*. Elsevier, 2018.

[22] L. Xu, Y. Yan, and X. Huang, "Deep learning in solar astronomy."

[23] E. Camporeale and S. O. C. of ML-Helio, "Ml-helio: An emerging community at the intersection between heliophysics and machine learning," *Journal of Geophysical Research: Space Physics*, vol. 125, no. 2, p. e2019JA027502, 2020.

[24] A. Koul, S. Ganju, M. Kasam, and J. Parr, "Spaceml: Distributed open-source research with citizen scientists for the advancement of space technology for nasa," *arXiv preprint arXiv:2012.10610*, 2020.

[25] E. Camporeale, "The challenge of machine learning in space weather: Nowcasting and forecasting," *Space Weather*, vol. 17, no. 8, pp. 1166–1207, 2019.

[26] N. Lugaz, H. Liu, M. Hapgood, and S. Morley, "Machine-learning research in the space weather journal: Prospects, scope, and limitations," p. e2021SW003000, 2021.

[27] G. Nita, M. Georgoulis, I. Kitiashvili, V. Sadykov, E. Camporeale, A. Kosovichev, H. Wang, V. Oria, J. Wang, R. Angryk *et al.*, "Machine learning in heliophysics and space weather forecasting: a white paper of findings and recommendations," *arXiv preprint arXiv:2006.12224*, 2020.

[28] G. Nita, A. Ahmadzadeh, S. Criscuoli, A. Davey, D. Gary, M. Georgoulis, N. Hurlburt, I. Kitiashvili, D. Kempton, A. Kosovichev *et al.*, "Revisiting the solar research cyberinfrastructure needs: A white paper of findings and recommendations," *arXiv preprint arXiv:2203.09544*, 2022.

[29] M. A. Reiss, S. J. Hofmeister, R. De Visscher, M. Temmer, A. M. Veronig, V. Delouille, B. Mampaey, and H. Ahammer, "Improvements on coronal hole detection in sdo/aia images using supervised classification," *Journal of Space Weather and Space Climate*, vol. 5, p. A23, 2015.

[30] M. A. Schuh, D. Kempton, and R. A. Angryk, "A region-based retrieval system for heliophysics imagery," in *The Thirtieth International Flairs Conference*, 2017.

[31] D. J. Kempton, M. A. Schuh, and R. A. Angryk, "Tracking solar phenomena from the sdo," *The Astrophysical Journal*, vol. 869, no. 1, p. 54, 2018.

[32] J. M. Banda and R. A. Angryk, "Selection of image parameters as the first step towards creating a cbir system for the solar dynamics observatory," in *2010 International Conference on Digital Image Computing: Techniques and Applications*. IEEE, 2010, pp. 528–534.

[33] A. Kucuk, J. M. Banda, and R. A. Angryk, "Solar event classification using deep convolutional neural networks," in *International Conference on Artificial Intelligence and Soft Computing*. Springer, 2017, pp. 118–130.

[34] E. A. Illarionov and A. G. Tlatov, "Segmentation of coronal holes in solar disc images with a convolutional neural network," *Monthly Notices of the Royal Astronomical Society*, vol. 481, no. 4, pp. 5014–5021, 2018.

[35] J. A. Armstrong and L. Fletcher, "Fast solar image classification using deep learning and its importance for automation in solar physics," *Solar Physics*, vol. 294, no. 6, pp. 1–23, 2019.

[36] J.-H. Baek, S. Kim, S. Choi, J. Park, J. Kim, W. Jo, and D. Kim, "Solar event detection using deep-learning-based object detection methods," *Solar Physics*, vol. 296, no. 11, pp. 1–15, 2021.

[37] E. Park, Y.-J. Moon, S. Shin, K. Yi, D. Lim, H. Lee, and G. Shin, "Application of the deep convolutional neural network to the forecast of solar flare occurrence using full-disk solar magnetograms," *The Astrophysical Journal*, vol. 869, no. 2, p. 91, 2018.

[38] N. Nishizuka, Y. Kubo, K. Sugiura, M. Den, and M. Ishii, "Operational solar flare prediction model using deep flare net," *Earth, Planets and Space*, vol. 73, no. 1, pp. 1–12, 2021.

