Machine Learning-Based Detection and Characterisation of Magnetopause Crossings in the Lunar-Distance Magnetotail

by

Ian Maes

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly November 13th, 2025.

Student number:	5216931	
Project duration:	February, 2025 – November, 2025	
Thesis committee:	Dr. I. Akay,	TU Delft, supervisor
	Dr. W. van der Wal,	TU Delft
	Dr. A. Menicucci,	TU Delft

Preface

This thesis was written as part of the MSc program in Aerospace Engineering at Delft University of Technology, and marks the completion of my studies. Over the course of this education, I have learned many invaluable lessons, both academically and personally, that have shaped me as an engineer and as a person. I am grateful for the opportunities I have had to explore my interests in space engineering and machine learning.

I would like to express my gratitude to my supervisor, Iklim Akay, for her amazing guidance throughout this project. Her support, insights, and encouragement have been essential, and without her, I would have been stuck many times.

I would also like to thank my friends for listening to my endless rambling whenever I got stuck and for keeping me motivated along the way. Maja deserves a special thanks for always being there for me and supporting me whenever I felt pressure. Finally, I am deeply grateful to my parents and my brother for their continuous support throughout my entire education and for always believing in me, even when I doubted myself.

Ian Maes
Delft, November 2025

Abstract

The Earth's lunar-distance magnetopause is a highly dynamic boundary. Its position plays a key role in shaping weather events within the magnetotail and other space-weather phenomena. Statistical studies of the magnetotail require the identification of magnetopause crossings from spacecraft observations to study its dynamics. However, current detections have some shortcomings. Detecting magnetopause crossings can be time-consuming with rule-based methods and manual inspection, especially at the lunar distance due to its variability. Recent automated classifiers do cover larger datasets, but often miss the dynamic nature of the magnetopause, due to their coarse timing and limited number of detected events. This limits the usability of these datasets for detailed studies of magnetotail dynamics.

This thesis addresses these challenges by developing and evaluating machine learning approaches for detecting magnetopause crossings in the lunar-distance magnetotail using ARTEMIS mission data. First, a gradient-boosted decision tree is used as a baseline, trained to classify magnetosheath and magnetotail samples. Afterwards, a double masked autoencoder (MAE) Transformer is introduced. This model uses a reconstruction-based method to detect changes in plasma regimes and shifts from the magnetotail to the magnetosheath and vice versa. Both models are trained and validated using a labelled dataset, combining ion spectrograms, plasma moments, and a list of known magnetopause crossings.

The results show that the MAE Transformer achieves higher precision with similar recall, and improves timing accuracy compared to the baseline. The MAE transformer is applied to twelve years of ARTEMIS P1 and P2 data, detecting around 3000 magnetopause crossings. The spatial distribution of these crossings matches well with empirical magnetopause models, and a clear correlation to the solar cycle is observed. Outliers in the detected crossings are linked to solar and geomagnetic activity.

Contents

Nomenclature

Acronyms

ACE	Advanced Composition Explorer
aGSM/aSWGSM	Aberration solar wind corrected Geocentric Solar Magnetospheric
ARTEMIS	Acceleration, Reconnection, Turbulence, and Electrodynamics of the Moon's Interaction with the Sun
BBF	Bursty Bulk Flow
BiLSTM	Bidirectional Long Short-Term Memory neural network
CAPS	Cassini Plasma Spectrometer
CME	Coronal Mass Ejection
CNN	Convolutional Neural Network
CPU	Central Processing Unit
CSV	Comma-Separated Values
EFI	Electric Field Instrument
ESA	Electrostatic Analyser instrument
FFN)	Feedforward Neural Network
FGM	Fluxgate Magnetometer
GAN	Generative Adversarial Network
GBDT	Gradient-Boosted Decision Tree
GPU	Graphical Processing Unit
GRU	Gated Recurrent Unit
GSE	Geocentric Solar Ecliptic
GSM	Geocentric Solar Magnetospheric
HDF5	Hierarchical Data Format
HEE	Heliocentric Earth Ecliptic
IMF	Interplanetary Magnetic Field
IMP	Interplanetary Monitoring Platform
ISEE	International Sun–Earth Explorer
LSTM	Long Short-Term Memory neural network
MAE	Masked Autoencoder

MHD	Magnetohydrodynamics
ML	Machine Learning
MLP	Multilayer Perceptron
MMS	Magnetospheric Multiscale Mission
MP	Magnetopause
MVA	Maximum Variance Analysis
PE	Positional Encoding
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROC	Receiver Operating Characteristic (curve)
SITL	Magnetohydrodynamics
SVM	Support Vector Machine
TCN	Temporal Convolutional Network
THEMIS	Time History of Events and Macroscale Interactions during Substorms
TP	True Positives
UT	Universal Time
UV	Ultraviolet

Constants

μ_0	Vacuum permeability
ε_0	Vacuum permittivity
c	Speed of light
k_B	Boltzmann constant

Greek Symbols

β	Plasma beta [-]
δ_y	Solar wind aberration in y direction [deg]
δ_z	Solar wind aberration in z direction [deg]
ρ	Electric charge density [C/m^3]
ρ_{sw}	Solar wind density [$\#$/cm^{-3}]
P_m	Magnetic pressure [nPa]
P_{sw}	Solar wind dynamic pressure [nPa]
$P_{thermal}$	Magnetosphere thermal pressure [nPa]

Roman Symbols

ΔE	Reconstruction error difference [-]
$\mathbf{B}$	Magnetic field [nT]
$\mathbf{E}$	Electric field [μ V/m]
$\mathbf{v}$	Velocity [km/s]
Φ_i	Ion energy flux [eV/(cm^2·s·sr·eV)]
B	Magnetic field magnitude [nT]
b_{emb}	Embedding bias [-]
B_{eq}	Magnetic field strength at magnetic dipole equator [nT]
d_{emb}	Embedded dimension [-]
E_{sheath}	Sheath reconstruction error [-]
E_{tail}	Tail reconstruction error [-]
F	Lorentz force [N]
J	Current density [A/m^2]
L	Loss [-]
M	Magnetic dipole moment [A·m^2]
N	Number of samples [-]
n_{ion}/n_i	Ion density [cm^{-3}]
q	Charge [C]
R_e	Earth radius
r_{mp}	Magnetopause radius [R_E]
T	Temperature [K]
t	Time [s]
T_i	Ion temperature [eV]
V	Ion velocity [km/s]
v_{sw}	Solar wind velocity [km/s]
$V_{th,i}$	Ion thermal velocity [km/s]
V_{th}	Thermal velocity [km/s]
W_{emb}	Embedding weights [-]
z_t	Embedded input [-]
z'_t	Embedded input with positional encoding [-]

List of Figures

List of Tables

1

Introduction

This introduction chapter presents the motivation, research objective, and structure of the thesis.

1.1. Motivation

The Earth's magnetopause, the dynamic boundary between the solar wind and the magnetosphere, plays a key role in regulating how the solar wind interacts with the Earth's magnetic environment. On the dayside, the solar wind compresses the magnetopause, while on the nightside it extends into the long magnetotail. The magnetotail can be heavily shaped by substorms and reconnection events in this tail. Identifying magnetopause crossings is crucial for studying these dynamics. Additionally, the data can help validate magnetopause models, and support space weather research.

Traditionally, boundary crossings have been detected using manual inspection or threshold-based methods. These approaches are effective but time-consuming, and in the case of a dynamic environment like the lunar-distance magnetotail, can be inconsistent and inaccurate. Recent machine-learning methods automate boundary crossing detection, but aren't precise enough to capture the rapid movement of the at large distances. This limits their usefulness for detailed studies of magnetotail dynamics.

1.2. Research Objective

This thesis aims to design and evaluate an automatic machine-learning method for detecting magnetopause crossings in spacecraft data, with emphasis on the lunar-distance magnetotail sampled by ARTEMIS. Additionally, a preliminary analysis of the detected crossings is performed to research their patterns and relation to solar wind and geomagnetic conditions. The central aim of this thesis is formalised as:

> **RO-1:To design and implement an automatic machine learning-based method for detecting magnetopause crossings in spacecraft data.**

To address RO-1, two complementary models are developed and compared:

1. A **gradient-boosted decision tree (GBDT)** baseline using reduced spectrogram statistics and plasma moments, extended to tail conditions, based on historical methods.

2. A **masked-autoencoder (MAE) Transformer**, which exploits spectrogram-moment windows and reconstruction-based change detection to capture temporal context.

Model training and selection are organised around criteria that reflect the needs for statistical studies of space-weather research. Criteria like precision, temporal accuracy, recall, F-scores, and computational efficiency are taken into account. Verification and validation is performed using a labelled dataset of known crossings, with sensitivity studies to assess robustness and pre- and post-processing effects.

1.3. Thesis Structure

The remainder of the thesis is organised as follows. Chapter 2 reviews the physical background of the magnetosphere. Afterwards, it delves into prior detection methods and other relevant magnetotail research. Chapter 3 formalises the problem, research questions, and constraints, and provides a work plan for the research. Chapter 4 delves into machine learning event detection methods and applies a qualitative trade-off to find the best models for boundary crossing detection. Chapter 5 discusses the data preprocessing pipeline as well as model architectures. Chapter 6 outlines the optimisation of the model, and performs an optimisation and sensitivity analysis on pre- and post-processing steps. Chapter 7 reports the results of the model, showing the boundary crossing dataset that is produced from ARTEMIS data between 2013 to 2023. Finally, Chapter 8 concludes the thesis, summarising the key findings and answering the research questions.

2

Literature Review

This chapter provides a comprehensive review of the literature relevant to the study of the Earth's magnetosphere and magnetotail. The chapter is organized into several sections, each addressing a specific aspect of the topic.

Section 2.1 introduces the Earth's magnetosphere and magnetotail, explaining their formation, structure, and interaction with the solar wind and interplanetary magnetic field (IMF). It also covers fundamental concepts such as magnetohydrodynamics, magnetic reconnection, and the near-Earth space environment properties. Section 2.2 discusses traditional methods for identifying the magnetotail boundary, including empirical models, visual boundary crossing determination, and semi-automated approaches. Section 2.3 explores state-of-the-art techniques for boundary crossing identification, focusing on neural network-based and decision tree-based methods. These approaches leverage machine learning to improve the accuracy and efficiency of boundary detection. Section 2.4 reviews recent advancements in magnetotail boundary modelling, highlighting the integration of machine learning with physics-based models to enhance predictive capabilities. Finally, Section 2.5 provides an overview of datasets used for magnetotail boundary crossing studies, detailing their sources, characteristics, and applications in research.

2.1. Introduction to the Magnetosphere and Magnetotail

The Earth's magnetosphere is formed due to the interaction between the planet's internally generated dipolar magnetic field and the solar wind, which consists of a supersonic, magnetised plasma emitted by the Sun. This interaction leads to the creation of complex electric current systems that influence the topology and dynamics of the magnetosphere. Understanding these current systems is essential for comprehending space weather phenomena and their impact on Earth.

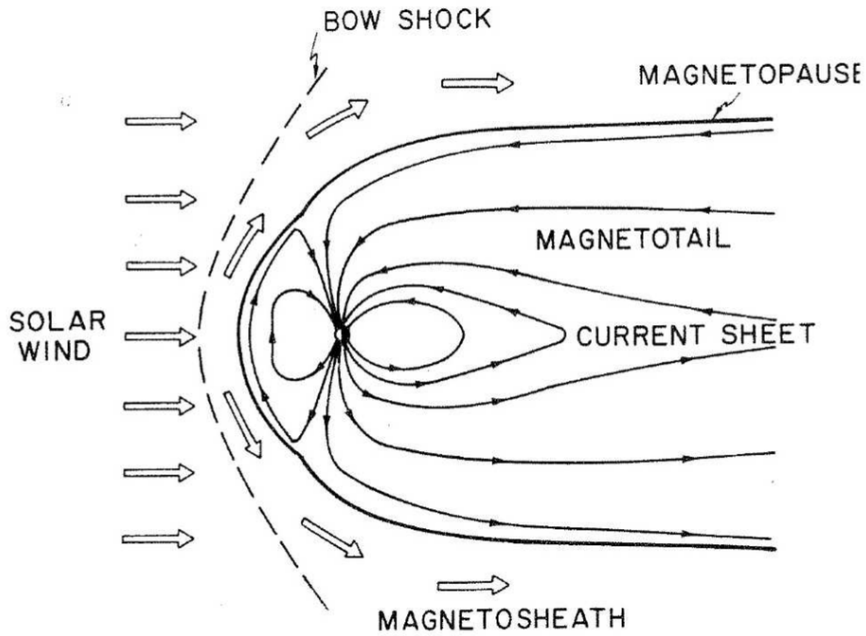


Figure 2.1: Schematic representation of the Earth's magnetosphere. The solar wind compresses the magnetosphere on the dayside and elongates it on the nightside. The boundary separating the magnetosphere from the solar wind is called the magnetopause [65].

Earth's Magnetic Field

Planets cannot have permanent internal magnets because their internal temperatures exceed the Curie point for ferromagnetic materials like iron to keep their magnetism [36]. Instead, magnetic fields are created by electric currents flowing within conductive fluids inside the planet through what's called a magnetic dynamo. This dynamo occurs because the planet's interior fluid moves due to heat-driven convection, combined with the planet's rotation. These fluid motions stretch and twist magnetic field lines, producing strong, looped (toroidal) magnetic fields from an initially simpler (poloidal) magnetic field. The convection cells continuously regenerate the poloidal field, allowing the magnetic field to maintain itself over very long periods [36].

Computer simulations since the 1990s have confirmed this dynamo theory, successfully explaining why Earth's magnetic field persists and occasionally reverses direction [36]. In planets like Jupiter and Saturn, the dynamo is driven in deep layers of conductive metallic hydrogen, whereas Uranus and Neptune generate magnetic fields in shallower ionic layers closer to their surfaces. The dynamo mechanism thus explains the long-lasting, self-sustaining, and periodically changing magnetic fields observed across planets and moons.

The Solar Wind and the Interplanetary Magnetic Field

In the Sun's chromosphere, a region of the solar atmosphere between 2000 and 10000 km above the Sun's surface, small loop-like structures connect sunspots. These sunspots are darker compared to the surrounding regions of the Sun due to their lower temperature ($T \approx 4000\,\mathrm{K}$) compared to the average temperature ($T \approx 5750\,\mathrm{K}$). These spots exhibit higher magnetic field strengths and are typically paired. Each pair consists of two sunspots with opposite polarity, and in a loop-like structure, magnetic field lines connect them. The number of sunspots follows an 11-year cycle, where the polarity, together with the Sun's magnetic field, reverses in every cycle. Charged solar particles follow these loop-like structures in a spiralling motion explained by the Lorentz force, and, in normal circumstances, do not escape the Sun's corona [36].

In some regions, called *coronal holes*, the magnetic field lines originating from the Sun open up into interplanetary space. In this case, charged particles originating from the Sun can easily escape the corona and feed the solar wind. Depending on solar activity, velocities near Earth's orbit vary. During *solar maxima* the solar wind velocity is around $400\,\mathrm{km/s}$, while during *solar minima* the velocity is still on average $400\,\mathrm{km/s}$ near the ecliptic plane but, at higher solar latitudes, can reach around $800\,\mathrm{km/s}$. The solar wind consists of charged particles, mainly protons and electrons in equal proportions, and a small fraction of heavier ions. At Earth's orbital distance, the ion density is typically 6–7 protons cm^{-3}, the temperature is around $10^5\,\mathrm{K}$, and the magnetic field strength is around $5\,\mathrm{nT}$ [36].

While the solar wind radially flows from the Sun's surface into outer space, the magnetic field lines follow a different path. Since the solar wind is a high-temperature neutral plasma, it acts similarly to a perfect conductor (infinite conductivity); magnetic field lines are carried with it due to the frozen-in condition as described by magnetohydrodynamics. Due to the Sun's rotation, these carried magnetic field lines form an Archimedean spiral structure called the Parker spiral, first described by Parker (1958, [54]). This phenomenon is illustrated in Figure 2.2.

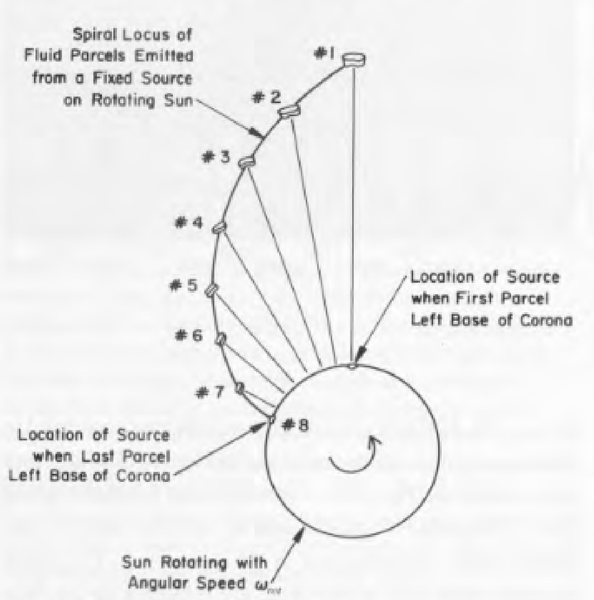


Figure 2.2: The Parker spiral, a spiral-like structure formed by the solar wind magnetic field lines [27].

The solar wind and IMF's properties significantly change due to *space weather*. *Space weather* is a broad term used to describe the changing conditions in the solar–terrestrial environment. The solar wind plays an important role in this, with many solar events influencing the environment. *Coronal holes*, as described earlier, feed the solar wind with high-velocity streams. These high-velocity winds, due to the Sun's rotation, run into slower ambient solar winds and cause shock fronts where particles accelerate locally, and magnetic field lines are compressed, increasing their strength [36]. *Coronal mass ejections* (CMEs) are another source of large disturbances in the solar wind. Sometimes large *prominences* (the loop-like structures connecting sunspots) can erupt due to magnetic reconnection and cause a CME, catapulting a large amount of plasma into space at high velocities (up to $2000\,\mathrm{km/s}$) [36]. These CMEs can cause shocks and compressions in the solar wind, which can create very high-energy particles and disturb planetary magnetospheres. *Coronal mass ejections* are often, but not always, associated with *solar flares*. At the start of *solar flares*, magnetic field lines reconnect, which causes high-energy release and the acceleration of charged particles. Afterwards, reconnection continues and becomes visible through UV and X-ray radiation, forming loops. The effect of these solar events on Earth's environment is often called *space weather storms* [36].

Magnetohydrodynamics

The solar wind's plasma consists of many charged particles. These charged particles interact intimately with magnetic fields, including the IMF and planetary magnetic fields. The interactions of these moving charged particles, electric fields and magnetic fields are described by the Maxwell equations, described here shortly in differential form.

Gauss's law shows how the electric field $\mathbf{E}$ is influenced by charge density ε_0.

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \tag{2.1}$$

Gauss's law for magnetism shows how magnetic fields $\mathbf{B}$ do not originate from point sources, and field lines form closed loops.

$$\nabla \cdot \mathbf{B} = 0 \tag{2.2}$$

Faraday's law of induction shows how electric fields and time-varying magnetic fields are related.

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{2.3}$$

The Ampère-Maxwell law shows how currents $\mathbf{J}$ and time-varying electric fields are related to magnetic fields.

$$\nabla \times \mathbf{B} = \mu_0 \, \mathbf{J} + \mu_0 \, \varepsilon_0 \, \frac{\partial \mathbf{E}}{\partial t} \tag{2.4}$$

Not one of the Maxwell equations, but still important in the context of the solar wind is the Lorentz force. The Lorentz force describes the force acting on a charged particle moving in an electric field $\mathbf{E}$ and a magnetic field $\mathbf{B}$.

$$\mathbf{F} = q \, (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \tag{2.5}$$

Typically, the term containing time-varying electric fields in Equation 2.4 is very small in planetary magnetic fields, and can be ignored [36]. Two important takeaways can be made from Maxwell's equations (and the Lorentz force) regarding the interaction of the solar wind and magnetic fields. Charged particles spiral around magnetic field lines in a spiral motion. Positively charged particles move clockwise and negatively charged particles anticlockwise. Additionally, the currents induce a magnetic field that opposes the original field that caused the generation of the current, weakening the magnetic field strength.

The solar wind acts as a plasma. A plasma is similar to a fluid; in the absence of magnetic fields, its movement can be described by hydrodynamic equations. However, electric and magnetic fields greatly influence the flow of plasma, and the hydrodynamic equations look slightly different because of their effects. The flow of plasma (or any conductive medium) in the presence of magnetic fields can be described by Equation 2.6, which is Ohm's law with an extra term accounting for the Lorentz force.

$$\mathbf{J} = \sigma\left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right) \tag{2.6}$$

In the case of plasma, which is highly conductive due to high ionisation and high temperatures, the term $\frac{\mathbf{J}}{\sigma}$ reduces to close to zero [10]:

$$\mathbf{E} + \mathbf{v} \times \mathbf{B} = 0 \tag{2.7}$$

This equation shows that when moving with the reference frame of the plasma, the electric field is zero. Substituting $\mathbf{E}$ in Faraday's law of induction with Equation 2.7 gives the following equation [57]:

$$\nabla \times (\mathbf{v} \times \mathbf{B}) = \frac{\partial \mathbf{B}}{\partial t} \tag{2.8}$$

A simple way to understand what this equation tells us is to imagine moving with the plasma's moving with the plasma's rest frame. In this case, the velocity $\mathbf{v}$ is zero, meaning that the magnetic field $\mathbf{B}$ is not changing over time in this frame and thus moves with the plasma. This is known as the frozen-in condition, and is a fundamental concept in magnetohydrodynamics. Plasma in ideal MHD cannot move across magnetic field lines; instead, magnetic field lines are dragged with the plasma. This phenomenon can be used to explain a lot of the behaviour of the solar wind and the IMF, and their interactions with planetary magnetic fields.

The Interaction of the Solar Wind and IMF with the Earth's Magnetosphere

Bodies with an internal magnetic field, like the Earth, interact with the solar wind and interplanetary magnetic field. The Earth's magnetic field is an approximate dipole field, with a north and south magnetic pole. This magnetic field is confined by the solar wind to a region called the magnetosphere. This magnetosphere's shape depends heavily on the solar wind's properties, the IMF's properties and the strength of Earth's magnetic field. The IMF's magnetic field lines get draped around the Earth's magnetic field, and the solar wind moves around it. The boundary between the solar wind and the magnetosphere is called the magnetopause. The magnetopause is a thin boundary where the solar wind's dynamic pressure is balanced by the Earth's magnetic field pressure. At the nightside of Earth, this magnetopause stretches out into a long tail-like structure called the magnetotail. The magnetotail has two lobes, the northern and southern lobes with opposite polarities, and a plasma sheet in between. A bow shock forms in front of the magnetosphere where the super-magnetosonic solar wind is suddenly slowed, heated, and compressed. This occurs because the solar wind is moving faster than the fast magnetosonic speed, which is the speed at which information (like pressure changes or magnetic disturbances) can propagate in a plasma. As a result, the solar wind can't "see" the magnetosphere in advance, and a shock forms to mediate this abrupt change [36]. A sketch can be seen in Figure 2.1.