[39] X. Li, Y. Zheng, X. Wang, and L. Wang, "Predicting solar flares using a novel deep convolutional neural network," *The Astrophysical Journal*, vol. 891, no. 1, p. 10, 2020.

[40] R. Jarolim, J. Thalmann, A. Veronig, and T. Podladchikova, "Probing the solar coronal magnetic field with physics-informed neural networks," 2022, preprint from the ML Helio 22 conference.

[41] J. M. Banda and R. A. Angryk, "Unsupervised learning techniques for detection of regions of interest in solar images," in *2015 IEEE International Conference on Data Mining Workshop (ICDMW)*. IEEE, 2015, pp. 582–588.

[42] M. E. Innocenti, J. Amaya, J. Raeder, R. Dupuis, B. Ferdousi, and G. Lapenta, "Unsupervised classification of simulated magnetospheric regions," in *Annales Geophysicae*, vol. 39, no. 5. Copernicus GmbH, 2021, pp. 861–881.

[43] E. Brown, S. Bonasera, B. Benson, J. Pérez-Hernández, G. Acciarini, A. Baydin, C. Bridges, M. Jin, E. Sutton, and M. Jah, "Learning the solar latent space: sigma-variational autoencoders for multiple channel solar imaging," in *Fourth Workshop on Machine Learning and the Physical Sciences (NeurIPS 2021)*, 12 2021.

[44] J. An and S. Cho, "Variational autoencoder based anomaly detection using reconstruction probability," *Special Lecture on IE*, vol. 2, no. 1, pp. 1–18, 2015.

[45] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *arXiv preprint arXiv:1312.6114*, 2013.

[46] C. Baur, B. Wiestler, S. Albarqouni, and N. Navab, "Deep autoencoding models for unsupervised anomaly segmentation in brain mr images," in *International MICCAI Brainlesion Workshop*. Springer, 2018, pp. 161–169.

[47] D. Zimmerer, F. Isensee, J. Petersen, S. Kohl, and K. Maier-Hein, "Unsupervised anomaly localization using variational auto-encoders," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2019, pp. 289–297.

[48] T. Schlegl, P. Seeböck, S. M. Waldstein, U. Schmidt-Erfurth, and G. Langs, "Unsupervised anomaly detection with generative adversarial networks to guide marker discovery," in *International conference on information processing in medical imaging*. Springer, 2017, pp. 146–157.

[49] N. Pawlowski, M. C. Lee, M. Rajchl, S. McDonagh, E. Ferrante, K. Kamnitsas, S. Cooke, S. Stevenson, A. Khetani, T. Newman *et al.*, "Unsupervised lesion detection in brain ct using bayesian convolutional autoencoders," 2018.

[50] W. H. L. Pinaya, P.-D. Tudosiu, R. Gray, G. Rees, P. Nachev, S. Ourselin, and M. J. Cardoso, "Unsupervised brain anomaly detection and segmentation with transformers," *arXiv:2102.11650*, 2021.

[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] B. R. Kiran, D. M. Thomas, and R. Parakkal, "An overview of deep learning based methods for unsupervised and semi-supervised anomaly detection in videos," *Journal of Imaging*, vol. 4, no. 2, p. 36, 2018.

[53] X. Wang, Y. Du, S. Lin, P. Cui, Y. Shen, and Y. Yang, "advae: A self-adversarial variational autoencoder with gaussian anomaly prior knowledge for anomaly detection," *Knowledge-Based Systems*, vol. 190, p. 105187, 2020.

[54] C.-L. Li, K. Sohn, J. Yoon, and T. Pfister, "Cutpaste: Self-supervised learning for anomaly detection and localization," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 9664–9674.

[55] D. H. Ballard, "Modular learning in neural networks." in *AAAI*, vol. 647, 1987, pp. 279–284.

[56] M. Ranzato, Y.-L. Boureau, Y. LeCun *et al.*, "Sparse feature learning for deep belief networks," *Advances in neural information processing systems*, vol. 20, pp. 1185–1192, 2007.

[57] S. Rifai, P. Vincent, X. Muller, X. Glorot, and Y. Bengio, "Contractive auto-encoders: Explicit invariance during feature extraction," in *Icml*, 2011.