The position of the magnetopause can be estimated by the balance of the solar wind's dynamic pressure and the Earth's magnetosphere pressure. The solar wind's dynamic pressure is given by:

$$P_{sw} = \rho_{sw} v_{sw}^2 \tag{2.9}$$

where ρ_{sw} is the solar wind density and v_{sw} is its velocity. The Earth's magnetospheric pressure is given by:

$$P_m = \frac{B^2}{2\mu_0} + P_{\text{thermal}} \tag{2.10}$$

where B is the magnetic field strength at the magnetopause, and $P_{thermal}$ is the thermal pressure of the magnetosphere. The $P_{thermal}$ is typically very small compared to the magnetic pressure, and can be ignored.

At the equator, using an approximation that the Earth's magnetic field is a dipole, the magnetic field strength at the magnetopause can be estimated by:

$$B_{\text{eq}}(r) = \frac{\mu_0}{4\pi} \cdot \frac{M}{r^3} \tag{2.11}$$

where M is the magnetic dipole moment of the Earth, and r is the distance from the centre of the Earth. The magnetopause radius can be estimated by setting the solar wind pressure equal to the magnetic pressure at the magnetopause. This gives us:

$$r_{\text{mp}} \;=\; \left(\frac{\mu_0 \, M^2}{32 \, \pi^2 \, \rho_{\text{sw}} \, v_{\text{sw}}^2} \right)^{\frac{1}{6}} \tag{2.12}$$

In normal conditions, Equation 2.12 gives a good estimate of the magnetopause radius at the dayside of Earth. The standoff distance of the magnetopause is typically around 6 to 15 Earth radii [36].

Due to the Earth's rotation around the Sun and the non-radial propagation of the solar wind, an aberration of the magnetotail's central axis occurs. This is typically taken into account by using a modified GSE coordinate system called the aGSE or aSWGSE [15]. The GSE coordinate system is a coordinate system where the x-axis points towards the Sun, the z-axis is perpendicular to the ecliptic plane and points north, and the y-axis completes the right-handed coordinate system. To alter this coordinate system to point towards the magnetotail axis instead of the Sun-Earth line, two rotations are applied. First, the rotation about the Z-axis, taking into account the angle at which the solar wind comes when looking at the ecliptical plane 2D view:

$$\delta_Y = \tan^{-1} \left(\frac{V_y + 30}{|V_x|} \right) \tag{2.13}$$

where V_x and V_y are the solar wind's velocity components in the GSE coordinate system. The second rotation is about the Y-axis, taking into account the angle at which the solar wind comes when looking at the xz-plane 2D view:

$$\delta_Z = \tan^{-1} \left(\frac{V_z}{\sqrt{V_x^2 + (V_y + 30)^2}} \right) \tag{2.14}$$

here, 30 is used as it is the average velocity of the Earth around the Sun. The new coordinates can then be given by the following transformation matrix [1]:

$$\begin{bmatrix} x_{\text{aswgse}} \\ y_{\text{aswgse}} \\ z_{\text{aswgse}} \end{bmatrix} = \begin{bmatrix} \cos\delta_Y \cos\delta_Z & -\sin\delta_Y \cos\delta_Z & -\sin\delta_Z \\ \sin\delta_Y & \cos\delta_Y & 0 \\ \cos\delta_Y \sin\delta_Z & -\sin\delta_Y \sin\delta_Z & \cos\delta_Z \end{bmatrix} \begin{bmatrix} x_{\text{gse}} \\ y_{\text{gse}} \\ z_{\text{gse}} \end{bmatrix} \tag{2.15}$$

Additionally, to take into account the rotated dipole of the Earth, the aSWGSE can further be rotated by an angle to the aSWGSM coordinate system [1, 21]. This is further explained in Appendix C.

Magnetic Reconnection

Magnetic reconnection events happen when oppositely oriented magnetic field lines enter a diffusion region from the top and bottom, where the frozen-in condition breaks down. These magnetic field lines are then reconnected to the oppositely oriented magnetic field lines, and leave from the sides. Energy released from this event energises the plasma that entered the diffusion region with the magnetic field lines, and accelerates it out the sides [36].

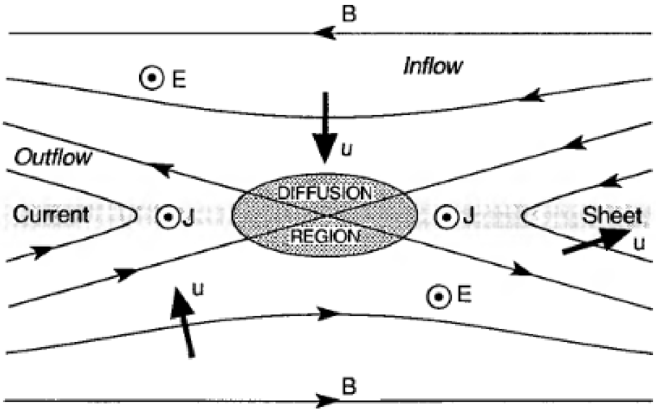


Figure 2.3: Magnetic reconnection illustration at an x-type magnetic neutral line. Magnetic field lines and plasma flow in from the top and bottom, and are reconnected and accelerated out the sides. [26]

In the magnetopause, reconnection can occur when the IMF and the Earth's magnetic field are oppositely oriented. Typically, this happens during a southward-oriented IMF. In this case, the IMF's magnetic field lines are oppositely oriented to the Earth's magnetic field lines at the subsolar point. This causes magnetic reconnection to occur. The reconnected magnetic field lines are dragged towards the tail of the magnetosphere by the solar wind flowing around it. The magnetic field lines enter the tail lobes and are stretched out into the tail. The magnetic field lines can cause another reconnection event to occur in the plasma sheet, a region with higher plasma pressure which interfaces the differently oriented magnetic field lines in the two lobes of the tail. This reconnection event can cause the plasma sheet to be energised and the plasma to be accelerated towards the Earth. Magnetic field lines can be convected back to the dayside again, forming a loop. In a stable loop, this was first described by Dungey (1961, [16]), and is called the Dungey cycle.

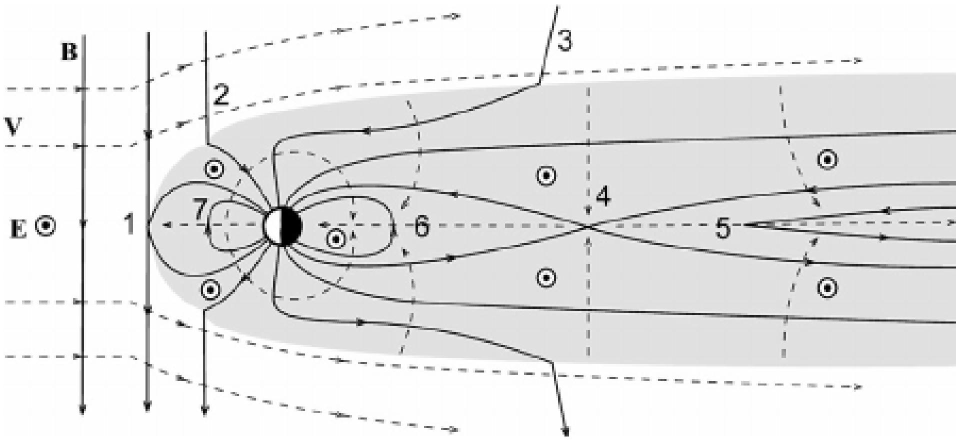


Figure 2.4: The Dungey cycle, a schematic representation of the Earth's magnetosphere and the solar wind. The solar wind compresses the magnetosphere on the dayside and elongates it on the nightside. The magnetic field lines are reconnected at the magnetopause (**1**), and are stretched out into the tail (**2-3**). The magnetic field lines can be reconnected again in the plasma sheet (**4**), and convected back to the dayside forming a loop (**6-7**) [66].

However, in reality, the Dungey cycle is not perfect. The IMF is rarely perfectly southward oriented. The amount of reconnection greatly depends on the IMF's orientation and strength. Moreover, reconnection rates in the tail and at the magnetopause are almost always different, and give rise to magnetospheric substorms due to magnetic field strength build-up. These substorms cause great variability in the magnetosphere's properties and shape [18]. Depending on the IMF's orientation and the solar wind's dynamic pressure, the magnetotail will look very different.

Near-Earth Space Environment Properties

Knowing the Near-Earth Space Environment Properties' differences in the near-Earth space environment properties is important for identifying boundary crossings and studying the magnetotail shape. Parameters like plasma density, temperature, and magnetic field strength help differentiate between regions such as the solar wind, magnetosheath, and magnetosphere. Changes in these parameters indicate boundary transitions, helping with the detection of magnetopause and bow shock crossings. These properties also affect the magnetotail's structure and behaviour, which can help model its shape and response to solar wind and IMF conditions. Table 2.1 below summarises the typical plasma and field properties in these regions, based on [39].

Property	**Solar Wind**	**Magnetosheath**	**Magnetosphere**
Density (cm^{-3})	1–10	10–100	$<$1 (in plasma sheet and lobes), localized peaks
Ion Temperature (eV)	$\sim$1–10 eV	$\sim$100–300 eV, increases near magnetopause	$\sim$1–10 keV, especially in the plasma sheet and outer regions
Electron Temperature (eV)	$\sim$1–10 eV	$\sim$10–100 eV	$\sim$100–1000 eV
Magnetic Field (nT)	$\sim$4 (typical IMF)	Enhanced, turbulent	Strong, structured (Earth's dipole)

Table 2.1: Comparison of plasma and field properties in the solar wind, magnetosheath, and magnetosphere based on [39].

Due to the solar cycle and Earth's orbit, long-term distribution shifts in plasma moments and magnetic field data are likely to occur. Ma et al. (2022, [38]) discusses that over the solar cycle, the solar wind's density, speed, and magnetic-field properties change and shows that those upstream shifts directly alter conditions in the magnetosheath and tail. Because the sheath is formed by the shocked solar wind and the tail is fed and shaped by that same flow, changes in solar-wind pressure and IMF strength over the cycle lead to corresponding shifts in plasma density, temperature, and structure in both regions.

2.2. Traditional Methods for Identifying the Magnetotail Boundary

The first spacecraft observations of Earth's geomagnetotail began in the mid-1960s, notably with NASA's IMP-1 mission launched in 1963, providing initial insights into the extended structure of Earth's magnetic field away from the Sun [47]. Subsequent missions, including Explorer 35 and the Vela satellites, deepened understanding by confirming the magnetotail's stretched configuration, plasma sheet characteristics, and the occurrence of magnetic reconnection [47, 8]. In the 1970s and 1980s, spacecraft like ISEE-1, ISEE-2, and ISEE-3 further advanced knowledge by exploring the tail dynamics, magnetic substorms, and particle acceleration processes [63, 76]. These early decades set the foundation for understanding the geomagnetotail as a dynamic, highly structured region crucial to space weather research [9].

2.2.1. Empirical Models for Magnetopause Determination

Since the first boundary crossings were recorded, many statistical studies have been performed to find empirical models for the magnetotail boundary. Many used similar methods with slightly different fitting functions, while taking the same parameters into account. In this subsection, some of the most remarkable of these studies will be discussed.

The first analytical expression derived for the magnetotail boundary came from Howe (1972, [25]). They did this by being the first to correct for the aberration due to the solar wind, and in this way, came up with the first solar wind-aligned coordinate system. Their fitting method was subjective, using inverse tangent and radical algebraic functions to fit the boundary crossings that came from the Explorer 33 and Explorer 35 spacecrafts. The cross-section of the magnetotail when looking from behind the Earth in the sun direction was assumed to be circular.

Many empirical models developed based on spacecraft measurements assume symmetry about the aberrated x-axis, circular cross-sections, and mostly do not take into account IMF orientation and variable dynamic pressure. In Sibeck, Lopez, and Roelof (1991, [71]), around 1800 boundary crossings were used from many different spacecraft and missions since the early 70s. These boundary crossings were found by visually inspecting plasma and magnetic field data. The study used quadratic polynomials to describe the square of the radius of the magnetopause in the yz-plane. To do this for differing IMF and solar wind conditions, they used a binning strategy, where five to six bins were made for both pressure ranges and IMF B_z ranges. No combined formula of both effects was calculated, but the effect of the solar wind dynamic pressure was taken into account for the IMF equation by changing the subsolar magnetopause location. In Roelof (1993, [61]), similar methods are used, but the radius is now defined by a bivariate function of the solar wind dynamic pressure and IMF B_z values for different bins. In this case, they constructed bins in a control space along different axes (rotated versions of pressure and IMF B_z), creating overlapping bins. Each bin encompassed magnetopause crossings corresponding to similar solar wind conditions.

Kuznetsov and Suvorova (1998, [34]) used parabolic models to try and fix discrepancies in Roelof (1993, [61]) for B_z values, giving a false dependency on positive values. Petrinec (1996, [55]) used an empirical model derived for the flaring angle of the magnetotail to determine a relation for the radius of the tail. They also addressed limitations of elliptical and parabolic empirical functions at large distances by using inverse trigonometric functions to better describe the nightside magnetopause.

Shue et al. (1998, [70]) developed an improved empirical model for magnetopause location that builds on Shue et al. (1997, [69]). Their model uses a flexible functional form to represent both open and closed magnetospheric configurations depending on IMF conditions. By incorporating the IMF B_y component to adjust tail flaring, the model better accounts for variations in magnetotail geometry. A key improvement was including non-linear relationships between solar wind dynamic pressure, IMF B_z, and the subsolar standoff distance, which captures saturation effects under extreme solar wind conditions where earlier linear models failed. This results in greater accuracy in reproducing observed magnetopause crossings, even during intense solar wind events when the boundary is compressed.

In many of the studies named, the contribution of the IMF on the magnetopause shape is considered by taking into account a B_z factor, and a circular cross-section is assumed. However, B_y can significantly flatten and elongate the tail cross-section, making boundary detection difficult. In Akay, Kaymaz, and Sibeck (2019, [1]), this study examines the cross-section in the yz-plane of the aSWGSM 2.15 coordinate system was studied at a distance of -60 R_E. The study used magnetopause boundary crossings from the ARTEMIS P1 and P2 satellites from August 2011 until August 2013. They used clock angles, split up into 4 regions, to describe the orientation of the IMF, and used this binning strategy to find the cross-section dependence of the magnetotail. They found that east-west-oriented IMF fits oblate ellipses more, and north-south IMF indicates a prolate ellipse. Both increasing positive and negative B_z were found to decrease the tail size, with the rate of decrease slightly faster in the negative case.

Nguyen et al. (2022, [49]) focuses on refining the empirical understanding of the magnetopause's location and shape under varying solar wind and IMF conditions. It uses a large, multi-mission dataset of 17,230 magnetopause crossings from 17 spacecraft (including THEMIS, MMS, Cluster, ARTEMIS, etc.), most of which were automatically detected using the gradient boosting region classifier described in Nguyen et al. (2022, [48]), later discussed in subsection 2.3.2. The study excludes cusp-region crossings to focus on more stable boundary geometries. The authors analyse how key factors like solar wind dynamic and magnetic pressure, IMF B_z and B_y components, and the Earth's dipole tilt influence the stand-off distance and magnetopause flaring. They confirm a power-law relation between stand-off distance and total pressure, consistent with earlier models, and show that meridional flaring increases under southward IMF, while equatorial flaring dominates during northward IMF. They also provide updated empirical formulas to describe this behaviour, accounting for clock angle and dipole tilt dependencies.

2.2.2. Visual Boundary Crossing Determination

Traditional boundary crossings were mostly found by inspecting the data visually. An example can be seen in Figure 2.5. This figure shows a 12-day segment of ARTEMIS-P1's journey through the magnetotail from November 4 to 16, 2011, highlighting the different space regions it encountered at lunar distance. The top panel uses coloured hatched lines to indicate these regions, while the other panels present magnetic field components (B_x, B_{tot}), plasma density, temperature, velocity (V_x, V_{tot}), and ion measurements.

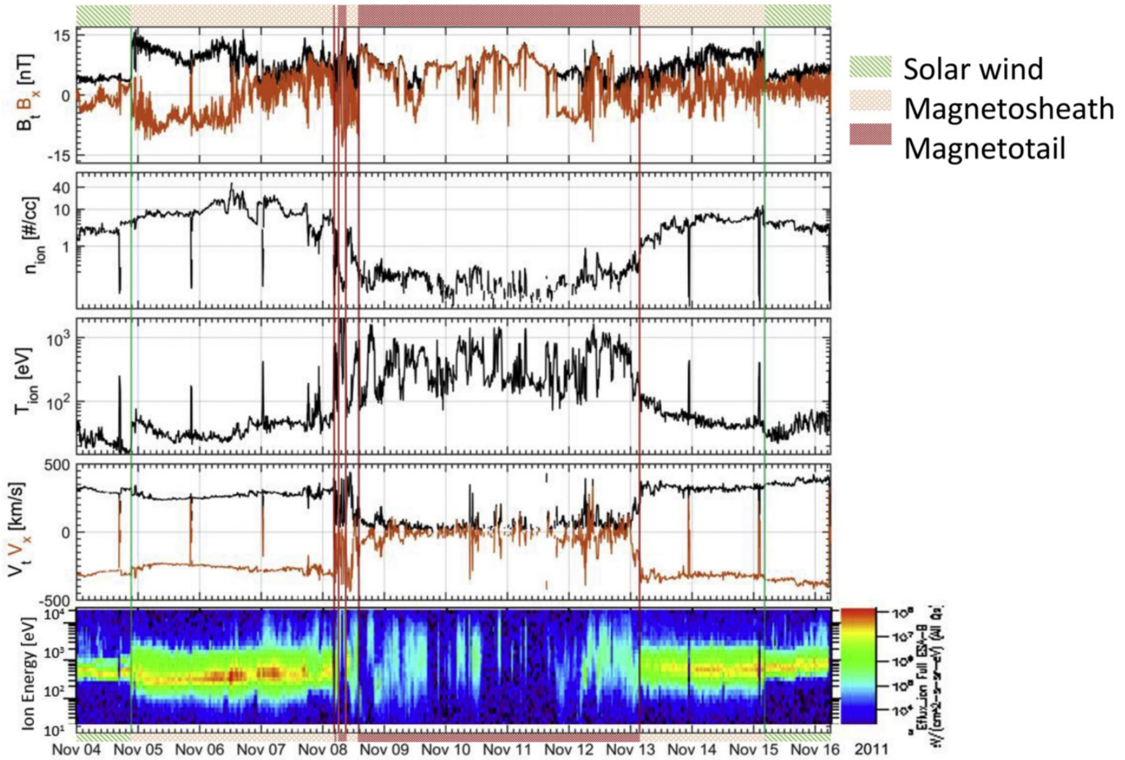


Figure 2.5: An example of magnetotail traversal in Nov. 2011. Vertical lines indicate bow shock (green) and magnetopause crossings (red) when ARTEMIS enters and exits the tail from left to right [1].

During a magnetopause crossing as seen in Figure 2.5, the ARTEMIS spacecraft typically detects clear changes in plasma properties. As it moves inward from the magnetosheath into the magnetotail, high-energy particle flux disappears, plasma temperature rises sharply, and both density and total velocity drop. The opposite occurs during outbound crossings. These transitions, along with differences in particle energy levels, help identify the magnetopause boundary.

2.2.3. (Semi-) Automated Methods for Boundary Crossing Detection

Staples et al. (2020, [73]) uses a semi-automated, rule-based algorithm based on sudden changes in magnetic field (B_z) and ion density, which had to occur in a short time window. Initial manual classification of 18 clear THEMIS-E crossings was used as training data. Candidate crossings were identified using empirically set thresholds and then manually verified to remove false positives. This resulted in a validated dataset of 20,000 dayside crossings.

In a similar way for the magnetotail, in Raymer (2020, [60]) the authors developed a new set of plasma-based criteria. The method is rule-based and uses thresholds on parameters like plasma velocity, density, and magnetic field variability. Specifically, a crossing is identified when there is a sharp change in X-direction plasma flow (greater than 100 km/s difference), combined with a velocity shift from near-stagnant inside the magnetosphere to faster anti-sunward flow in the magnetosheath (less than -100 km/s). Density must also jump, and magnetic field variability must increase across the boundary.

2.3. State of the Art: Boundary Crossing Identification

Boundary crossing detection is a crucial task in space physics, particularly for understanding the dynamics of the magnetosphere and its interaction with the solar wind. Traditional methods, while historically effective, often rely on subjective interpretation of data. Recent advancements in machine learning and data analysis techniques have opened new ways of automating this process.

2.3.1. Neural Network Based Methods for Boundary Crossing Detection

Argall et al. (2020, [2]) presents a machine learning model developed to automate the detection of magnetopause (MP) crossings in data from NASA's MMS mission. These crossings are key to studying magnetic reconnection, but due to data limits, only 4% of high-resolution data can be downlinked. Traditionally, a Scientist-in-the-Loop (SITL) manually selects these events. The goal of the paper is to reduce that manual burden through automation. The machine learning model is a Bidirectional Long Short-Term Memory (BiLSTM) neural network, designed to process time series data and identify MP crossings using the same low-resolution data available to SITLs. It takes in 123 features, including magnetic field vectors, electric fields, plasma density, velocity, temperature and anisotropy, and more derived metrics. Data is normalised and broken into sequences of 250 time steps (at 4.5 s resolution) for training. The model was trained on SITL-labelled data and implemented into the mission's operational system. It achieved a high performance, identifying 78% of MP crossings labelled by scientists.

Cheng, Achilleos, and Smith (2022, [13]) explores how to automate the detection of Saturn's bow shock and magnetopause using Cassini spacecraft data. They compare two methods. The threshold method calculates magnetic and plasma parameters within sliding windows and applies optimised fixed criteria to detect crossings. While simple and fast, it struggles with variable field features. The deep learning method uses a ResNet18 convolutional neural network (CNN) trained on images of magnetic field and electron spectrogram data. An LSTM autoencoder first filters out uninteresting intervals. The CNN is trained on 2004-2011 data and tested on 2012, with gradient-weighted class activation mapping (Grad-CAM) used to interpret its predictions. Results show that the CNN, especially with CAPS spectrograms, significantly outperforms the threshold method. It achieved F1 scores of 92.1% (BS) and 84.7% (MP), and identified many new crossings, updating existing catalogues. Its compact size and fast inference time make it well-suited for onboard use in future missions.

Azib et al. (2023, [4]) introduces a regression-based method for detecting events in multivariate time series, addressing issues like imbalanced datasets, rare events, and events defined over time intervals. The method transforms the time series into overlapping windows (partitions), each converted into a fixed-size input vector. Instead of classifying each time step as “event” or “non-event,” a regression model predicts how closely each window overlaps with a ground-truth event, using a Jaccard similarity-based score. These scores are treated as a continuous signal, where peaks indicate events. A feedforward neural network (FFN) with a single hidden layer is trained to learn this mapping. The authors demonstrate mathematically (via universal approximation theorems) that this simple FFN can approximate the overlap function with arbitrary precision under mild continuity assumptions. After training, the output of the FFN is smoothed using a Gaussian kernel, and peak detection is applied to extract final event predictions. Parameters like the Gaussian width and peak threshold are optimised to maximize the F1 score. Tested on credit card fraud and bow shock crossing datasets (from Mars Express), the method outperforms traditional classifiers and deep models like ResNet18, achieving an F1 score of 95% for bow shock detection, while using far fewer parameters, making it suitable for resource-limited environments.

Julka et al. (2023, [30]) presents a deep learning approach to detect bow shock and magnetopause crossings in Mercury’s magnetosphere using data from NASA’s MESSENGER mission. The task is framed as a multi-class, time-step-level classification problem with a future prediction component. Input data windows are passed through a neural network to predict the class of both the current and upcoming time steps. They test several model architectures (MLP, CNN, RNN, CRNN, CANN), and find that a Convolutional Recurrent Neural Network (CRNN) performs best, achieving a macro F1 score of 82%, with recall of 78% for bow shock and 86% for magnetopause crossings.

In Julka, Ishmukhametov, and Granitzer (2024, [29]), using MESSENGER magnetic field data, the authors aim to detect concept drifts, changes in orbit data distribution caused by variable space weather, and adaptively sample the most informative orbits for training a CRNN (Convolutional Recurrent Neural Network) classifier to detect bow shock and magnetopause crossings. The approach uses a GAN-based drift detector to identify orbit groups with similar magnetic features, then applies entropy-based sampling within each group to select training data. This ensures diverse coverage while minimising the number of labelled orbits needed. Compared to baseline methods using entropy alone or random sampling, this method achieved higher F1 scores, especially for bow shock crossings, and reached saturation in performance with only 250-300 orbits (under 10% of the dataset). A custom smoothing filter was also introduced to improve the temporal consistency of crossing predictions.

2.3.2. Decision Tree Based Methods for Boundary Crossing Detection

Nguyen et al. (2022, [48]) presents a fast and efficient method to classify space environment regions, the magnetosphere, magnetosheath, and solar wind from in-situ spacecraft data. Similar to previously discussed neural network-based methods, the purpose is to automatically detect magnetopause and bow shock crossings across several missions.

The authors propose a Gradient Boosting Classifier to perform the classification. This supervised learning algorithm was chosen for its speed and efficiency over more complex neural networks, and for its strong performance on imbalanced classification problems. Gradient boosting operates by sequentially building an ensemble of decision trees, each correcting the residuals of the previous one. The final prediction aggregates the outputs of these trees, producing well-calibrated class probabilities. Input features include basic plasma moments and magnetic field components: velocity (V_x, V_y, V_z), magnetic field (B_x, B_y, B_z), ion density, and temperature. The model was initially trained on data from the THEMIS mission, which included manually labelled examples across various near-Earth regions.

The model was evaluated using standard metrics like True Positive Rate, False Positive Rate, and Area Under the ROC Curve. Results showed AUCs above 0.99 for all three classes on THEMIS data, with TPRs above 94% and very low FPRs. The algorithm was then tested on data from other missions (Double Star, MMS, Cluster, ARTEMIS). For equatorial missions like MMS and Double Star, the model generalised well without retraining. For polar orbit missions like Cluster and more distant ones like ARTEMIS, some retraining or feature adjustment (e.g. including spacecraft coordinates or a fourth class for the lunar wake) was required to maintain high accuracy. The method consistently outperformed traditional threshold-based classification and matched or surpassed neural network-based models—all while training significantly faster.

The model was used to generate extensive magnetopause and bow shock crossing catalogues from 11 spacecraft over 83 mission-years. Magnetopause crossings were defined as a 1-hour interval where as many data points were classified as being in the magnetosphere and the magnetosheath. In total, over 15,000 magnetopause and 17,000 bow shock crossings were identified. For ARTEMIS, 702 crossings were detected. These catalogues, validated through statistical and spatial consistency checks, offer a foundation for large-scale analyses of near-Earth boundaries.

2.4. State of the Art: Magnetotail Boundary Modelling

The magnetotail is a complex and dynamic region of the Earth's magnetosphere, influenced by various factors including solar wind conditions, IMF orientation, and internal magnetospheric processes. Traditional methods for identifying the magnetotail boundary have relied on empirical models and manual inspection of spacecraft data. However, recent advancements in machine learning and data analysis techniques have opened new avenues for automating this process, improving accuracy and efficiency.

2.4.1. Neural Network Based Studies for Magnetotail Boundary Modelling

Hou and Hsieh (2025, [24]) introduces Reg-PINN, a method that combines machine learning with empirical physics-based models to predict the Earth's magnetopause location. Traditional empirical models (like Shue et al. (1998, [70])) generalise well but lack precision, while machine learning models can be highly accurate but often fail to respect physical constraints. Reg-PINN addresses this by embedding the empirical model directly into the neural network's loss function, balancing data fitting with physics-inspired regularisation.

The model was trained on over 34,000 magnetopause crossings using solar wind parameters such as dynamic pressure (p_{dyn}) and IMF B_z. It was compared against the Shue model, an overfitted empirical model, and a standard neural network. Across all tests, including low-data scenarios and extreme solar wind conditions, Reg-PINN consistently reduced RMSE by 30%, outperforming other methods in both precision and generalisation. The approach shows strong potential for space weather modelling and future scientific applications that require physics-informed learning.

2.5. Magnetosphere Datasets

Datasets are essential for magnetospheric research. Statistics-based methods and machine learning based methods require large amounts of data to train and validate their models. In this section, we will discuss some important datasets often used in magnetospheric research.

2.5.1. Spacecraft Measurement Datasets

The NASA Coordinated Data Analysis Web [44] provides a comprehensive database of spacecraft measurements, including data from missions like ACE, Cluster, Geotail, THEMIS, and others. This database is often used for analysing the magnetosphere and its boundaries. For the THEMIS and ARTEMIS missions, the database includes data from the five THEMIS spacecraft (THA, THB, THC, THD, THE), of which THB and THC are now ARTEMIS P1 and P2. The database contains a wide range of measurements, including magnetic field data, plasma density, temperature, and velocity. The [46] provides a database for the same spacecraft, but is focused on spacecraft position and trajectory data. These databases are often used together for magnetospheric research.

2.5.2. Magnetopause Crossing Datasets

The magnetopause crossing datasets in the NASA Magnetopause Database [45] include observations from numerous spacecraft (e.g., IMP, ISEE, Geotail) spanning 1963-1998. Crossings were identified based on combined measurements of magnetic field changes and particle data, typically a sharp shift in magnetic field orientation/magnitude and variations in energetic and suprathermal particle populations. The database also mentions additional databases, including the ones produced by Raymer (2020, [60]) and Staples et al. (2020, [73]).

Nguyen et al. (2022, [48]) also provides a magnetopause crossing dataset from 11 spacecraft (THEMIS, MMS, ARTEMIS, Cluster, Double Star) over 83 mission-years. This dataset includes over 15,000 magnetopause crossings and 17,000 bow shock crossings, identified using a Gradient Boosting Classifier trained on manually labelled examples. For MMS, THEMIS, Cluster and Double Star, this dataset is also publicly available.

2.6. State of the Art: Event Detection Using Machine Learning

This section aims to provide an overview of the state-of-the-art machine learning techniques used for event detection in time-series data, with a focus on methods used in physics-based applications.

2.6.1. Sequence Modelling/Forecasting

Sequence modelling and forecasting involve predicting future values or sequences based on historical data. This is particularly useful in time-series data, where observations are recorded over time, such as sensor data and weather data. The goal is to capture temporal dependencies and patterns in the data to make accurate predictions.

Two popular neural network architectures for sequence modelling are Recurrent Neural Networks (RNNs) and Temporal Convolutional Networks (TCNs):

Recurrent Neural Networks

RNNs are designed to process sequential data by maintaining a hidden state that captures information about previous time steps. At each time step, the network updates its hidden state based on the current input and the previous hidden state. This allows RNNs to model temporal dependencies effectively. However, traditional RNNs suffer from issues like vanishing gradients, and many timesteps make the model learn very slowly, which makes it difficult to learn long-term dependencies. Variants like Long Short-Term Memory (LSTM) networks and Gated Recurrent Units (GRUs) address this limitation by introducing gating mechanisms to control the flow of information [84].

They have been applied to physics-informed modelling of dynamical systems, including fluid flow and climate processes, as well as forecasting chaotic time series like the Lorenz attractor with physics-constrained architectures [59, 86].

Temporal Convolutional Networks

TCNs are an alternative to RNNs for sequence modelling. They use 1D convolutional layers to process sequential data, enabling parallel computation and capturing temporal dependencies through dilated convolutions. Dilated convolutions allow the network to have a larger receptive field without increasing the number of parameters, making TCNs efficient for learning long-term dependencies. Additionally, TCNs avoid the vanishing gradient problem and are often more stable during training compared to RNNs [6].

TCNs have been used to model input-output mappings in quantum many-body systems driven by optical pulses, and for constitutive modelling of stress-strain relationships by learning history dependence in materials [78, 67].

These forecasting models can be used to predict future states of the data they see, and deviations from these predictions can be used to detect anomalies or change points in the data.

2.6.2. Reconstruction-Based Change Point Detection

Change point detection involves identifying points in a time series where the statistical properties of the data change significantly. These changes could indicate shifts in the underlying process generating the data, such as system failures, regime changes, or anomalies.

Autoencoder-Based Change Detection

Autoencoders are neural networks designed to learn a compressed representation of input data and then reconstruct it. For change point detection, autoencoders are trained on normal data and encode data into a latent space. Due to this smaller representation, the model is forced to learn common patterns in the data. The model is trained to decode this data as close as possible to the original. When a change point occurs, the reconstruction error increases significantly, signalling a deviation from the learned normal behaviour.

In physics or engineering contexts, this approach has been applied to detect abrupt shifts in sensor measurements (e.g. in structural health monitoring), or to flag transitions in climate or geophysical time series by monitoring deviations in learned latent reconstructions [19].

MAE Transformer

The Masked Autoencoder Transformer is based on the Transformer architecture, which uses self-attention mechanisms to capture dependencies in sequential data. The MAE Transformer is designed to handle image-like data, and can be adapted for time-series data [23]. It resembles an autoencoder, with an encoder and decoder, but instead of using a latent space, it splits up the input sequence into patches, masks a portion of them, and trains the model to reconstruct the missing patches. Change points are detected by monitoring the reconstruction error over time, with significant increases indicating potential change points.

CNN Autoencoder

A Convolutional Neural Network (CNN) autoencoder uses convolutional layers to capture spatial or temporal patterns in the data. For time-series data, the CNN autoencoder processes sequences to learn local dependencies. The encoder compresses the input sequence into a latent representation, while the decoder reconstructs the sequence. Change points are detected by monitoring the reconstruction error over time. CNN autoencoders are particularly effective for data with strong local patterns.

These models are effective in domains where local correlations dominate. They have been applied to detect abrupt shifts in vibration signals of machinery, or to spot transitions in radar time-series echo profiles [64, 75].

Anomaly Transformer

The Anomaly Transformer is a specialised model for detecting anomalies and change points in time-series data [81]. It combines the self-attention mechanism with an anomaly-aware module to focus on irregular patterns. The model computes an anomaly score for each time step by comparing the attention distribution of the current time step with its neighbours. A significant deviation in the anomaly score indicates a potential change point.

2.6.3. Classification

Classification follows the premise of supervised learning. A model is trained to label data points based on their features as one of several classes. The model learns from a labelled dataset, where each data point is associated with a specific class label. In binary classification, there are two classes, and each data point is classified as one of these two. In magnetopause boundary crossing detection, the task would be to classify a given window of data as either a crossing or not crossing the magnetopause boundary.

Pointwise Classification for Region Shift Detection

Pointwise classification treats each time series data point independently. Meaning each classification is made independently of the surrounding time series data points. Pointwise classification can be used to find magnetopause crossings by observing a shift in the classification of data points. By finding the moment this shift occurs, the timing of the crossing can be determined. Pointwise classification is a simple and effective approach for many classification tasks, but it may not capture temporal dependencies effectively. Such a method is used by Nguyen et al. (2022, [48]) in their Gradient Boosting Classifier for magnetopause and bow shock crossing detection. Many machine learning models can be used for pointwise classification, including decision trees, support vector machines (SVMs), and neural networks.

Window Classification for Event Detection

Window classification involves classifying a sequence of data points as a whole, rather than individually. This approach captures temporal dependencies and patterns within the window, allowing for more accurate classification. In the context of magnetopause boundary crossing detection, window classification can be used to identify periods of time where the data indicates a crossing event. By analysing the entire window, the model can learn to recognise complex patterns that may indicate a crossing. Again, many machine learning models can be used for window classification, including RNNs, TCNs, and CNNs.

Windowed classification could also be used to classify different regions of the magnetosphere, such as the magnetosheath, magnetosphere, but would severely limit the temporal resolution of the crossings detection.

3

Research Proposal

This chapter presents the research proposal for the thesis project. It outlines the knowledge gap in the field of magnetopause crossing detection. The research objectives are outlined using a main research question and subquestions, which will be answered during the project. The research plan is presented in a Gantt chart, which outlines the phases of the project and their respective timelines.

3.1. Knowledge Gap

The determination of magnetopause crossings is a topic that many statistical studies on Earth's magnetotail cover. These large statistical studies often use datasets consisting of thousands of magnetopause crossings. Nguyen et al. (2022, [48]) used a decision tree to classify magnetospheric regions. Using this data, they could pinpoint magnetopause crossings to approximate timeslots of one hour. While this technique works very fast and detects many magnetopause crossings, the timeslot range is rather large. This can induce problems in further statistical studies that might require exact timings to match solar wind data with, and accuracy in positions is also lost. Additionally, due to the dynamic nature of the magnetopause, multiple crossings can occur during this time period, meaning multiple crossings could be missed by this detection method. This is especially true for the lunar distance magnetotail region (where the ARTEMIS mission resides), where the magnetopause is highly dynamic and can change rapidly due to solar wind conditions.

Raymer (2020, [60]) and Staples et al. (2020, [73]) used a different approach. They used a semi-automatic rule-based method to detect magnetopause crossings. This method is useful for finding exact timings, but human intervention is needed to determine whether detections are correct. This can cost a lot of time, as stated in Staples et al. (2020, [73]), daily inspections were performed by scientists to check event recordings. Argall et al. (2020, [2]) used a Bidirectional Long Short-Term Memory (BiLSTM) neural network, which was specific to the MMS missions, and not generalised, and achieved a limited F1 score of 0.41, and detected 78 % of SITL-identified MP crossings.

As discussed in subsection 2.3.1, most machine learning based boundary detection models were designed for interplanetary missions, and not for Earth-based magnetospheric missions. With limited magnetopause boundary crossing datasets at lunar distance, an increased accuracy, temporal performance and an increased amount of datapoints could greatly improve the quality of data used in statistical studies of the magnetosphere shape. Additionally, the use of machine learning models can help to automate the detection process, reducing the need for human intervention and allowing for real-time analysis of in-situ spacecraft data.

3.2. Stakeholder Analysis

A stakeholder analysis helps identify which parties will use or benefit from the research, and more specifically, what they need. It ensures the research goals match the needs of active users like space weather scientists, and possibly space mission designers and operators. This process helps set clear requirements and constraints for the machine learning model, making sure it works well for its intended purpose. Stakeholders and their main use cases and needs are identified and discussed here.

Active stakeholders:

- **Space Weather Scientists:** Can use the model efficiently and accurately, finding magnetopause crossings in their data. Need a fluent interface to use the model, and benefit from a clear database of magnetopause crossings already identified during the research phase.
- **Spacecraft Engineers and Operators:** Can use the model to find magnetopause crossings in their data, and relay this information to scientists. In the long term, this can help to improve the quality of data collected by magnetospheric missions due to better models of its workings. Similarly, they need a fluent interface as well.
- **Principal Investigator:** Actively develops the system. Can provide further support after the project is finished.
- **Thesis Supervisor:** Influences the project's direction by suggesting improvements, identifying potential challenges, and helping to refine the research questions and methodology.

Passive stakeholders:

- **Space Weather Scientists:** Can use the results of the model for detailed studies into the magnetotail. More specifically, use the results for statistical and deep learning research of its tail.
- **Space Mission Designers:** Can use the results of the model, and results of subsequent studies, to more accurately predict the effects of the magnetosphere on spacecraft's orbits and electronics (energetic particles can damage these).

Space weather scientists were identified as both active and passive stakeholders. They can use the model for their own research, but can also solely use the results of the model produced by other scientists or entities.

Space weather scientists were identified to be the most important stakeholders to consider in this project. Mainly because space weather scientists are the primary users of the system, and their research heavily relies on accurate and efficient detection of magnetopause crossings. Their needs and requirements will directly influence the design, functionality, and success of the system, making them, overall, the most critical stakeholders. The insights provided by the system can significantly enhance their ability to study the magnetosphere's dynamics, conduct statistical analyses, and improve models of solar wind-magnetosphere interactions.

The needs of the stakeholders are included as requirements in Appendix B. The analysis of whether these needs are met is done in subsection 7.1.3.

3.3. Research Objectives

Based upon the knowledge gap discussed earlier, and the stakeholder needs, the following research objective is defined:

RO-1 — To design and implement an automatic machine learning-based method for detecting magnetopause crossings in spacecraft data, and analyse its results.

To achieve this objective, the following research questions are formulated:

MRQ — How can machine learning models be used for detecting magnetopause crossings in Earth's magnetosphere from in-situ spacecraft data?

- **SRQ-1** — What preprocessing steps are required to convert raw or region-labelled spacecraft data into a form suitable for boundary crossing detection using machine learning models?
- **SRQ-2** — What performance metrics best capture the performance of magnetopause crossing detection?
- **SRQ-3** — How can a magnetopause crossing ML model be evaluated and verified with little verification data?
- **SRQ-4** — How do different machine learning models, and more traditional methods, compare for magnetopause crossing detection?
- **SRQ-5** — What patterns do machine learning models detect in the data to identify magnetopause crossings?
- **SRQ-6** — What patterns can be found in the detected crossings using the machine learning model, and how does it compare to the solar cycle?
- **SRQ-7** — What are outliers (if any) in the detected crossings caused by?

By answering these research questions, this research aims to contribute to the field of magnetospheric physics by providing a more accurate and efficient method for detecting magnetopause crossings, which can provide a better understanding of the magnetosphere's dynamics and its interactions with solar wind and IMF.

The research questions are answered throughout the thesis, and are explicitly answered in chapter 8.

3.3.1. Hypotheses

The following hypotheses are formulated based on the research objectives and questions:

H1 — Machine learning models have the potential to significantly improve the accuracy and temporal precision of magnetopause crossing detection compared to traditional methods. This can likely be achieved by using models strong in data pattern recognition, like CNNs and transformer models. The THEMIS missions have collected large amounts of data, which benefits machine learning applications strongly. Several interplanetary missions have benefited from machine learning models which detected bow shock crossings and magnetopause crossings [30, 4, 13]. These models have been shown to be able to detect crossings, even with more limited datasets. While not all machine learning models and methods have been explored yet, it is clear that machine learning models have found their place in the field of space weather. With a large dataset available, it is expected that machine learning models can be trained to detect magnetopause crossings with high accuracy and temporal precision, and to help automate this process.

3.4. Model Constraints

It is important to define the model constraints, as they clearly define what the research project will focus on and what it will not focus on. The unprocessed input data features are discussed. Defining what preprocessing is necessary is done during the design of the machine learning model to find optimal performance. Additionally, output data is defined, and a description of the design constraints is provided. Here, a description is given of the model constraints; a requirement formatted version can be found in Appendix B.

3.4.1. Input Data Constraints

The model can take input in-situ spacecraft data, which includes the following features over multiple-hour/day time periods:

- **Ion and Electron Density:** Ion density measurements.
- **Ion and Electron Flow Velocity:** ion velocity magnitude.
- **Ion and Electron Temperature:** Plasma temperature measurements for both ions and electrons.
- **Ion Energy Spectrograms:** Energy spectrograms for ions and electrons, providing energy flux as a function of energy and time.
- **Magnetic Field Data:** Magnetic field vector measurements in appropriate coordinate systems (e.g., GSE, GSM).

Important to note is that in the eventual models developed, not all features are used. However, the data pipeline and model creation tools do allow for this. A smaller subset of features with the Ion Energy Spectrogram ended up being the most performant. Reasoning for these features is further explored in subsection 6.2.2.

3.4.2. Output Data Constraints

The model will output the following:

- **Timestamps of Magnetopause Crossings:** A list of timestamps (if any) indicating when magnetopause crossings were detected.
- **Crossings Markers:** If any other relevant information is necessary, like whether the crossing was possibly another event with similar markers.

Output data will be formatted to allow for easy integration with downstream analysis, possibly in a database.

3.4.3. Design Constraints

It is important to note that the model will not be specifically developed as a real-time operational model on a satellite. Rather, it is designed with the intended purpose of serving as a tool for analysing public science data. The model will be designed to process earth-based data, specifically that of the **ARTEMIS** missions. Research will focus on the lunar distance magnetotail region and find magnetopause crossings there.

The model will also be designed to work on a single computer, with modern GPU and CPU capabilities. This constrains the model to a certain size of parameters and its complexity. Runtime benchmarking numbers will be provided for the computer the model was initially trained.

3.5. Risk Analysis

The main goal of this risk analysis is to make sure the research project runs smoothly. This means avoiding and managing technical difficulties that can impact the development and results, and non-technical difficulties which could impact the timeline of the project and its progress. Through a risk identification process, risks are identified and described, and their impact and likelihood are assessed. Afterwards, mitigation and avoidance strategies are developed to both reduce the likelihood of the risk occurring, and to reduce the impact of the risk if it does occur.

3.5.1. Risk Identification

The following risks were identified during the project planning phase. The risks are divided into two categories: technical and non-technical. Technical risks are related to the machine learning model, while non-technical risks are related to the project timeline and other external factors.

Table 3.1: Identified Risks and Descriptions

Non-Technical Risks	
Risk ID	**Risk Description**
NT-1	Limited access to computational resources, such as GPUs, which could slow down model training and evaluation.
NT-2	Integrating feedback consumes more time than expected, pushing the timeline along.
NT-3	External factors such as illness or unforeseen personal circumstances significantly affecting the project timeline.
NT-4	Tasks are more challenging and time-consuming than initially expected, pushing the timeline along.
Technical Risks	
Risk ID	**Risk Description**
T-1	Extracting and preprocessing publicly available data is slower due to unforeseen data extraction hurdles, slowing progress.
T-2	Labelled dataset creation takes longer than expected, leading to delays in model training and evaluation (Work Package 3).
T-3	Labelled dataset contains errors or inconsistencies, affecting model performance (Work Package 3).
T-4	Labelled dataset is too small or unbalanced, which could lead to overfitting or underfitting of the model (Work Package 3).
T-5	Risk of overfitting or underfitting during model training, leading to poor generalisation to unseen data (Work Package 4).
T-6	Inadequate performance of the selected model compared to existing methods, failing to meet the research objectives (Work Package 5).
T-7	Errors in preprocessing pipelines or feature extraction leading to incorrect input data for the model.
T-8	Errors in the model code cause unexpected behaviour or crashes during training or evaluation, prolonging the project timeline.