[58] P. Vincent, H. Larochelle, I. Lajoie, Y. Bengio, P.-A. Manzagol, and L. Bottou, "Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion." *Journal of machine learning research*, vol. 11, no. 12, 2010.

[59] G. Alain and Y. Bengio, "What regularized auto-encoders learn from the data-generating distribution," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 3563–3593, 2014.

[60] D. P. Kingma and M. Welling, "An introduction to variational autoencoders," *arXiv preprint arXiv:1906.02691*, 2019.

[61] D. J. Rezende, S. Mohamed, and D. Wierstra, "Stochastic backpropagation and approximate inference in deep generative models," in *Proceedings of the 31st International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, E. P. Xing and T. Jebara, Eds., vol. 32, no. 2. Bejing, China: PMLR, 22–24 Jun 2014, pp. 1278–1286. [Online]. Available: https://proceedings.mlr.press/v32/rezende14.html

[62] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[63] D. Pathak, P. Krahenbuhl, J. Donahue, T. Darrell, and A. A. Efros, "Context encoders: Feature learning by inpainting," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 2536–2544.

[64] B. Panos and L. Kleint, "Real-time flare prediction based on distinctions between flaring and non-flaring active region spectra," *The Astrophysical Journal*, vol. 891, no. 1, p. 17, 2020.

[65] C. Huwyler and M. Melchior, "Using multiple instance learning for explainable solar flare prediction," *arXiv preprint arXiv:2203.13896*, 2022.

[66] A. Ahmadzadeh, D. J. Kempton, and R. A. Angryk, "A curated image parameter data set from the solar dynamics observatory mission," *The Astrophysical Journal Supplement Series*, vol. 243, no. 1, p. 18, 2019.

[67] R. Galvez, D. F. Fouhey, M. Jin, A. Szenicer, A. Muñoz-Jaramillo, M. C. Cheung, P. J. Wright, M. G. Bobra, Y. Liu, J. Mason *et al.*, "A machine-learning data set prepared from the nasa solar dynamics observatory mission," *The Astrophysical Journal Supplement Series*, vol. 242, no. 1, p. 7, 2019.

[68] L. van Driel-Gesztelyi and L. M. Green, "Evolution of active regions," *Living Reviews in Solar Physics*, vol. 12, no. 1, pp. 1–98, 2015.

[69] P. R. Young, N. M. Viall, M. S. Kirk, E. I. Mason, and L. P. Chitta, "An analysis of spikes in atmospheric imaging assembly (aia) data," *Solar Physics*, vol. 296, no. 12, pp. 1–21, 2021.

[70] L. F. Dos Santos, S. Bose, V. Salvatelli, B. Neuberg, M. C. Cheung, M. Janvier, M. Jin, Y. Gal, P. Boerner, and A. G. Baydin, "Multichannel autocalibration for the atmospheric imaging assembly using machine learning," *Astronomy & Astrophysics*, vol. 648, p. A53, 2021.

[71] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

[72] C.-H. Teh and R. T. Chin, "On the detection of dominant points on digital curves," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 11, no. 8, pp. 859–872, 1989.

[73] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*. Springer, 2016, pp. 21–37.

[74] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.

[75] T. Denouden, R. Salay, K. Czarnecki, V. Abdelzad, B. Phan, and S. Vernekar, "Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance," *arXiv preprint arXiv:1812.02765*, 2018.

[76] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[77] X. Hou, K. Sun, L. Shen, and G. Qiu, "Improving variational autoencoder with deep feature consistent and generative adversarial training," *Neurocomputing*, vol. 341, pp. 183–194, 2019.

[78] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

Statement of Authorship

By signing this statement, I hereby declare that I completed this Master's thesis on my own and without help of third parties. Further information and sources have been cited and added into references.

Place, date, signature

University of Applied Sciences and Arts Northwestern Switzerland
School of Engineering

Institute for Data Science

Title of work:

Unsupervised Anomaly Detection with Variational Autoencoders in Heliophysics

Thesis type and date:

Master's Thesis, Brugg, August 2022

Supervision:

Prof. Dr. André Csillaghy

Prof. Dr. Jean Hennebert

Student:

Name:	Marius Giger
E-mail:	gigermarius@gmail.com
Legi-Nr.:	15-680-036
Semester:	HS21/FS22