3.5.2. Risk Assesment

The risk assessment allows for a visual way of identifying which risks need to most thought-out mitigation and avoidance strategies. Each risk is evaluated based on its impact and likelihood. Impact is rated on a scale from 1 to 3, where 1 represents low impact, 2 represents medium impact, and 3 represents high impact. Similarly, likelihood is rated from 1 to 3, with 1 indicating low likelihood, 2 indicating medium likelihood, and 3 indicating high likelihood. The overall risk score is determined by multiplying the impact and likelihood ratings. Risks with higher scores are given greater priority for mitigation. The risks are ranked in the case of no mitigation strategies. The risk assessment matrix is shown in Table 3.2.

Table 3.2: Risk Assessment Matrix

Likelihood / Impact	**Low (1)**	**Medium (2)**	**High (3)**
High (3)	T-6, NT-1	T-4	T-1, NT-4
Medium (2)		NT-3	T-2, T-3, T-5
Low (1)	T-7		NT-2, T-8

Some clarifications are necessary for some of the choices in this table. On first inspection, **T-4** and **T-5** seem to have the same impact. The difference is that **T-4** does not focus on the model's inadequacies, but rather data's inadequacies, the implications on the preprocessing methods used, and the needed changes in the dataset. **T-5** focuses on the model and/or hyperparameters being the problem, which often occurs, and can be more 'easily' fixed by model/hyperparameter tuning. Additionally, **T-6** assumes that **T-4** and **T-5** are not the root cause for performance issues, but the model type itself is inadequate to perform the task, meaning another type of model might be necessary.

3.5.3. Risk Mitigation and Avoidance

Risk mitigation and avoidance strategies are developed to reduce the likelihood of the risk occurring and to reduce the impact of the risk if it does occur. The following strategies are proposed for each identified risk:

Table 3.3: Risk Mitigation and Avoidance Strategies

Risk ID	Mitigation and Avoidance Strategy
Non-Technical Risks	
NT-1	Use cloud-based computational resources (e.g., Kaggle, Google Cloud) as a backup in case local resources are unavailable.
NT-2 & NT-3	Schedule regular feedback sessions to make integration workload lower. Additionally, allocate buffer time in the project timeline for feedback incorporation.
NT-4	Break down tasks into smaller, manageable subtasks. Regularly review progress and adjust the timeline as needed.
Technical Risks	
T-1	Develop automated scripts for data extraction and preprocessing. Test data pipelines early to identify potential bottlenecks.
T-2	Use available datasets as a basis for classification.
T-3	Perform thorough quality checks on the labeled dataset.
T-4	Look for similar use of machine learning models as a basis for dataset size. Ensure balanced class distribution during preprocessing. Use weighting methods to avoid performance issues due to an unbalanced dataset.
T-5	Implement cross-validation and regularization techniques to prevent overfitting. Use a diverse validation set to ensure generalisation.
T-6	Evaluate multiple models and architectures. If performance is inadequate, document why this is the case, and possibly look for solutions by using hybrid models.
T-7	Validate preprocessing pipelines with small test datasets before full-scale implementation. Document and review all preprocessing steps.
T-8	Use version control (git) for code and maintain detailed logs of changes. Conduct regular code reviews and stick to clear coding practices.

These mitigation strategies should minimise the impact of the risk on the project's success and timeline.

3.6. Research Plan

To define the research plan, the work will be divided into six work packages, each with specific objectives and tasks. The Gantt chart in Figure 3.1 provides a visual representation of the timeline for each work package, including the key milestones set by the thesis standards.

- **Work Package 1: Literature Review**
  This work package focuses on reviewing existing work related to magnetopause crossing detection and the use of machine learning in space physics. It includes a theoretical description of the magnetosphere to give the reader a fundamental understanding. The goal is to understand current limitations (SRQ-1) and identify opportunities for improvements.
- **Work Package 2: Model Selection**
  In this work package, candidate machine learning models suitable for time series event detection are explored. Particular attention is paid to their ability to handle dynamic input

data and their suitability for magnetospheric applications (SRQ-1, SRQ-4). Models like CNNs, LSTMs, and transformers are considered.

- **Work Package 3: Dataset Preprocessing**
  Publicly available spacecraft data is collected (e.g., from THEMIS or ARTEMIS), cleaned, and transformed into a format suitable for training machine learning models. Region labelling, feature extraction, and event labelling pipelines are built and evaluated (SRQ-2, SRQ-3).
- **Work Package 4: Model Development**
  The selected machine learning models from Work Package 2 are implemented and trained using the preprocessed dataset from Work Package 3. Necessary architecture adjustments to be able to process the multidimensional time series data are made.
- **Work Package 5: Model Evaluation**
  This work package focuses on evaluating the performance of the trained models using various metrics, including precision, recall, F1 score, and temporal accuracy. This evaluation is performed against the labelled crossing dataset. The goal is to compare the models' effectiveness in detecting magnetopause crossings with existing methods (SRQ-4).
- **Work Package 6: Analysis of Detected Crossings**
  This work package focuses on analysing the detected magnetopause crossings from the trained model. The analysis includes identifying patterns in the crossings, their relation to the solar cycle, solar wind conditions, and investigating outliers (SRQ-5, SRQ-6, SRQ-7).
- **Work Package 7: Writing**
  The Writing Work Package focuses on documenting the research process, findings, and analysis in a structured manner, aligned with academic standards.
- **Work Package 8: Finalisation**
  This work package focuses on integrating feedback on the draft, preparing for the final presentation, and the final report is handed in, and the thesis is presented.

These work packages should give an overview of the main tasks that will be performed during the project.

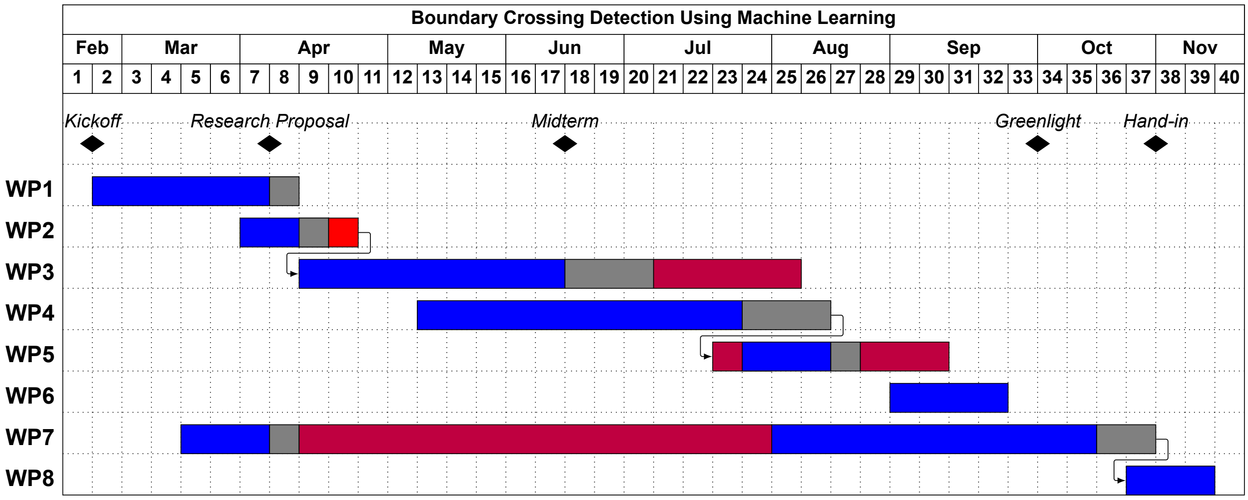


Figure 3.1: Gantt chart of the research plan. Showing each work package with its start and end weeks, including the main milestones. Grey is the runout period, where work can be finished, or extra work can be done. Purple indicates changes made to the research plan, and red shows unallocated time that was spent on certain tasks.

The length of the tasks was chosen based on personal experience in certain domains and their difficulty level. The Gantt chart is used as a guideline during the project and to check progress occasionally. It will be dynamically updated depending on the progress of certain tasks.

4

Model Selection

In this section, we will discuss the model selection process in detail. This process tries to find the optimal machine learning models for detecting crossings in a qualitative way, to limit the number of models to be tested. First off, the model selection strategy will be discussed. Afterwards, a design option tree will show all the possible design options for the magnetopause boundary crossing model. Afterwards, the criteria with which the models will be evaluated are defined. Finally, the most fitting model(s) will be selected based on the evaluation criteria.

4.1. Model Selection Strategy

The model selection strategy will be based on the following steps:

1. **Initial Model Selection**: Based on a literature review, a set of models will be selected that might be relevant and suitable for magnetopause crossing detection. These will be shown in the design option tree. They will be based on whether they can theoretically meet the requirements defined in Appendix B.
2. **Criteria Definition**: The evaluation criteria with which the models will be evaluated will be defined. This will include both qualitative and quantitative criteria.
3. **Model Evaluation**: The remaining models will be evaluated based on the defined criteria. The best model(s) will be selected based on the evaluation.

Following these steps should ensure a thorough and systematic approach to model selection. The selected models will be used for the next steps in the design process and research.

4.2. Initial Model Selection

Zamanzadeh Darban et al. (2024, [83]) offers a survey on anomaly detection in time series data. Many different types are discussed based on two main methods: forecasting and reconstruction. Forecasting methods predict future values based on past data, while reconstruction methods learn a representation of the data and identify anomalies based on deviations from this representation. The paper contains a comprehensive list of many models used with these 2 methods. Based on this list, the most popular models' main architectures will be considered for the design option tree and further model evaluation. Additionally, the list shows whether point and/or subsequence anomalies can be detected by certain models. Boundary crossings are most similar to subsequence anomalies, since crossings rarely occur from one timestep to another. Therefore, subsequence detection is a hard requirement, and architectures need examples of subsequence detection. For forecasting methods, the following models will be considered: Recurrent Neural Networks (RNNs), Temporal Convolutional Networks (TCNs). For reconstruction methods, the following models will be considered: Neural Network-Based Autoencoders and Transformers (here named MAE transformer). GANs are not considered, as they have mainly been used in research, and offer unique challenges in training due to dependence on game theory [83].

Pointwise classification methods are not typically used for anomaly detection, but can be used for magnetopause crossing detection. This is due to crossings moving across regions with different field and plasma properties. Pointwise classification methods can be used to classify these regions, and crossings can be detected by looking at the classification of the points in a certain window. Traditional methods for pointwise classification (Random Forest, SVM, Decision Tree, etc.) have been previously used for this purpose and will be considered, as well as neural network-based methods, which work very similarly, but might offer better performance [4, 48] .

Window-based classification methods are also not typically used for anomaly detection, but have been effective in the past [13]. Here, they used a 2D CNN-based model to classify windows of time series data to determine whether crossings occurred. Vision transformers are growing in popularity for image classification practices (these 2 architectures are the most popular by far) and sometimes outperform CNNs, and are therefore also considered [31].

4.3. Criteria Definition

Model evaluation will be based on four criteria: performance, flexibility, scalability, and complexity/efficiency. Reasoning is explained below:

- **(Historical) Performance/Reliability**: The model should be able to accurately detect magnetopause crossings. Based on a literature review, the performance of models will be evaluated based on applications similar to magnetopause boundary crossing detection. This essentially entails reliability; how sure we can be that the model is up to the task, based on previous research. (Requirement SR-4).
- **Flexibility**: The models will be evaluated based on their flexibility. This means the ability to adapt to different features and parameters, the ability to add or remove features, and the ability to fit within the pipeline. Additionally, the datasets are not very clean; noise, inconsistencies and gaps are present. Flexibility accounts for the model's ability to handle these issues (after preprocessing). Additionally, due to the size, dynamic behaviour and noisiness of the data, as well as the dataset being made of manually labelled crossings, giving hard labels would be challenging. (Requirements SR-1, SR-6, SR-7, SR-11).
- **Scalability**: Scalability is important, as the datasets vary in size. Models will be evaluated based on their ability to handle large datasets and their ability to scale with the size of the dataset. (Requirements SR-1, SR-10)
- **Complexity/Efficiency**: The complexity and efficiency criterion contains the size of the model, the number of parameters, and the time it takes to train and evaluate the model. This criterion is important, as larger models, which are less efficient, take longer to optimise and evaluate. A complex model is also less interpretable, which makes optimisation harder. (Requirement SR-10)

Additionally, these criteria will be weighted based on their importance. Performance is the most obvious and important criterion. The model's main goal is to accurately detect magnetopause crossings. Therefore, performance will be weighted the heaviest with a weight of **4**. Flexibility is important, as the data used for magnetopause crossing detection is quite varied, but not extremely so. However, the time series data is very inconsistent. Preprocessing should take care of most of these issues, but inconsistencies will always partially propagate, and the model should be able to handle that. Flexibility is given a weight of **2** for these reasons. Scalability is less important, as datasets vary in size; this can also be handled by preprocessing. Large datasets can be split into smaller datasets for the model to handle. Scalability is given a weight of **1**. Complexity and efficiency are very important as they heavily influence one's ability to optimise the model. This also influences interpretability, which is important for researchers to understand the magnetopause crossing detection process. Complexity and efficiency are given a weight of **3**.

4.4. Model Evaluation

Every model will be evaluated based on the four criteria defined in section 4.3. The models will be scored for each criterion on a scale from 1 to 5. It should be noted that the scores given are based on the specific context, goal, and timeline of this research.

Model	**Performance/ Reliability**	**Flexibility**	**Scalability**	**Complexity/ Efficiency**	**Total**
Sequence Modelling/Forecasting					
1. Recurrent Neural Network (LSTM, GRU)	3	3	2	2	27
2. Temporal Convolutional Network (TCN)	4	3	4	3	35
3. Time Series Transformer	5	5	4	3	45
Reconstruction					
4. Neural Network Based Autoencoder	4	5	5	4	44
5. MAE Transformer	5	5	4	4	47
Pointwise Classification					
6. Traditional Methods (Random Forest, SVM, ...)	2	4	2	5	32
7. Gradient Boosted Methods (MLP, 1D CNN, ...)	3	4	5	5	39
Window Based Classification					
8. Vision Transformer Model	4	3	4	3	35
9. 2D CNN	3	2	4	3	28
Weights	**4**	**3**	**1**	**2**	**10**

Table 4.1: Evaluation of models based on weighted performance, flexibility, scalability, and complexity/efficiency.

To explain the scores given in the evaluation table, similar models with respect to the criteria are grouped. The following subsection briefly discusses the reasoning behind the scoring.

4.4.1. Performance

As discussed in Figure 2.1, the near-Earth space environment depends heavily on solar wind conditions, which vary over the solar cycle. This means distribution shifts in plasma moments and magnetic field data are likely to occur. Therefore, models with resistance to distribution shifts are favoured.

Unsupervised methods are often more resilient to distribution shifts, as they learn the underlying structure of the data, rather than specific patterns tied to a certain distribution [68]. Therefore, reconstruction and forecasting-based methods are favoured and given a higher score.

Transformer models are highly performant in time series machine learning tasks. They capture long-term dependencies and can learn complex patterns [85]. Therefore, the Vision Transformer Model gets a score of **4**. The Time Series Transformer (3) and the MAE Transformer

(4) are given a score of **5** due to their unsupervised resistance against distribution shifts.

The 2D CNN model (9) is given a score of **3**. These models are good at capturing localised patterns in spectrograms, but would likely struggle to relate patterns occurring farther apart, and find cross-variable patterns, since they are optimised for 2-dimensional data. Additionally, since this is a classification-based model, they are less resistant to distribution shifts.

Recurrent neural networks (1) typically perform well in small data sets without long-term dependencies and highly multivariate data, but struggle more when these are present. Since the forecasting will be performed on multivariate long-time series, accuracy might be limited. A TCN (2) can model long-term dependencies, but at the cost of needing much deeper networks. Since both models are forecasting-based and are more resistant against distribution shifts, they are given a score of **3** and **4**, respectively.

Gradient Boosted Methods (7) have historically been used for magnetopause crossing detection at lunar-distance [48]. They showed promise in that area, but for lunar distance detection, they needed strong preprocessing and feature engineering, and the addition of spacecraft position to perform well. Since they are also not very resistant to distribution shifts, they are given a score of **3**. Traditional methods (6) typically perform worse than Gradient Boosted Methods, and are given a score of **2**.

4.4.2. Flexibility

Due to the dynamic nature of the magnetopause and the noisy data. Labelled datasets are hard to create and often limited in size. Due to this limitation, flexibility scores are heavily influenced by their performance when only soft labels in clearly defined regions are available. While hard labels are possible to create, it is a very intensive and time-consuming task, where human errors are likely to occur. Therefore, models that can work with self-supervised learning are favoured. This means that reconstruction and forecasting-based methods get higher scores.

Reconstruction and forecasting-based methods (1, 2, 3, 4, 5) have an advantage over other models as they can be trained self-supervised. This means that no extensive dataset of hard labels is needed, and dynamic regions do not have to be labelled, but can be excluded. This earns the reconstruction-based models an additional point.

Transformer-based models (3, 5, 8) are highly flexible due to their ability to handle variable-length sequences, multivariate inputs, and adapt to a wide range of time series tasks [82, 37]. This flexibility earns them a score of **5**. Except for the vision transformer, which might need adjacent models for non-2-dimensional features, and isn't as resistant against distribution changes, earning it a score of **3**.

Neural network-based autoencoders (4) and pointwise classification models (7) also offer considerable flexibility; variables of different dimensionality can easily be integrated into the model, giving a score of **5** and **4** as well [83]. Where NN-based autoencoders are given a score of **5** due to their self-supervised learning capabilities.

Recurrent neural networks (1) and temporal convolutional networks (2) are moderately flexible, capable of modelling sequential data but sometimes limited by architectural changes for being able to model higher-dimensional data, so they receive a score of **3** [7].

2D CNN models (9) are less flexible in this context, as they are best suited for structured 2D data like spectrograms and may not generalise well to other data types. Features of other dimensionalities will need to be implemented for processing in adjacent structures, limiting flexibility, resulting in a score of **2**. Traditional methods (6) are flexible. It is easy to add features. However, due to their limited capability in modelling distribution shifts, they do not get a score of **5**, but **4**.

4.4.3. Scalability

Transformer-based models (the MAE Transformer (5), Time Series Transformer (3), and Vision Transformer (8)) are highly scalable due to their parallelizable architectures, which allow them to efficiently process large datasets and long sequences. However, their memory requirements can grow quickly with sequence length due to their quadratic memory scaling, so they are given a score of **4**.

2D CNN models (9) are also highly scalable, as convolutions can be efficiently parallelised and are well-suited for large-scale data due to their sparser nature, but when deeper networks are necessary due to long time series sequences, memory use can increase, earning it a score of **4**.

Neural network-based autoencoders (4) have not been specified to use convolutional layers in this context, but can certainly do so, and benefit similarly from the parallelization and sparsity of convolutions. Additionally, in the context of usage, sequence length will always be limited to a certain amount of time steps, and only the number of features could drastically increase. The argument is the same for Gradient Boosted Models (7) and Temporal Convolutional Networks (2). Therefore, they are given a score of **5**.

Recurrent Neural Networks (1) are less scalable due to their sequential nature, which makes parallelization difficult and slows down training on long sequences. Thus, they are given a score of **2** [7]. Traditional Methods (Random Forest, SVM, etc.) (6) are typically less scalable, especially with high-dimensional or very large datasets, as their complexity needs to increase drastically before being able to perform. Therefore, they are given a score of **2**.

4.4.4. Complexity/Efficiency

Transformer-based models (including the MAE transformer (5), time series transformer (3), and vision transformer (8)) are generally complex due to their large number of parameters, especially for long sequences. However, their attention maps offer somewhat moderate interpretability. Their self-attention mechanisms scale quadratically with sequence length, making them less efficient for long time series sequence length [35]. As a result, these models are given a score of **3** for complexity/efficiency. The MAE transformer (5) is given a score of **4**, as it will use lower sequence lengths.

2D CNN models (9), while powerful, also tend to be complex when applied to time series data, especially if deep architectures are required to capture temporal dependencies. However, their convolutional operations are highly parallelizable and can be efficient on modern hardware [20]. If convolutional networks become deep, their interpretability decreases heavily. Which is likely to occur in its usage for boundary crossings, due to the many features and possibly long sequences. Therefore, they receive a score of **3**. Temporal convolutional networks (2) are also relatively efficient due to their convolutional structure, which allows for parallelization, and for similar reasons will be given a score of **3**.

Neural network-based autoencoders (4) and especially Gradient Boosted Methods (7) are generally less complex than transformer-based models. They have fewer parameters and are more efficient to train and evaluate, especially for moderate-sized datasets. Additionally, due to the pointwise classification, and the short sequence length to encode (compared to forecasting methods), complexity is not high [81]. Therefore, Neural network-based autoencoders are assigned a score of **4**, and Gradient Boosted Methods is given a score of **5**.

Recurrent neural networks (1) are less efficient due to their sequential processing, which limits parallelization and increases training time, especially for long sequences. Their complexity is moderate, so they receive a score of **2**. Traditional methods (Random Forest, SVM, etc.) (6) are typically the most efficient and least complex for small to moderate datasets, as they require less computational power and are easier to interpret. However, their efficiency decreases with very large or high-dimensional datasets. Overall, they are given a score of **5**.

4.5. Conclusion

Based on the evaluation in Table 4.1, the following models are selected for further research and development:

- **MAE Transformer (5)**: This model is selected due to its high performance, flexibility, and scalability. It is specifically designed for reconstruction of data, making it a strong candidate for magnetopause crossing detection. Additionally, it is both resistant to distribution shifts and can be trained in a self-supervised manner.
- **Gradient Boosted Decision Tree (GBDT) (6)**: This pointwise classification method is selected for both its simplicity and historical use in magnetopause crossing detection. It will serve as a strong baseline for comparison with the more complex MAE Transformer based model.

This gives, based on a combination of qualitative analysis with a tradeoff analysis, a set of models that can be used for magnetopause crossing detection. This answers **SRQ-1**. In the upcoming sections, the overall methods of both models are discussed.

5

Methods

This chapter outlines the methodology used to detect magnetotail boundary crossings from spacecraft measurements. It describes the data sources, preprocessing steps, and feature preparation procedures required to construct a consistent and analysable dataset. The chapter then details the implementation of two main approaches: a baseline gradient boosted decision tree classifier and a reconstruction-based method using Masked Autoencoder (MAE) Transformers.

5.1. Data Collection

Data collection was performed using a combination of automated scripts and manual data entry. The NASA Coordinated Data Analysis Web [44] database provides access to THEMIS satellite data, which can be easily extracted using the Python-based Space Physics Environment Data Analysis Software (PySPEDAS) framework. The data was downloaded in the form of Computable Document Format (CDF) files. The user can specify the time range for which data needs to be downloaded. All data in the specified time range is put into a single dictionary for easy data handling.

The data extracted contains three dimensional magnetic field and electric field measurements (taken from the Electric Field Instrument (EFI) [11], and the Fluxgate Magnetometer (FGM) [3]) ion and electron spectrograms, three dimension ion and electron velocities, ion and electron densities and ion and electron temperatures (taken from the Electrostatic Analyzer instrument (ESA) [42]).

The gradient boosted decision tree classifier is built using the standard implementation from scikit-learn 1.4.2. The MAE Transformer is built using the PyTorch 2.2.1 framework with CUDA support. The computer uses a Ryzen 9 AI HX 370 CPU, 32GB RAM and a NVIDIA RTX 4060 GPU with 8GB of VRAM.

5.2. Initial Data Preparation

The initial data preparation involved converting the CDF files into a format suitable for analysis, as well as some initial cleaning steps.

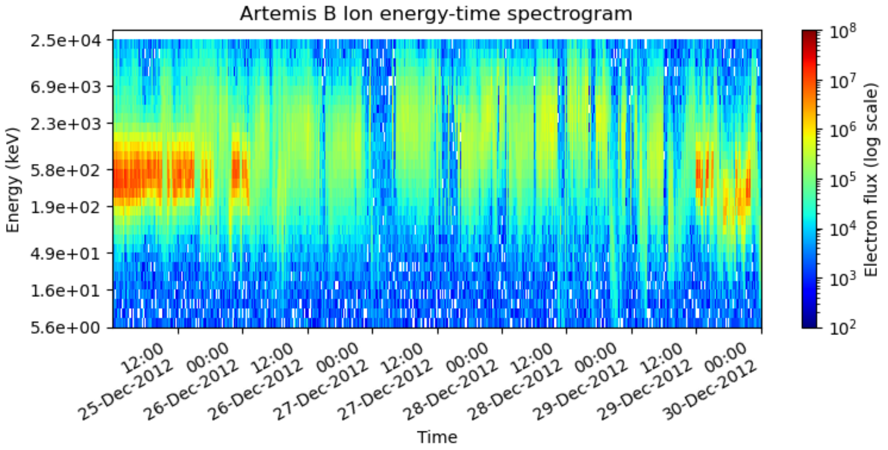


Figure 5.1: Example of ARTEMIS B ion energy flux data. The plot shows the energy flux in bins, where the y-axis denotes the energy bins in keV and the x-axis shows the time. The colour scale represents the flux in particles per square centimetre per second per keV.

For machine learning models to effectively process spacecraft measurement data, consistent temporal sampling is necessary. For example, electron and ion energy flux are typically measured at different time intervals, and finding cross-variable correlations is impossible if samples are provided of different time ranges. Additionally, a consistent time delta allows the models to find consistent patterns across timesteps. This is part of what steps are needed for preprocessing raw data and thus partially answers **SRQ-2**.

This initial pipeline performs several key steps to standardise and clean the dataset. First, it removes any variables that contain only NaN (missing) values to eliminate empty datasets. This was often electric field data. Next, a universal time resolution is set by the user. For the rest of this research, a time resolution of 10 minutes is used. This is quite a low resolution, but necessary to use spectrogram data, which is often sampled at these lower time resolutions.

Finally, it converts all variables, also non-spectrogram variables, to this uniform time delta through resampling and linear interpolation, ensuring temporal consistency across the entire dataset. Depending on what is possible, variable resampling or linear interpolation is used. If more than one timestep of the variable data falls within the window of the newly decided timesteps, the average is taken. If this is not the case, linear interpolation is used between the two closest timesteps. In the case that 30 minutes (three times the set time resolution) of data is not available, the dataset is split, meaning that a separate section of data is created on both sides of this gap. Additionally, trimming the data is also possible to exclude parts of the start and end of the spectrogram, which might not be needed. An example of a preprocessed ion spectrogram can be seen in Figure 5.1.

After the initial data preparation, the dataset is stored as HDF5 files, which allows for easy and fast retrieval of large datasets, especially in PyTorch, which requires fast access to the data during training. The HDF5 format is particularly suitable for handling large datasets, as it supports hierarchical data organisation and efficient compression. Each time range of the dataset is stored as a different session in the HDF5 file, allowing for easy access to specific time periods. This split is often done per mangetotail pass (around a week of data), but can be split up if gaps in data are too large, as mentioned earlier. This ensures that the model does not have to deal with large gaps in data during training or inference, which could lead to poor performance.

5.3. Training Data Preparation

After the initial data preparation, the dataset is accessible using a dataloader class, which allows for easy access to the data during training. The dataloader class is designed to handle the HDF5 files and provides methods for loading the data in batches, as well as applying any necessary transformations. Transformations depend on the model being trained, but some common transformations include normalisation, window averaging, and logarithmic scaling.

5.3.1. Data labelling

For supervised learning tasks, the data needs to be labelled. In this research, the labelling is done by using a list of known magnetotail boundary crossings, taken from [1]. This list contains the timestamps of known crossings; these crossings are used to split the data into magnetotail and magnetosheath regions. All data points beyond the first entry and last exit are labelled as magnetosheath, with a buffer of several hours. Magnetotail regions are labelled as all data points between the last entry and first exit of the magnetotail region, also with of buffer of 30 minutes. After this automatic sorting, all sections are manually checked to ensure the labelling is correct, and if data seems unfit, it is removed from the dataset.

5.3.2. Feature Preparation

When loading the data using the dataloader class, some derived parameters are calculated and stored in the class. These include the plasma beta, the magnetic field magnitude, the absolute value of the magnetic field x component, and the x component of the ion velocity, all in the GSM coordinate system. These derived parameters are often used in magnetospheric physics, as they provide important information about the plasma environment. All calculations of these parameters are fairly straightforward, and the plasma beta is calculated as follows [5]:

$$\beta = \frac{2\mu_0 n_{ion} k_B T}{B^2} \tag{5.1}$$

Where μ_0 is the permeability of free space, n_{ion} is the ion density, k_B is the Boltzmann constant, T is the ion temperature, and B is the magnetic field magnitude. In current models, only the absolute value of the magnetic field x component is used.

When data is loaded in the data loader class and is ready for training, the data is prepared within the machine learning model class, with the same subsequent process for both the MAE and GBDT models. Data preparation first stores all data of each variable, across all sections, is added to a large variable-specific array.

All one-dimensional variables are stripped of large outliers within a 20-timestep window (10 timesteps each direction) around them. This is done by calculating the mean and standard deviation in each window, and removing any values that are above two standard deviations from the window mean. Subsequently, the global mean and standard deviation are calculated, and stored in the class. This means future normalisation of unseen data can be performed using the same mean and standard deviation, ensuring consistency in the data preprocessing.

For spectrogram data, ion temperature and ion density, the normalisation is performed on the logarithm of the data. The logarithm of the data is thus also used in training and inference. This follows the standard feature extraction approach in audio deep learning, a domain which often deals with spectrograms [58].

The data is then grouped into window samples of several timesteps. This is configurable, so depending on user input, several timesteps of spectrogram and variable data will be grouped into a single sample. Reasoning for this transformation is explained in section 5.5.

5.4. Gradient Boosted Decision Tree Baseline (Classifier)

As a baseline model, a decision tree classifier is implemented. This idea is taken from Nguyen et al. (2022, [48]), who used a decision tree classifier to detect near-Earth magnetopause crossings in THEMIS satellite data, and magnetopause crossings in ARTEMIS satellite data. This method will serve here as a baseline to compare the more complex machine learning models against, in a more dynamic environment. One large difference with the method from Nguyen et al. (2022, [48]) is that no spacecraft positional data is used as input for training or inference.

5.4.1. Overview

A gradient boosted classifier is an ensemble learning method that combines multiple decision trees to improve prediction accuracy. Each tree is trained sequentially, with new trees focusing on correcting the errors made by the previous trees. The gradient boosting algorithm in scikit-learn uses the log loss (logistic loss) function for binary classification:

$$L(y,p) = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log(p_i) + (1-y_i)\log(1-p_i)] \tag{5.2}$$

where y_i represents the true region labels (0 for magnetosheath, 1 for magnetotail), p_i is the predicted probability of belonging to the magnetotail region, and N is the number of samples. This loss function penalises incorrect predictions more heavily when the model is confident but wrong.

The algorithm works by fitting each new tree to the residuals (prediction errors) of the ensemble built so far, then adding this tree to the model with a small learning rate to prevent overfitting. This process continues until a specified number of trees is reached or the model converges. In this research, the gradient boosted classifier is used to classify spacecraft measurements as either magnetotail or magnetosheath regions.

5.4.2. Feature Selection and Preprocessing

This includes the ion spectrogram, ion density, ion velocity and ion temperature. The full input dimension of the ion spectrogram is not used. Instead, the spectrogram is averaged over all energy bins per timestep, reducing the input dimension to a single scalar value per timestep. The standard deviation across energy bins is also calculated and used as an additional feature to give the model some information about the distribution of energy bins. This is done to reduce the input dimension, as the 31 energy bins of the ion spectrogram could lead to overfitting, and might make other features have less importance. Timesteps are also not grouped, and instead, single timesteps are used as input.

5.4.3. Crossing Detection Method

To detect magnetotail boundary crossings using the decision tree classifier, a sliding window approach is used. The classifier predicts the class (magnetotail or magnetosheath) for each timestep in the dataset. A crossing is detected when the predicted class changes from magnetotail to magnetosheath or vice versa. To reduce noise and false positives, a window of several timesteps is used to check whether a certain number of half of the timesteps in the window belong to a different class. If this condition is met, a crossing is detected at the centre of the window. The size of the window and the threshold for the number of timesteps can be adjusted to optimise performance. This is the same method as used in Nguyen et al. (2022, [48]) who identified crossings if 1 hour of data contained as many region classifications of one region as the other. This method, while removing a lot of false positives, can also lead to a smaller number of detected crossings close to each other. Additionally, for smaller window sizes, in dynamic environments, with quick successive changes in region predictions, this method can identify multiple crossings very close to each other, when in reality, there might only be a single crossing. To prevent this, an adjustable minimum time between crossings is also implemented, where if two detected crossings are within this time range, the second crossing is removed. An analysis of the impact of this parameter is performed in section 6.1.

An example of the decision tree classifier predictions and crossing detections can be seen in Figure 5.2.

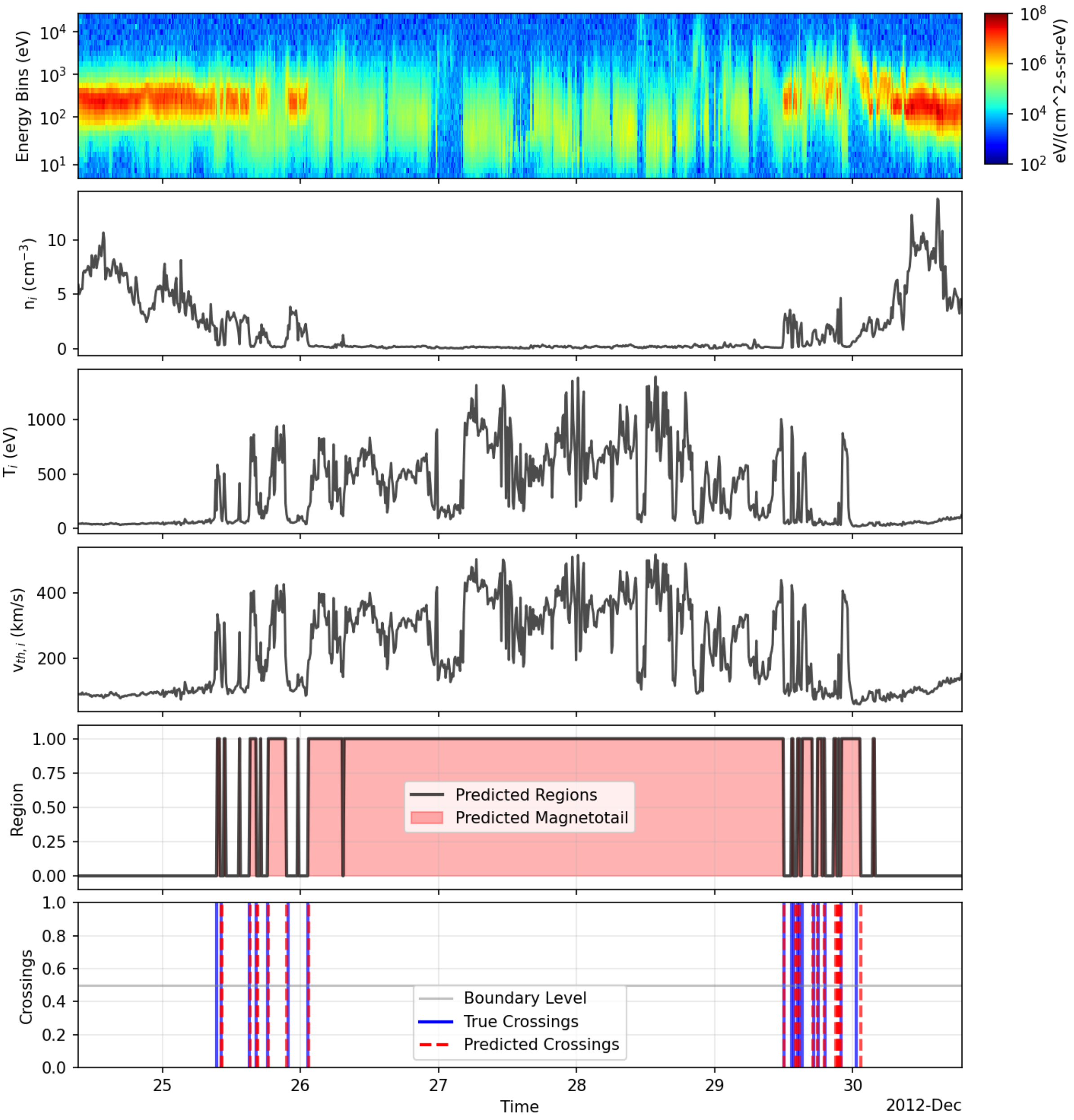


Figure 5.2: Example of decision tree classifier predictions and crossing detections. The top plot shows the ion spectrogram, with the ion density, temperature and thermal velocity below. The two bottom plots show the decision tree classifier predictions (0 = magnetosheath, 1 = magnetotail) and the detected and labelled crossings.

5.4.4. Optimization

An optimisation of the decision tree classifier will be performed. These are not on typical decision tree hyperparameters, but instead on the crossing detection method parameters, as described in subsection 5.4.3, as well as data smoothing. The data smoothing is performed using a rolling mean, where a window of several timesteps is used to average the data. This can help reduce noise in the data and make it easier for the classifier to find patterns. The window size for the rolling mean is a parameter that can be adjusted. Additionally, the minimum time between crossings and the window size for crossing detection are also parameters that can be adjusted. An analysis of the impact of these parameters is performed in subsection 6.3.2 to find the optimal values, which detect crossings matching the labelled crossings as closely as possible, depending on some chosen metrics.

5.5. Reconstruction Based Approach

This section describes the machine learning approach used to detect magnetotail boundary crossings in spacecraft measurement data, using MAE Transformers.

To detect boundary layer crossings in the data, a reconstruction-based approach is used. Hawkins et al. (2002, [22]) was one of the first to introduce the idea of using reconstruction error to detect anomalies in data. The idea is that a model trained on normal data will reconstruct the normal data well, but will struggle to reconstruct anomalous data, which it hasn't seen before. Malhotra et al. (2016, [40]) applied this idea to time series data, using LSTM networks to reconstruct the time series data. This even worked well for noisy and aperiodic data. LSTM networks are a type of recurrent neural network (RNN) that are well-suited for sequential data, as they can capture long-term dependencies in data, but are computationally expensive to train, as they are not parallelisable. In Vaswani et al. (2017, [77]), widely regarded as one of the most influential papers in machine learning research of the last decade, the transformer architecture was introduced, which is a type of neural network that uses self-attention mechanisms to process sequential data. Transformers are highly parallelisable, reducing computational cost, and can be trained on large datasets, and when provided with positional encodings, they can also capture sequential information. CNNs are also widely used in time series data, as well as boundary crossing detection, as seen in Cheng, Achilleos, and Smith (2022, [13]) and Julka et al. (2023, [30]), since they can capture local patterns in the data, and are also highly parallelisable.

Initially, one model versions were tested on detecting boundary crossings. This yielded limited results. Eventually, the following approach was taken:

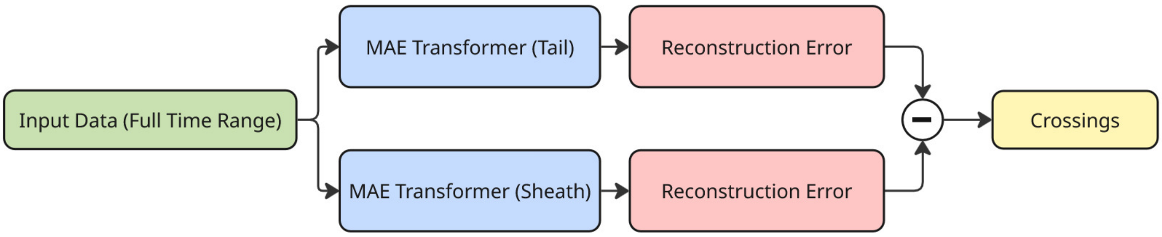


Figure 5.3: The machine learning approach used for boundary crossing detection. The two models are trained on different regions of the magnetosphere, the magnetotail and the magnetosheath. The reconstruction error difference is then used to detect boundary crossings.

Both models were trained on separate regions, the magnetotail and the magnetosheath. The boundary crossing detection is then performed by calculating the difference between the reconstruction errors of the two models:

$$\Delta E(x) = E_{\text{tail}}(x) - E_{\text{sheath}}(x) \tag{5.3}$$

where $E_{\text{tail}}(x)$ and $E_{\text{sheath}}(x)$ are the reconstruction errors of the magnetotail and magnetosheath models, respectively, for input x. The zero crossings of $\Delta E(x)$ indicate potential boundary crossing locations, as the reconstruction error difference changes sign when transitioning between regions.

5.6. MAE Transformer

The Masked Autoencoder (MAE) Transformer is a neural network architecture designed to learn representations by reconstructing masked portions of its input.

5.6.1. Overview

The complete MAE transformer architecture looks as follows:

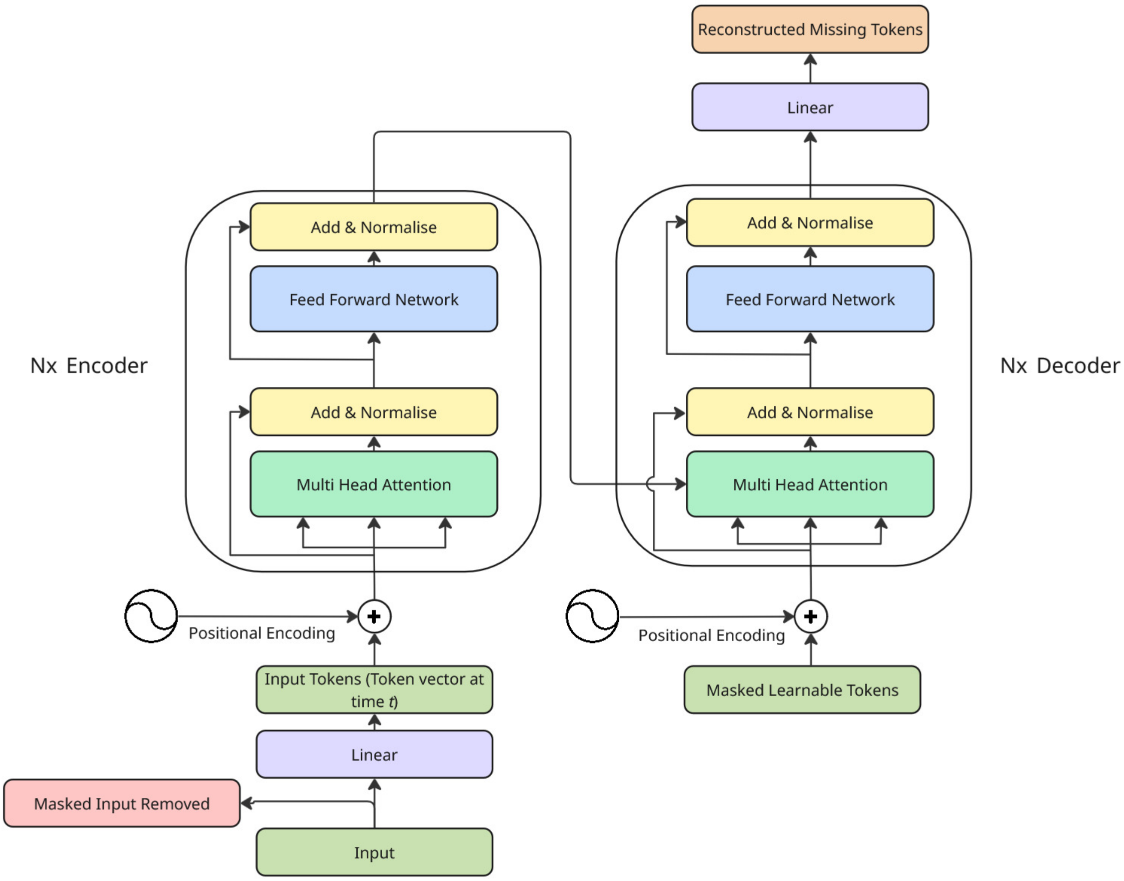


Figure 5.4: The MAE Transformer architecture.

A portion of the input data is masked and not included as input towards the encoder layers. A positional encoding is added to the input tokens, which have been embedded into a higher-dimensional space. Then, input tokens go into an N number of transformer encoder layers. Each layer first performs multi-head self-attention, further explained in 5.6.5. The results of this operation are added and normalised to the input tokens and go into a feed-forward network.

This feed-forward network only works on each token independently. Afterwards, these results are again added and normalised to the input tokens.

The decoder layers use as initial input learnable masked tokens, with positional encodings added to them. The decoder layers also use multi-head self-attention and take information from the encoded tokens to fill in the gaps of information. After an N number of decoder layers, a last linear layer is used to project the output of the decoder layers back to the original input dimension. The model is trained to minimise the reconstruction error between the masked input tokens and the output of the decoder layers, effectively learning to predict the masked portions of the input based on the unmasked ones.

5.6.2. Input Patching

The input data is divided into tokens, which are the basic units of input for the transformer model. For the spectrogram, these are patches, small segments of the spectrogram that capture local patterns in the data. These are typically chosen to be per timestamp, so each spectrogram patch is a vector of energy bins for a single timestamp. In the model used in this research, the number of patches per timestamp is customisable. A visualisation of the input tokens can be seen in Figure 5.5.

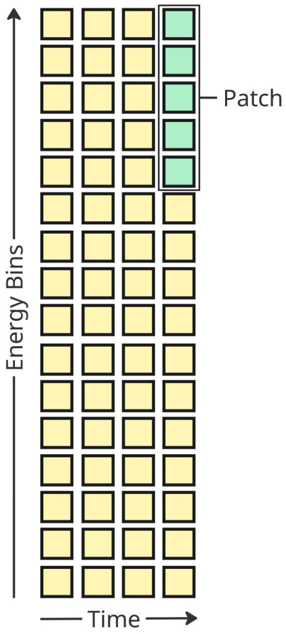


Figure 5.5: Visualisation of the input tokens. The spectrogram is divided into patches, which are then used as input tokens for the transformer model. The patches are typically chosen to be per timestamp, so each patch is a vector of energy bins for a single timestamp.

Each input patch will then be tokenised by using a linear neural network layer, making it an embedded input token.

Patches for non-spectrogram variables are created differently. For vector variables, like three-dimensional magnetic field data, the components are grouped and linearly embedded into the model dimension, and now act as an embedded patch. Scalar variables, like plasma density, velocity, and temperature, are grouped together into a single vector and then linearly embedded into the model dimension, using a consistent linear embedding, acting like a patch as well. After all variables are embedded, they are concatenated together per timestamp, creating a single input token per timestamp.

5.6.3. Input Embedding

The input data, which consists of multivariate time series (e.g., spectrograms and physical variables), is first embedded into a higher-dimensional space. The reasoning is that using a linear neural network layer to embed the input data allows the model to learn more features from the data, like distributions and correlations between the energy bins of a spectrogram. The embedding is performed as follows:

$$z_t = W_{emb} x_t + b_{emb} \tag{5.4}$$

where x_t is an input vector of timestamp t and dimension d, W_{emb} and b_{emb} are the weights and biases (learnable parameters), and z_t is the embedded input. Positional encodings are added to z_t to retain temporal information. This method of positional encoding is taken from Vaswani et al. (2017, [77]), and uses sine and cosine functions of different frequencies to encode the position of each token in the sequence. These can be expressed as follows:

$$\begin{cases} \mathrm{PE}_{(pos,2i)} & = \sin\left(\frac{pos}{10000^{2i/d_{emb}}}\right) \\ \mathrm{PE}_{(pos,2i+1)} & = \cos\left(\frac{pos}{10000^{2i/d_{emb}}}\right) \end{cases} \tag{5.5}$$

where pos is the position of the token in the sequence (timestep in this case), i is the dimension index, and d_{emb} is the model's embedding dimension. This encoding allows the model to understand the order of the input tokens, allowing it to learn time-related patterns. It is added to the embedded input, giving the final input to the transformer encoder layers as:

$$z'_t = z_t + PE \tag{5.6}$$

where z'_t is the final input to the transformer encoder layers, and PE is the positional encoding matrix.

It should be noted that the positional encoding is one-dimensional, while often dealing with spectrograms, which are two-dimensional. A two-dimensional positional encoding, with similar workings as this one, did not lead to better results in initial experiments, and thus the simpler one-dimensional encoding is used.

5.6.4. Masking

The MAE Transformer employs a masking strategy to encourage the model to learn robust representations. This masking is performed by randomly selecting a subset of input tokens to be masked and setting their values to zero. The model is tasked with reconstructing these masked tokens based on the unmasked ones. In the case of the magnetopause crossing spectrograms, the masking is performed on around 30-40% of the input tokens during testing. However, only patches of the spectrogram are masked. This showed increased stability, where the model can still derive context from the other variables. Different masking strategies are used during training, and during validation or during inference. During training, a random masking strategy is used, where a random subset of tokens is masked for each training sample. This encourages the model to learn to reconstruct any part of the input, rather than relying on specific unmasked tokens. During validation and inference, a fixed masking strategy is used, but instead of using a single mask, several hundred different masks are used, and the reconstruction error is averaged over all masks. This provides a more robust estimate of the model's performance, as it reduces the variance in the reconstruction error due to the specific choice of masked tokens.

5.6.5. Self-Attention

The core of the transformer is the self-attention mechanism, which allows the model to weigh the importance of different tokens when encoding information. The attention equation is defined as follows:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V \tag{5.7}$$

where Q, K, and V are the query, key, and value matrices derived from the input embeddings, and d_k is the dimension of the keys. The query matrix Q represents what information each token is seeking from other tokens in the sequence, the key matrix K represents what information each token can provide to others, and the value matrix V contains the actual content that gets aggregated based on the attention weights. Together, these matrices enable the model to determine which parts of the input sequence are most relevant for understanding each position, allowing it to capture complex temporal dependencies in the spacecraft measurement data.

The query, key, and value matrices are computed as follows:

$$Q = W_Q z'_t, \quad K = W_K z'_t, \quad V = W_V z'_t \tag{5.8}$$

where W_Q, W_K, and W_V are learnable weight matrices for the query, key, and value transformations, respectively.

Equation 5.7 produces a context-aware representation for each token that incorporates information from all other tokens in the sequence, with higher attention weights given to more relevant tokens (E.g. close energy bins, close timestamps).

Multi-Head Self-Attention
Multi-head self-attention works by dividing the input embeddings into multiple smaller subsets, each with its own set of query, key, and value matrices. Each subset is processed independently, allowing the model to capture different relationships and patterns in the data. The outputs of these independent attention heads are then concatenated and linearly transformed to produce the final output of the self-attention layer.

Output Projection
The output projection layer maps the decoder's high-dimensional feature representations back to the original input format through three separate linear transformations, similarly to input embeddings. For scalar projection, a linear layer projects decoder outputs to scalar values such as energy measurements and timestamps. Vector projection uses a linear layer to reconstruct multi-dimensional vector features from the learned representations. Finally, spectrogram projection employs a linear layer that maps back to the original spectrogram dimensions while preserving the frequency-time structure.

5.7. Training

Training the MAE Transformer involves updating the model parameters to minimise the reconstruction error between the masked input tokens and the output of the decoder layers.

5.7.1. Updating Model Parameters

As mentioned in section 5.5, the model is trained to minimise the reconstruction error between the masked input tokens and the output of the decoder layers. The reconstruction error is measured using the Mean Squared Error loss function, which is defined as follows:

$$\text{Loss}(y, \hat{y}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{5.9}$$

where y represents the masked input tokens, $\hat{y}$ is the output of the decoder layers. This function is used to calculate gradients with respect to the model parameters, which are subsequently updated using the Adam optimiser.

Like most machine learning models, the MAE Transformer is trained using gradient descent. In this case, the Adam optimiser is used [32]. Adam (Adaptive Moment Estimation) is an optimisation algorithm that combines the advantages of two other extensions of stochastic gradient descent: AdaGrad and RMSprop. It computes learning rates for each parameter by maintaining exponentially decaying averages of past gradients and their squared magnitudes. This allows for efficient training when gradients are low and data is noisy. The adam optimiser updates weights in the machine learning model as follows:

$$w_{t+1} = w_t - \alpha\frac{m_t}{\sqrt{v_t} + \epsilon} \tag{5.10}$$

where w_t is the parameter at time step t, α is the learning rate (a hyperparameter), m_t is the mean of gradients, v_t is the variance of gradients, and ϵ is a small constant to prevent division by zero. The first and second moment estimates are updated as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)\frac{dL}{dw} \tag{5.11}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)(\frac{dL}{dw})^2 \tag{5.12}$$

where $\frac{dL}{dw}$ represents the gradient computed at time step t, and β_1 and β_2 are hyperparameters governing the exponential decay rates for the first and second moment estimates, respectively.

5.7.2. Training Strategy

The training strategy employs a typical train-test split approach, with approximately 85% of the data used for training and 15% reserved for testing. This split is necessary to evaluate the model's performance on unseen data and prevent overfitting. When training loss and test loss diverge, it indicates that the model is learning the training data by heart, and not finding the underlying patterns in the data to generalise well to unseen data. A typical loss curve can be seen in Figure 5.6.

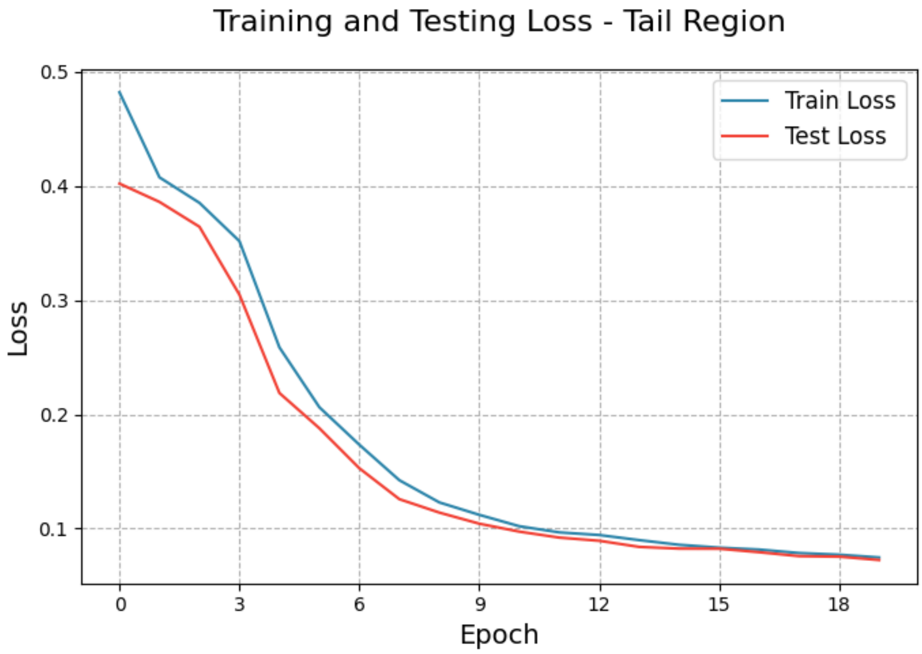


Figure 5.6: Example of a loss curve during training.

For minimising training time, the model is trained in batches, where each batch contains a fixed number of samples. This means that gradients will be computed for each batch, and the model parameters will be updated after each batch is processed.

5.7.3. Measures for Preventing Overfitting

To prevent overfitting, several measures are taken during training. A dropout layer is used after each transformer layer, which randomly sets a fraction of the input units to zero during training. This prevents the model from relying too heavily on specific ”neurons” and encourages it to learn more robust features. Early stopping is also used, where the training process is halted if the test loss does not improve for a certain number of epochs. This prevents the model from continuing to learn the training data by heart, and instead focuses on generalising well to unseen data.

5.8. Conclusion

This chapter described the methods used to detect magnetotail boundary crossings in spacecraft data. The data preparation steps included downloading THEMIS satellite data, converting it to a consistent time format, and storing it in HDF5 files for efficient access. The labelling process used existing lists of known boundary crossings to create training data.

Two main approaches were implemented. First, a gradient boosted decision tree classifier was used as a baseline method, similar to previous work on magnetopause detection. This approach classifies individual data points and uses a sliding window to detect crossings when the predicted region changes.

Second, a reconstruction-based approach using MAE Transformers was developed. This method trains separate models on magnetotail and magnetosheath data, then compares their reconstruction errors to identify boundary crossings. The transformer architecture processes the data using self-attention mechanisms and is trained to reconstruct masked portions of the input.

Both approaches aim to automatically detect boundary crossings in the dynamic magnetotail environment, with the reconstruction method offering an alternative to traditional classification approaches. The training procedures include standard machine learning practices like train-test splits and overfitting prevention measures.

6

Optimisation and Verification

This chapter presents the optimisation and verification of the magnetotail boundary crossing detection models. It covers the dataset splitting strategy, model training procedures, and performance evaluation of both the gradient boosted classifier and MAE transformer approaches. The results show optimal crossing detection parameters for each model type and identify the best-performing models based on performance metrics.

6.1. Model Optimisation, Verification and Validation Strategy

Model optimisation and verification are performed by splitting the dataset into three parts: a training set, a test set, and a validation set. The training set is used to train the model, focused on optimising the model's parameters to minimise the reconstruction error for samples within the training set. The test set is used to evaluate the model's performance during training, seeing whether the model is overfitting to the training data or not. Based on this test set performance, typical hyperparameters such as learning rate, batch size, and model architecture are optimised to prevent overfitting and optimise reconstruction on unseen data. Finally, the validation set is used slightly differently than usual. The validation set will be used to compare crossing detection performance, not reconstruction error, between different models. Based on this performance, the final model will be selected.

6.1.1. Dataset

The dataset used for training, testing and validation is based on Akay, Kaymaz, and Sibeck (2019, [1]), which provides a dataset of several hundred crossings between 2011 and 2013 for the ARTEMIS P1 and ARTEMIS P2 satellites. Using the crossing timestamps, data from the ESA instrument [42], the FGM instrument [3], and the EFI instrument [11] are gathered from the NASA Coordinated Data Analysis Web [44]. Data is split up into either magnetotail or magnetosheath data as described in subsection 5.3.1. All data from before 2012-08-29 ARTEMIS P1, and all data of ARTEMIS P2 (up until August 2013) are used for training and testing, while all data from ARTEMIS P1 after this date is used for validation. This amounts to approximately 25% of all data and crossings. This might seem like much, but it is desired for accurate computation of performance metrics during model validation. The training and test set is split up into 85% training data and 15% test data.

6.2. Model Training

To verify that the model is learning to reconstruct the input data, the reconstruction error on both the training and test sets is monitored during training. The reconstruction error on the test set must be similar to the reconstruction error on the training set, as this indicates that the model is not overfitting to the training data. Both train loss and test loss for both the magnetosheath and magnetotail models can be seen in Figure 6.2 and Figure 6.1, respectively.

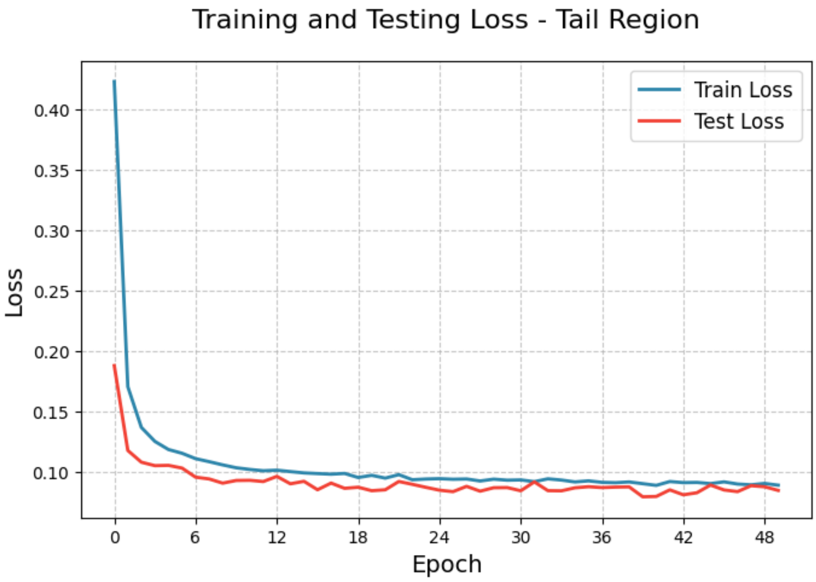


Figure 6.1: Train and test loss during training of the magnetotail model. The loss is the mean squared error between the input and reconstructed output, normalised against the training set mean and standard deviation.

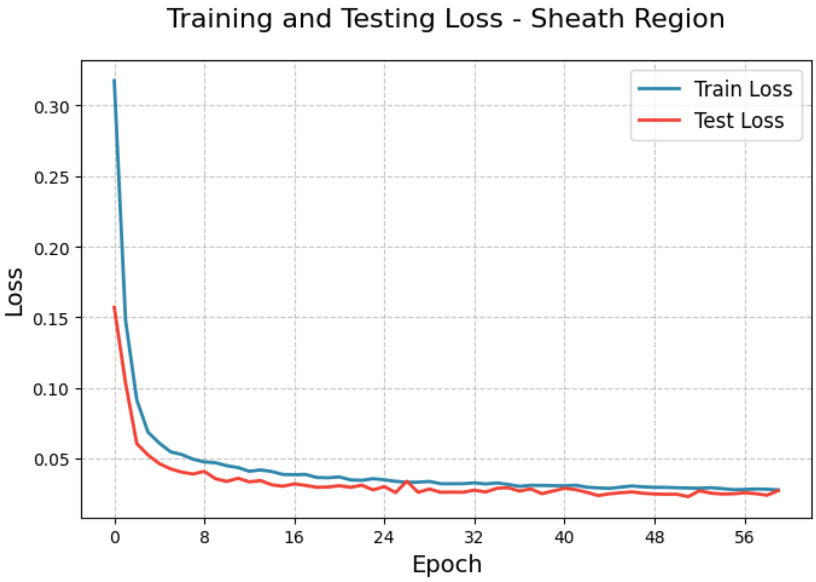


Figure 6.2: Train and test loss during training of the magnetosheath model. The loss is the mean squared error between the input and reconstructed output, normalised against the training set mean and standard deviation.

As can be seen in Figure 6.2 and Figure 6.1, a sharp decrease in reconstruction error occurs in the first 5 epochs, after which the reconstruction error decreases more slowly. The magnetotail seems to settle at a higher average reconstruction error than the magnetosheath, which is likely due to the relative complexity of the magnetotail data. In the end, both models' constrution errors are normalised against their training means and standard deviations during crossing detection, to prevent this difference from affecting crossing detection performance, and possibly shifting crossing detection across time. Something noticable is that the test loss is consistently lower than the train loss. Since the models are set to an evaluation mode during test loss computation, this means that the dropout layers are not active, and thus the model is more robust during testing. This is a good sign, as it indicates that the model is not overfitting to the training data, and is able to generalise well to unseen data.

6.2.1. Model Hyperparameters

The common model parameters used for all MAE transformer models are shown in Table 6.1. These hyperparameters are difficult to optimise; only small performance changes are observed when changing these parameters. Additionally, higher reconstruction performance does not necessarily translate to better crossing detection. Models should be able to reconstruct by learning the general trends in each region, and not overfit on anomalies. One observation was that increasing the model dimension, number of attention heads, encoder and decoder layers, more than 128, 4, 4 and 4, respectively, would hurt training by a lot. Large fluctuations and instability in loss were more common at higher hyperparameter values. The parameters seen in Table 6.1 were selected based on the size of samples (31 energy bins, 1-5 timesteps), which are quite small. Therefore, a small model is sufficient to capture the general patterns in the data, while also being computationally efficient.

Parameter	**Value**
`input_dim`	31
`d_model`	64
`nhead`	2
`num_encoder_layers`	2
`num_decoder_layers`	2
`dim_feedforward`	64
`mask_ratio`	0.4
`patches_per_timestamp`	8
`epochs_per_model`	60
`learning_rate`	1e-3

Table 6.1: Hyperparameters used in the `DoubleTransformer` model.

`input_dim` is the number of input variables, in this case 31, as there are 31 energy bins in the spectrogram. `timestamps` is the number of timestamps per sample, `d_model` is the dimension of the model, the same as the input embedding dimension. `nhead` is the number of attention heads in the multi-head self-attention mechanism, `dim_feedforward` is the dimension of the feedforward network in each transformer layer, `mask_ratio` is the ratio of input tokens that are masked during training, and `patches_per_timestamp` is the number of patches per timestamp in the input data. `epochs_per_model` is the number of epochs that the magnetosheath reconstructor and the magnetotail reconstructor used during training.

The common model parameters used for all GBDT models can be seen in Table 6.2. Again, only small performance changes are observed when changing any of the hyperparameters.

Classification accuracy was consistently above 99%, with little to no change.

Parameter	**Value**
`n_estimators`	100
`max_depth`	6
`learning_rate`	0.1
`min_samples_leaf`	100

Table 6.2: Hyperparameters used in the `GradientBoostingClassifier` baseline.

`n_estimators` is the number of boosting stages (trees) in the ensemble. `max_depth` is the maximum depth of each individual regression tree. `learning_rate` is the shrinkage factor applied to each tree's contribution, trading off step size vs. number of trees. `min_samples_leaf` is the minimum number of samples required at a leaf node, acting as regularisation to reduce overfitting.

6.2.2. Feature Selection

Feature selection was based on two main reasons. The first reason is based on what the most prevalent markers were for magnetotail boundary crossings at lunar distance. This is discussed in Akay, Kaymaz, and Sibeck (2019, [1]), where some of the most important markers were deduced to be ion density, ion velocity magnitude, ion temperature, and ion energy spectrograms. Furthermore, training stability and performance were also taken into account. Features that were too noisy, had an impact on training stability, or did not improve performance were removed. Some derived parameters, like plasma beta, were also considered and tested, but decreased training stability due to high noise levels. The final selected features can be seen in Table 6.3. Furthermore, electron properties were also tested, but did not improve performance, likely due to their similarity to the ion properties. Magnetic field properties were also included in testing. With their inclusion, similar performance was observed overall, but a slight increase in the false positive rate was observed.

Feature	**Description**
Ion Density	Ion density measurements
Ion Velocity Magnitude	Magnitude of ion flow velocity
Ion Temperature	Plasma temperature measurements
Ion Energy Spectrograms	Energy flux as a function of energy and time
Magnetic Field X-Component	Absolute value of GSM Magnetic Field X-Component

Table 6.3: Selected features for model training and their description.

It is important to note that these features were chosen for their stable training results and predictions. However, other features might also be important for improving boundary crossing detection, as will be seen in the subsection 7.3.3. Future work could focus on improving the quality of preprocessing other features, such as more magnetic field data, to make them more suitable for model training and potentially improve performance.

6.3. Model Comparison and Performance Evaluation

Model performance is evaluated using various metrics to assess accuracy and reliability. Metrics will be based on how well magnetotail boundary crossings are predicted compared to labelled data. It is important to note that labelled data may contain inaccuracies, which can affect the evaluation of model performance. For this evaluation, it is assumed that labelled data is accurate enough to indicate model performance based on its use. A large number of models were trained during the pre- and post-processing tuning phase; the best-performing models were selected for further evaluation. Several models were also selected with different goals in mind, such as high precision, high recall, or a balance.

6.3.1. Performance Metrics

Before discussing the performance metrics, it is important to clarify what predictions are considered to be true positives, false positives, true negatives and false negatives. Since the magnetotail at lunar distances is very dynamic, multiple crossings can occur within a short time frame. In the labelled dataset, crossings are marked with a timestamp, which can be as close as 5 minutes apart from each other. Since data is sampled every 10 minutes, it is hard to detect crossings at this resolution. Therefore, (and also based on the requirements stated in Appendix B) if a model predicts a crossing within a 30-minute radius of a labelled crossing, it is considered a true positive. If labelled crossings are within 10 minutes of another labelled crossing which has been predicted, it is not considered a false negative, but simply removed from the performance evaluation. False positives are predictions that are not within 30 minutes of a labelled crossing, or a labelled crossing with another detected crossing closer to it. True negatives are not well-defined in this reconstruction-based approach, as any timestamp without a labelled crossing could theoretically constitute a true negative, leading to an unbounded negative class. Instead, we focus on precision-based metrics and false positive rates, which are more meaningful for time series crossing detection.

Several performance metrics are used to evaluate the models, each with a certain goal in mind. The most important ones are discussed here:

Precision is used to measure the accuracy of the models in predicting true positives. This is important as false positives (incorrectly predicting a magnetopause crossing) can lead to wrong conclusions in magnetotail studies, potentially affecting the interpretation of the magnetotail's interaction with the magnetosheath. High precision ensures that when the model identifies a boundary crossing, it is likely to be correct. Precision is calculated as:

$$\text{Precision} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} \tag{6.1}$$

Recall measures the model's ability to identify all actual boundary crossings in the dataset. High recall ensures that boundary crossing databases are comprehensive, allowing a more detailed analysis of magnetotail dynamics. Recall is calculated as:

$$\text{Recall} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} \tag{6.2}$$

The **F1-score** provides a balanced measure between precision and recall, offering a single metric that considers both false positives and false negatives. This is valuable when comparing different models, as it accounts for the trade-off between identifying all boundary crossings and maintaining prediction accuracy. The F1-score is calculated as:

$$\text{F1-score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6.3}$$

F0.5-score is also reported, as it weights precision higher than recall. This is useful in scenarios where false positives are more detrimental than false negatives. In the context of magnetotail boundary crossing detection, a false positive could lead to incorrect interpretations of plasma dynamics, while a false negative might simply mean missing out on some events. The F0.5-score is calculated as:

$$\text{F0.5-score} = (1 + 0.5^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{(0.5^2 \cdot \text{Precision}) + \text{Recall}} \tag{6.4}$$

Temporal precision is also an important metric, as it indicates how close the predicted crossing times are to the actual crossing times. Since these spacecraft move quickly through the magnetotail, temporal inaccuracies can lead to fairly significant spatial inaccuracies. Therefore, metrics such as the mean and standard deviation of the time difference between true positive predictions and actual crossing times are also reported.

The **F0.5-score and precision will serve as the primary criteria** for model evaluation and selection. This is to emphasise the importance of precision over recall, as wrong boundary crossing detections can lead to wrong interpretations of magnetotail behaviour. A false positive could result in wrong conclusions about magnetotail structure, while a false negative will likely have less of an effect on scientific analyses.

6.3.2. Impact of Pre- and Post-processing on Model Performance

The impact of pre- and post-processing techniques is high, and thus, a detailed analysis of these techniques is performed to find the optimal settings. This is done for both the gradient boosted classifier and MAE transformer models.

Gradient Boosted Classifier:
To find the best gradient boosted classifier model, a grid search was performed over the window size settings, as discussed in subsection 5.4.3, and for how many timesteps crossing windows could not overlap. This yielded these results for an ROC-style curve Figure 6.3, as well as precision, recall, F0.5 and F1 score curves in Figure 6.4, Figure 6.5, Figure 6.6a, and Figure 6.6b respectively.

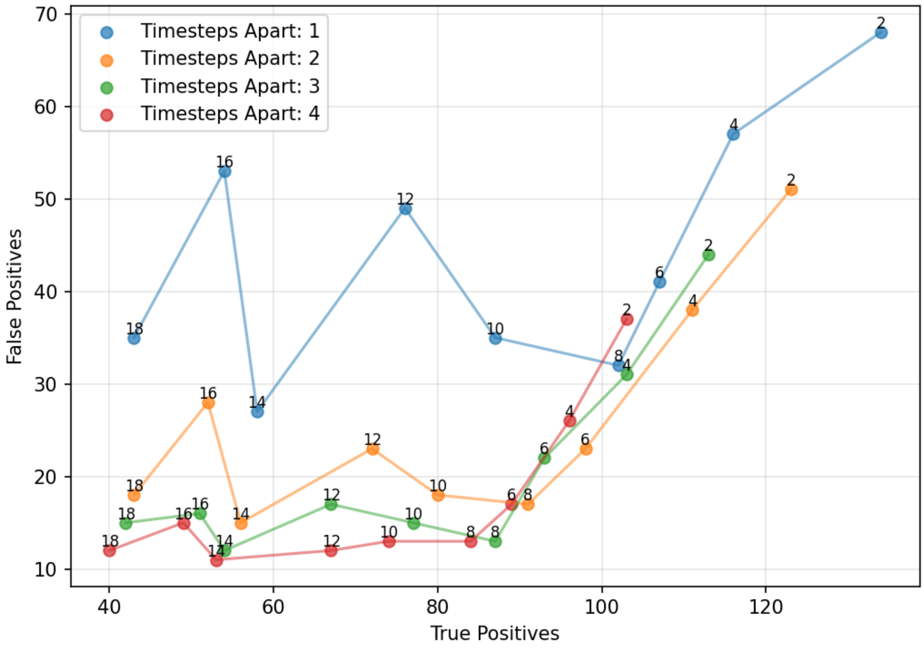


Figure 6.3: ROC-style curve for gradient boosted classifier models with different window sizes and non-overlapping timestep settings. Each point represents a different model configuration. The number at each datapoint denotes the detection window size. The goal is to maximise the true positives while minimising the false positives.

Figure 6.3 shows that there is a clear trade-off between true positives and false positives. Models with smaller window sizes and less restrictive non-overlapping timestep settings tend to have higher true positive rates, but also higher false positive rates. Conversely, models with larger window sizes and more restrictive non-overlapping timestep settings tend to have lower false positive rates, but also lower true positive rates. This trend is particularly visible at a window size of 8 or higher. This is mostly as expected, as smaller window sizes allow the model to be more sensitive to changes in the data, but also more prone to false alarms. The optimum depends on the specific use case, but a good balance is often desired. In the case of magnetotail boundary crossing detection, a strong emphasis is placed on minimising false positives, and thus models with a high precision and F0.5 scores are preferred.

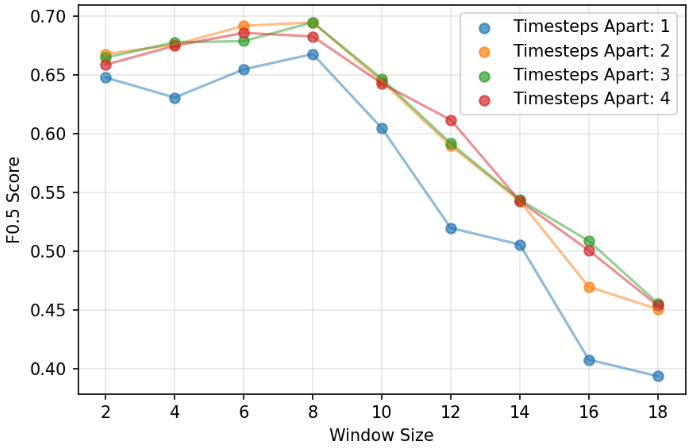


Figure 6.4: Model F0.5 Score of boundary crossing predictions with different window sizes and non-overlapping timestep settings.

According to Figure 6.4, models with a window size of 6 or 8 and non-overlapping timestep settings of 2 to 4 tend to have the highest F0.5 scores. This indicates that these models strike a good balance between precision and recall, with a slight emphasis on precision. Additionally, for window sizes of 10, the F0.5 score tends to only slightly decrease, meaning that these models are still quite effective. The reasoning for this is quite simple. The timesteps apart requirement forces the model to make fewer predictions in areas where a crossing occurs, making it pinpoint only one crossing in a certain time frame. This reduces the number of false positives close to crossing regions. Additionally, the higher window sizes of 4, 6, and 8 (compared to 2) allow the model to better capture the temporal context of data, leading to fewer false positives as well. Since these window sizes are not too large, the model can still be sensitive enough to detect crossings.

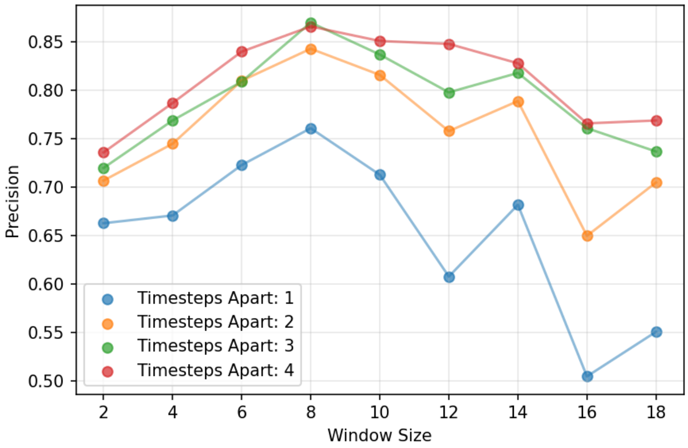
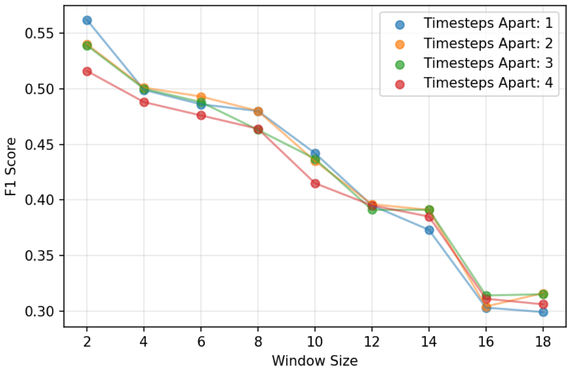
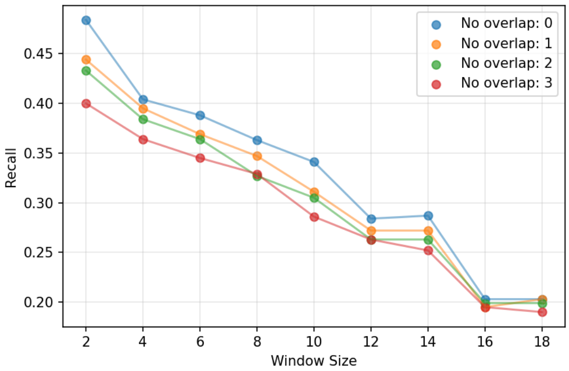


Figure 6.5: Model Precision of boundary crossing predictions with different window sizes and non-overlapping timestep settings.

Figure 6.5 shows that precision tends to be higher for models with larger window sizes and more restrictive non-overlapping timestep settings, up until a window size of 8, where the precision score flattens out, and decreases afterwards. This is for similar reasons to the F0.5 score, as larger window sizes and more restrictive non-overlapping timestep settings allow the model to better capture the temporal context of data, leading to fewer false positives. However, if the window size is too large, the model may become less sensitive to changes in the data, leading not to more false positives, but rather more false negatives, which in turn lowers precision.



(a) Model F1 Score of boundary crossing predictions with different window sizes and non-overlapping timestep settings.



(b) Model Recall of boundary crossing predictions with different window sizes and non-overlapping timestep settings.

The trend in F1 score and Recall is more one-sided, due to the higher restrictions of the higher window sizes and non-overlapping timestep settings. As can be seen in Figure 6.6a and Figure 6.6b, A steady decrease in both metrics is observed as window size and non-overlapping timestep settings increase.

Based on these results, three models were selected for further evaluation: a model with a high F0.5 score, a model with high precision, and a model with a more balanced approach. The parameters and performance metrics of these models can be seen in Table 6.4. An averaging window of 3 was used. While this was not specifically quantitatively optimised for, it seemed to offer the best performance across the whole spectrum of models. Lower values would significantly increase the number of false positives. The model with the highest F0.5 score and the model with the highest precision are the same. This is not unusual; F0.5 score sets a high weight on precision, and thus the model with the highest precision will often also have a high F0.5 score. The balanced model will therefore be chosen to have a higher recall, at the cost of some precision, for comparison purposes to MAE models later on.

Table 6.4: Model parameters and performance metrics for selected models.

Model Parameters	**High F0.5**	**High Precision**	**Balanced**
Averaging window	3	3	3
Window size	8	8	6
Non-overlapping timesteps	3	3	2
Performance Metric			
F0.5 Score	0.695	0.695	0.692
Precision	0.870	0.870	0.810
F1 Score	0.534	0.534	0.568
Recall	0.385	0.385	0.438
True Positives	87	87	98
False Positives	13	13	23
False Positive Rate	0.130	0.130	0.190
False Negative Rate	0.615	0.615	0.562
In Range Positives	120	120	129
Out Range Predicted Positives	13	13	21
False Negatives	139	139	126
Total True Crossings	235	235	235
Total Predicted Crossings	100	100	121
Time Window (min)	30	30	30
TP Time Accuracy Mean (min)	12.590	12.590	12.590
TP Time Accuracy Std (min)	7.540	7.540	7.370
Variables Used	n_i, $\|\mathbf{v}_i\|$, T_i, Φ_i	n_i, $\|\mathbf{v}_i\|$, T_i, Φ_i	n_i, $\|\mathbf{v}_i\|$, T_i, Φ_i

In the table above, the variables used for the models are: ion density (n_i), magnitude of ion flow velocity ($|\mathbf{v}_i|$), ion thermal velocity ($v_{th,i}$), ion temperature (T_i), and ion energy flux (Φ_i) using the spectrograms' mean and standard deviation per timestamp. These parameters represent the key plasma properties used to train the different model variants.

MAE Transformer:

The MAE transformer models have pre- and post-processing features that act similarly to the gradient boosted classifier's features discussed in subsection 5.4.3. A similar analysis is done for these features to find the optimal settings. Since the MAE transformer models use a reconstruction-based approach, the window size and non-overlapping timestep settings are not applicable. Instead, the number of timesteps used as input, the averaging window size in pre-processing, and the post-processing reconstruction error averaging window size are varied. This resulted in 48 models with different hyperparameter combinations. The relation between the earlier discussed metrics isn't as clear as with the gradient boosted classifier models, but some trends can still be observed. One important note is that models were trained with the ion density, ion velocity and ion temperature variables, or with ion density, ion thermal velocity and ion temperature. An asterisk (*) denotes models trained with ion thermal velocity instead of ion velocity. In general, both versions of the models perform similarly. All models in the figures are trained with ion velocity.

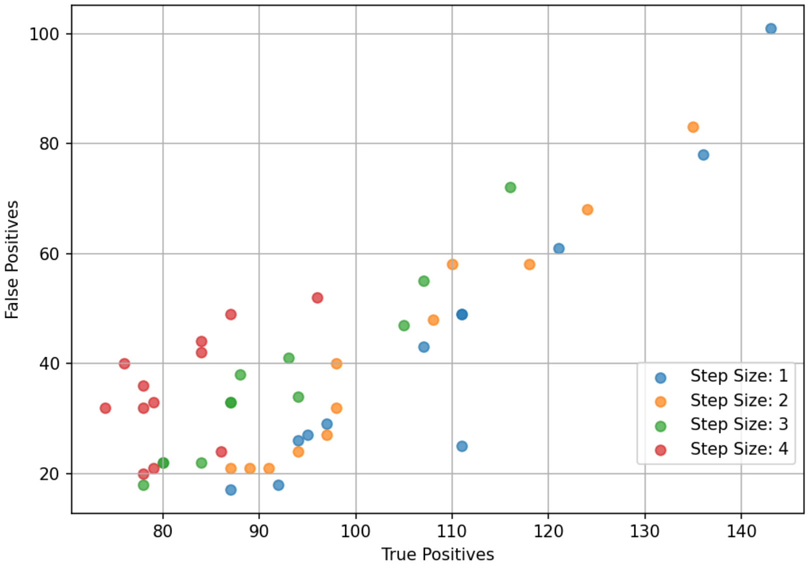


Figure 6.7: ROC-style curve for MAE transformer models with different numbers of timesteps. Each point represents a different model configuration. This figure is shown as a general trend to see how step size influences the distribution of true positives and false positives.

As can be seen from Figure 6.7, models with a higher number of timesteps tend to struggle more with detecting a large amount of true positives, even for models where sensitivity is higher (low data averaging window and low reconstruction error averaging window), where false positives increase.

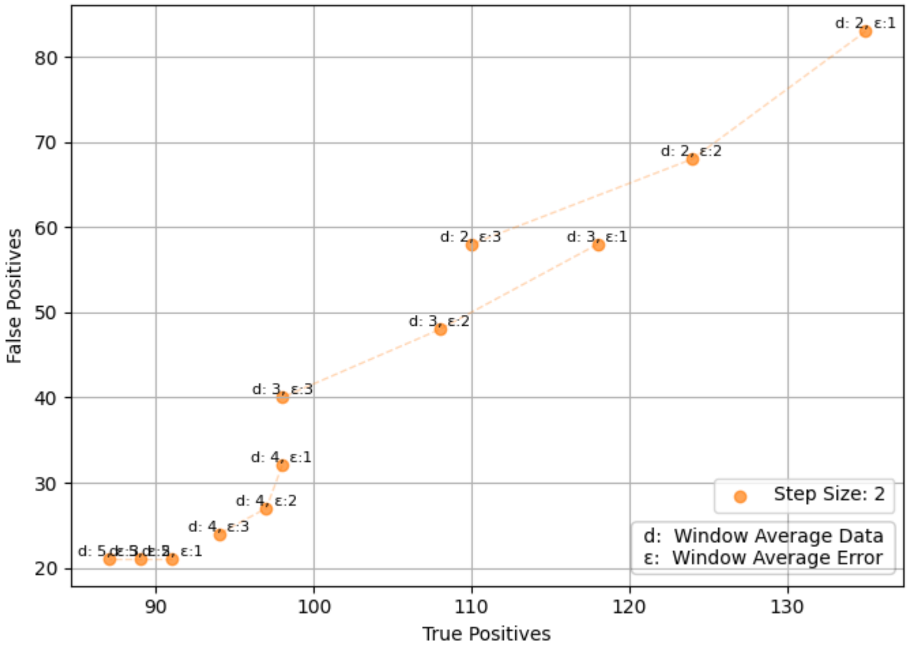


Figure 6.8: ROC-style curve for MAE transformer models with 2 timesteps. Each point represents a different model configuration. The number at each datapoint denotes the data averaging window size and reconstruction error averaging window size, respectively.

Figure 6.8 shows the ROC-style curve for models with 2 timesteps. Both window averaging data and window averaging reconstruction error affect the model's ability to detect true positives while reducing false positives. Window averaging errors seem to have a larger effect on this, heavily reducing both true positives and false positives as it increases. Window averaging data seems to incite large decreases in false positives, but not as high a decrease in true positives, and sometimes even increases true positives.

The mixing of these 3 parameters makes it hard to find clear trends in the other performance metrics, such as precision, recall, F1 score and F0.5 score. Different combinations of these parameters can lead to similar performance metrics, even if the parameters themselves are quite different. Based on the performance metrics, three models were selected for further evaluation. One with a high precision (High Precision), one with a good balance between precision and recall (Balanced), and one with a high F0.5 score (High F0.5). Full model performance metrics can be seen in Table 6.5.

Table 6.5: Model parameters and performance metrics for selected models.

Model Parameters	**High F0.5**	**High F0.5***	**High Precision***	**Balanced***
# Timesteps	1	1	3	1
Data averaging window	3	3	4	4
Error averaging window	3	2	3	2
Performance Metric				
F0.5 Score	0.715	0.720	0.685	0.685
Precision	0.804	0.782	0.853	0.833
F1 Score	0.613	0.644	0.529	0.541
Recall	0.496	0.547	0.383	0.400
True Positives	111	122	87	90
False Positives	27	34	15	18
False Positive Rate	0.196	0.218	0.147	0.167
False Negative Rate	0.504	0.453	0.617	0.600
In Range Positives	139	151	118	121
Out Range Predicted Positives	25	33	15	17
False Negatives	113	101	140	135
Total True Crossings	235	235	235	235
Total Predicted Crossings	138	156	102	108
Time Window (min)	30	30	30	30
TP Time Accuracy Mean (min)	6.92	6.63	8.94	8.46
TP Time Accuracy Std (min)	6.19	5.70	7.07	5.87
Variables Used	n_i, $\lvert\mathbf{v}_i\rvert$, T_i, Φ_i	n_i, $v_{th,i}$, T_i, Φ_i	n_i, $v_{th,i}$, T_i, Φ_i	n_i, $v_{th,i}$, T_i, Φ_i

In the table above, the variables used for the models are: ion density (n_i), magnitude of ion flow velocity ($|\mathbf{v}_i|$), ion thermal velocity ($v_{th,i}$), ion temperature (T_i), and ion energy flux (Φ_i) using the full spectrogram. These parameters represent the key plasma properties used to train the different model variants.

Two models were selected with a high F0.5 score, one trained with ion velocity and one with ion thermal velocity due to their proximity in critical performance metrics, but their larger difference in others.

Now that both the gradient boosted classifier and MAE transformer models have been evaluated based on their pre- and post-processing features, the best performing models from each category will be further evaluated and compared in subsection 7.1.2.

6.3.3. Computational Efficiency

Computational efficiency is an important factor to consider when evaluating the practicality of using these models for real-time or large-scale applications. All models with use only timestamps have a major advantage in this regard. As discussed earlier, models use a pre-generated set of masks during inference to stabilise predictions. For multiple timestamps, this number needs to be quite high to ensure good stability, and is therefore set at 1000. This produced very stable results, with the same crossing predictions no matter the set of masks used. If 1 timestep is used, this number can be reduced significantly due to the number of possible masks being 70. This means that the model only needs to perform 70 forward passes through the model, instead of 1000, leading to a significant reduction in computation time. For the validation set, 1 year of data, this meant a difference of 20 seconds compared to 300 seconds to run inference for the 3-timestep models.

6.3.4. Inhibition Gate Improvement

To improve the MAE models' performance, some reconstruction error-based approaches can be used. If both MAE models (magnetotail and magnetosheath) have high reconstruction errors before and after a candidate crossing, the data is likely either anomalous or not part of either region. Both models seem to recognise the region. Therefore, a further filter can be applied to remove candidate crossings where both models have extremely high reconstruction errors before and after the crossing. This can be done by setting a threshold based on the mean and standard deviation of reconstruction errors in both regions during training. This check will essentially act as an inhibition gate, only allowing crossings to be detected if at least one model has a reasonable reconstruction error before or after the crossing. The mathematical expression for this gate is as follows:

$$(min(E_{tail,i}, E_{tail,i+1}) < \mu_{tail} + 2\sigma_{tail}) \vee (min(E_{sheath,i}, E_{sheath,i+1}) < \mu_{sheath} + 2\sigma_{sheath}) \tag{6.5}$$

This method was tested on a high F0.5 score model (with the absolute value of the magnetic field GSM X component added). And led to greatly increased precision and a slight decrease in recall. The F0.5 score saw a significant increase. The improvement can be seen in Table 6.6.

Table 6.6: Model parameters and performance metrics for selected models.

Model Parameters	**Inhibitor Gate**	**No Inhibitor Gate**
# Timesteps	1	1
Data averaging window	3	3
Error averaging window	3	3
Performance Metric		
F0.5 Score	0.731	0.715
Precision	0.912	0.837
F1 Score	0.564	0.587
Recall	0.408	0.452
True Positives	93	103
False Positives	9	20
False Positive Rate	0.088	0.163
False Negative Rate	0.592	0.548
In Range Positives	117	126
Out Range Predicted Positives	9	20
False Negatives	135	125
Total True Crossings	235	235
Total Predicted Crossings	102	123
Time Window (min)	30	30
TP Time Accuracy Mean (min)	7.25	7.46
TP Time Accuracy Std (min)	6.14	6.28
Variables Used	n_i, $\lvert\mathbf{v}_i\rvert$, T_i, Φ_i, $\lvert B_x\rvert$	n_i, $v_{th,i}$, T_i, Φ_i, $\lvert B_x\rvert$

These models also used the absolute value of the magnetic field GSM X component ($|B_x|$) as an additional feature. As can be seen, the inhibitor gate significantly improves precision, at the cost of some recall. The F0.5 score sees a significant increase, indicating that this method is effective for improving model performance.

Instead of the usage of the other MAE models for further comparison, this inhibitor gate method will be used for the final MAE model, as it significantly improves performance.

6.4. Conclusion

This chapter showed how well both gradient boosted classifiers and MAE transformers can detect magnetotail boundary crossings. Several key findings emerged from the analysis:

Pre- and post-processing settings made a big difference in how well the models performed. For gradient boosted classifiers, window sizes of 6-8 with non-overlapping timestep settings of 2-4 gave the best F0.5 scores. For MAE transformers, models using 1-3 timesteps with moderate data and reconstruction error averaging windows (3-4 samples) worked best for balancing precision and recall.

The MAE transformers' reconstruction approach worked well on unseen data, with test losses consistently lower than training losses during training, likely due to the disabling of dropout. Both model types converge, with strong improvements at first followed by gradual refinement.

The optimisation process successfully found the best hyperparameters and configurations for both approaches, creating a solid methodology for automated boundary detection. The systematic evaluation of pre- and post-processing techniques gives clear guidance for future use of these methods in magnetotail research.

7

Results

This chapter presents the results of the model training and evaluation process. The performance of the selected models is analysed based on various metrics, including precision, recall, F-scores, and temporal accuracy of detected magnetopause crossings. The results are compared against the labelled dataset of known crossings, and the physical plausibility of detected crossings is validated through spatial distribution analysis in relation to established magnetopause models. Outliers in the detected crossings are investigated to understand potential causes and implications for model performance.

7.1. Training Results

This section evaluates the performance of the MAE transformer and gradient boosted classifier models developed in Chapter 6. The analysis compares quantitative metrics like precision, recall, and F-scores, examines model predictions and their physical interpretations, and validates detected crossings against established magnetopause models. Finally, the results of the generated crossing datasets are presented.

7.1.1. MAE Reconstruction Examples

Some examples of the reconstruction of spectrogram samples can be seen in figures 7.1, 7.2 and 7.3. In this case, the samples have 4 timesteps. These samples are taken from a validation set. The validation set was not used in either the training or test sets.

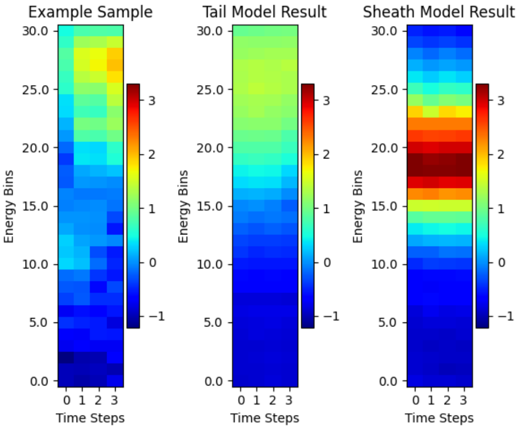


Figure 7.1: Example of a magnetotail spectrogram sample. The recreation by the MAE magnetotail transformer is shown in the middle plot, and the recreation by the MAE magnetosheath transformer is shown on the right.

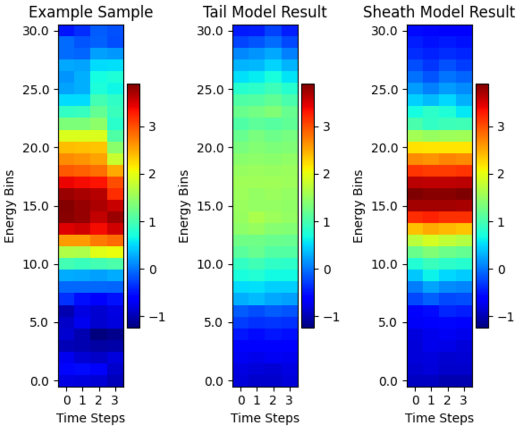


Figure 7.2: An example of a magnetosheath spectrogram sample. The recreation by the MAE magnetotail transformer is shown in the middle, and the recreation by the MAE magnetosheath transformer is shown on the right.

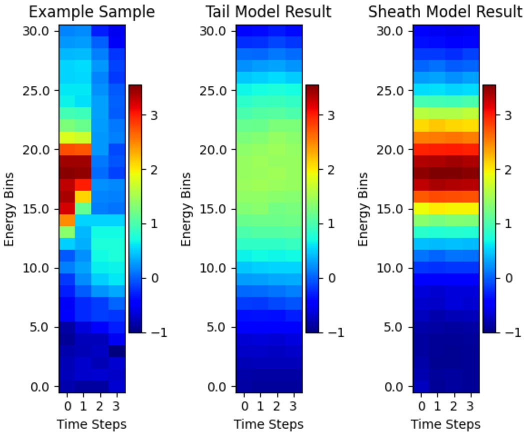


Figure 7.3: Example of a magnetopause crossing sample. The middle plot shows the reconstruction by the MAE magnetotail transformer, and the right plot shows the reconstruction by the MAE magnetosheath transformer.

These recreations show that the models are able to reconstruct the spectrogram samples from regions they are trained on well, and others less so. The crossing is difficult to reconstruct for both models. It should be noted, of course, that reconstruction error is only calculated on patches that were masked. In Figure 7.1, Figure 7.2, and Figure 7.3, the full ”recreation” is shown, which includes both masked and unmasked patches. The reconstruction error is only calculated on the masked patches.

7.1.2. Quantitative Results

In this subsection, a comparison is made between the best-performing gradient boosted classifier and MAE transformer models. For the gradient boosted classifier, both a high precision model, a model based on a balance between precision and F0.5 score, and a high F0.5 model are compared to the best performing MAE transformer model in Table 7.1, Table 7.2 and Table 7.3, respectively. The MAE transformer model used for comparison is the one using ion velocity and the X-component of the magnetic field, with additional inhibition methods, as described in subsection 6.3.4.

Table 7.1: Model parameters and performance metrics for high precision MAE transformer and gradient boosted classifier models.

Performance Metric	MAE Model	Classifier Precision
Precision	0.912	0.870
F0.5 Score	0.731	0.695
F1 Score	0.564	0.534
Recall	0.408	0.383
True Positives	93	87
False Positives	9	13
False Positive Rate	0.088	0.190
False Negative Rate	0.592	0.562
TP Time Accuracy Mean (min)	7.25	12.590
TP Time Accuracy Std (min)	6.14	7.54

The MAE transformer model outperforms the gradient boosted classifier, particularly in temporal accuracy and F0.5 scores. While classifiers achieve close precision (0.912 vs 0.870), they struggle with temporal accuracy, a lower recall, and still a higher amount of false positives. In the case of high-precision gradient boosted decision tree models, the MAE model also achieves better recall, detecting more true positive crossings.

Table 7.2: Model parameters and performance metrics for balanced MAE transformer and gradient boosted classifier models.

Performance Metric	MAE Model	Classifier Balanced
Precision	0.912	0.810
F0.5 Score	0.731	0.692
F1 Score	0.564	0.568
Recall	0.408	0.438
True Positives	93	98
False Positives	9	23
False Positive Rate	0.088	0.165
False Negative Rate	0.592	0.623
TP Time Accuracy Mean (min)	7.25	12.59
TP Time Accuracy Std (min)	6.14	7.37

The balanced MAE model maintains strong performance across key metrics when compared to the balanced classifier. The MAE model achieves significantly higher precision (0.912 vs 0.810) while the classifier achieves slightly better recall (0.438 vs 0.408) at the cost of significantly more false positives (23 vs 9). Both models show similar F1 scores, but the MAE model's superior temporal accuracy (7.25 vs 12.59 minutes) continues to demonstrate its advantage in precise crossing detection. Since the precision and F0.5 scores were deemed to be more important than recall, the MAE model is preferred here as well.

Table 7.3: Model parameters and performance metrics for higher recall score MAE transformer and gradient boosted classifier models.

Performance Metric	MAE High F0.5	Classifier High Recall
Precision	0.912	0.745
F0.5 Score	0.731	0.676
F1 Score	0.564	0.594
Recall	0.408	0.493
True Positives	93	111
False Positives	9	38
False Positive Rate	0.088	0.255
False Negative Rate	0.592	0.507
TP Time Accuracy Mean (min)	7.25	13.530
TP Time Accuracy Std (min)	6.14	6.720

When evaluating models optimised for higher recall, the MAE transformer model continues to perform better in most key metrics compared to the gradient boosted classifier. The MAE model, again, has high precision (0.912 vs 0.745). Although the classifier detects more crossings overall (111 vs 93), this comes at a significant cost of substantially more false positives (38 vs 9), resulting in a much higher false positive rate and lower F0.5 score. The MAE model's temporal accuracy advantage remains consistent. While the classifier achieves marginally better F1 scores, the MAE model's superior F0.5 score (0.731 vs 0.676) reflects better performance when precision is prioritised over recall.

Overall, the MAE transformer model consistently outperforms the gradient boosted classifier across all evaluated configurations based on the previously determined most important metrics. It achieves higher precision, better temporal accuracy, and a more favourable balance between true positives and false positives. The combination of the MAE architecture and the ability to interpret reconstruction errors allows for additional optimisation methods, which proved more effective for the magnetopause crossing detection task than the gradient boosted decision tree approach.

7.1.3. Requirements Evaluation

The requirements set out in chapter 3 based on the stakeholder analysis are evaluated here based on the results presented in subsection 7.1.2. Requirements can be found in Appendix B. The requirements satisfaction is summarised below:

- **SR-1** - The system accepts in-situ spacecraft data as input. While the specific data sources listed in this requirement are not all utilised, using the data flow, it is easy to train models with the mentioned data sources as well.
- **SR-2** - The system automatically preprocesses input data by pulling data from CDAWeb, cleaning it, normalising it, and creating time-windowed inputs.
- **SR-3** - The system outputs timestamps of detected magnetopause crossings within the input time period. This is done with a precision of several minutes.
- **SR-4** - The MAE model can pinpoint each crossing with a maximum error of 20 minutes 97% of the time, assuming a half-normal distribution of timing errors.
- **SR-5** - Both classifier and MAE models can process input data for magnetopause crossing detection within 1 hour. Multiple-step models can too, but with lower consistency.

The classifier inference takes 5 seconds for 1 year of data. MAE model inference takes 20 seconds for 1 year of input data, in the case of 1-timestep models. Multiple-timestep models take 6 minutes for 1 year of input data.

- **SR-6** - The system uses ARTEMIS mission data for training and evaluation.
- **SR-7** - The system can output detected crossings in a CSV format, which can be easily integrated with downstream analysis tools or databases.
- **SR-8** - The system provides an interface to ingest input data from the NASA Coordinated Data Analysis Web (CDAWeb) database, and automatically preprocesses it.
- **SR-9** - The system is designed to operate on a single computer equipped with a Ryzen AI 9 hx 370 and an NVIDIA RTX 4060 laptop GPU.
- **SR-10** - The system can process multidimensional time series data, including 1D data, vector data, and 2D data (spectrograms).
- **SR-11** - The MAE model has a precision of 91.2% when evaluated against a labelled dataset of known magnetopause crossings.

Overall, all requirements are met by the developed system, with the slight exception of SR-1, where not all data sources are used, but the system is designed to be easily extendable to include these data sources.

7.2. MAE Prediction Interpretation

To understand how the model makes its predictions, a qualitative feature importance analysis is performed. This involves examining which input features the model relies on most heavily when making predictions about magnetotail boundary crossings. Two MAE models will be analysed. For this, the high F0.5* using ion thermal velocity is used for predictions, as described in Table 6.5. Other models show very similar behaviour, with models with high precision setting harsher thresholds for what constitutes a crossing.

The high F0.5* score model's predictions on a full magnetotail pass of ARTEMIS P1 are shown in Figure 7.4. The model seems to pick up on several crossings, but also misses some others. Additionally, some crossings are detected fairly far away from the labelled crossing time.

As seen from Figure 7.4, the model seems to pick up on crossings where there is a clear change in the spectrogram, ion temperature, density and velocity. Detected crossings are very close to labelled crossings in these cases, with one exception being the last detected crossing. To find out what the differences are between detected and missed crossings, a closer view of some of these crossings is shown in Figure 7.5.

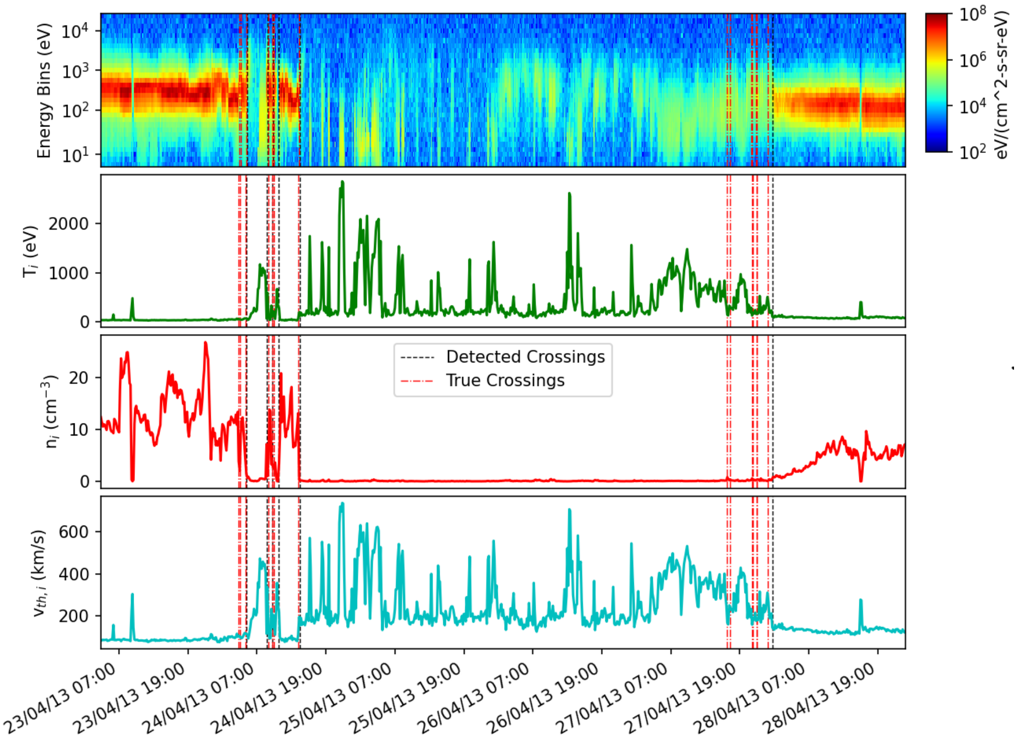


Figure 7.4: ARTEMIS P1 full pass showing features and detected crossings. The top plot shows the spectrogram, the following plots show average ion temperature, density and velocity in that order. Detected crossings are shown as vertical dashed black lines, and labelled crossings as vertical dashed dotted red lines.

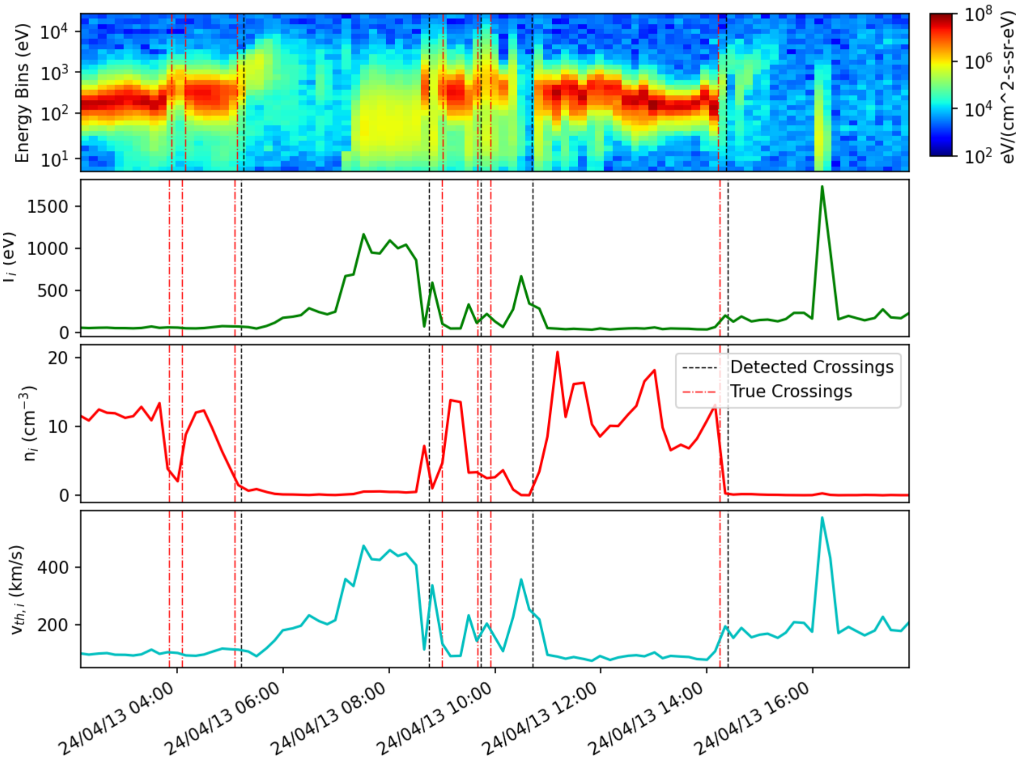


Figure 7.5: Close-up of ARTEMIS P1 full pass showing features and detected crossings. The top plot shows the spectrogram, the following plots show average ion temperature, density and velocity in that order. Detected crossings are shown as vertical dashed black lines, and labelled crossings as vertical dashed dotted red lines.

In Figure 7.5, the difference in detected and missed crossings is clearer. Within this snippet, and in other snippets as well, the model seems to require harsher changes in ion density and the ion energy flux to detect a crossing. The 2 crossings at around 04:00 are not detected, likely because the changes in ion density and energy flux are not as pronounced as in crossings as those at ~5:15 and ~14:30. Additionally, the crossing at ~9:00 is detected slightly earlier than the labelled crossing. An initial spike in ion density and energy flux is seen at ~8:50 (and a decrease in ion temperature and velocity), which settles down, to pick up right afterwards. Since data is window-averaged, this initial spike smoothens with the later increase, and the model sees a more gradual increase, leading to an earlier crossing detection. This happens again with the first labelled crossing at ~9:50, where small spikes are smoothed into a more gradual decrease, which makes the model detect the crossing slightly later than the labelled crossing. Most notably, the crossing at 10:00 (the second of the two labelled crossings around that time) is detected significantly later (at ~10:45) than the labelled crossing. Looking at the spectrogram, ion temperature, density and velocity, this makes a lot of sense. Large changes occur in all variables, which can be seen very pronouncedly in the spectrogram, which would constitute a crossing. The labelled crossing is likely slightly inaccurate. Since the detected and labelled crossings are outside the 30-minute window. This is counted as a false positive and a false negative, technically reducing reported model performance, even though the model is likely correct in this case.

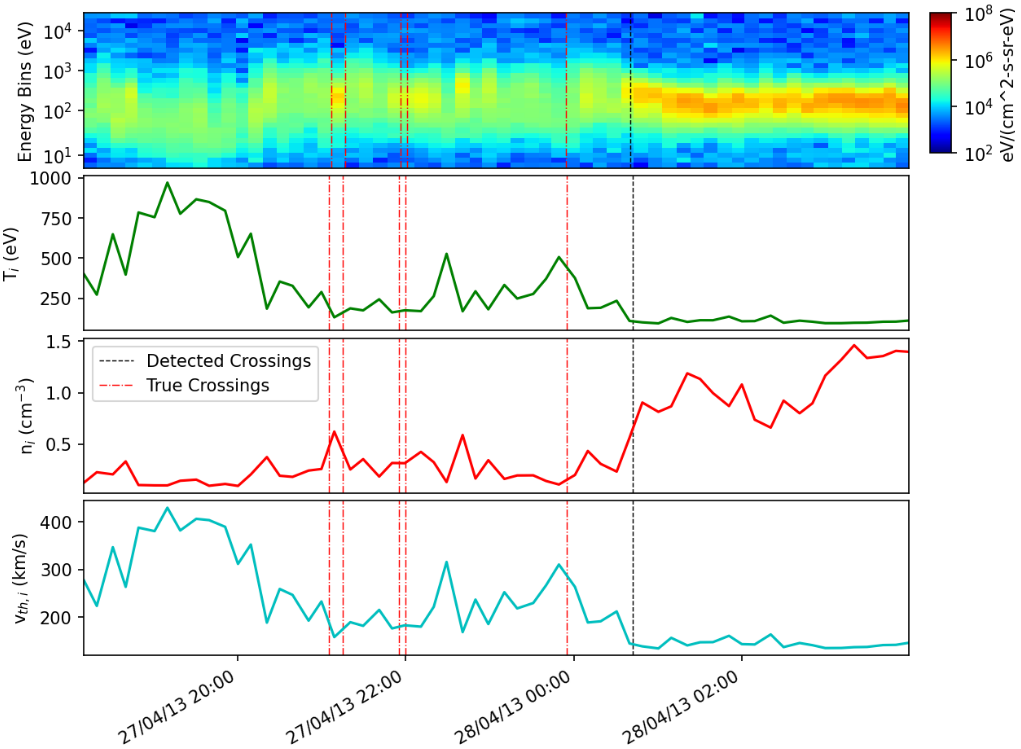


Figure 7.6: Close-up of ARTEMIS P1 full pass showing features and detected crossings. The top plot shows the spectrogram, the following plots show average ion temperature, density and velocity in that order. Detected crossings are shown as vertical dashed black lines, and labelled crossings as vertical dashed dotted red lines.

Something similar happens within this snippet towards the final magnetotail exit at 28/04/13 00:00. The zoom in can be seen in Figure 7.6. Here, a significant amount of crossings are not detected, and the final crossing is detected more than half an hour later than the labelled crossing. In this figure, it is important to note the difference in value scale compared to Figure 7.5. The ion density, velocity and temperature are all significantly lower. Many labelled crossings occur at comparatively small changes compared to the other zoomed-in snippet. The model needs more pronounced changes to detect a crossing, and thus misses many of these crossings. However, across both zoomed-in snippets, the model seems consistent with what plasma property values constitute a crossing, showing that the model is robust in this regard. This is slightly less so in the labelled dataset. One very important note to make is that the labelled dataset was created at a higher sampling rate for scalar data, and sometimes spectrogram data as well. This means some changes might not be as pronounced in the 10-minute sampled data, leading to missed crossings or crossings occurring at different times. The used raw data (before model pre-processing steps) were always averaged across the samples within each 10-minute window, which can lead to some changes being smoothed out. This is especially true for entry and exit crossings closely spaced in time. A lot of entry and exit crossings are labelled within 5-10 minutes of each other, and are not detected. 2 instances can be seen in Figure 7.6 at around 21:00 and 22:00, likely caused by the resampling effect, and pre-processing averaging. These trends were also observed in other magnetotail passes and with the other MAE models.

7.2.1. Model Limitations

Several limitations have been identified in the model performance and evaluation process:

Labelled Data Quality: The labelled dataset used for training can contain inaccuracies due to manual identification of crossings. Some labelled crossings may not perfectly align with the actual crossing times, making the evaluation of model performance less precise.

Sensitivity Requirements: The models require pronounced changes in plasma properties to detect boundary crossings. This means they may miss more subtle in-and-out crossings, or identify crossings later or earlier than they actually occur.

Temporal Resolution Challenges: Due to the 10-minute sampling resolution of the input data, an inherent temporal accuracy limitation exists. Specifically, closely spaced crossings within 5-10 minutes are very difficult to detect.

Data Processing Trade-offs: Pre-processing averaging helps to reduce the amount of false positives and increase accuracy, but similarly to the temporal resolution limits, it can cause closely spaced crossings to be missed, and smooth out some crossings.

Coverage Limitations: The lower recall rates needed to achieve high precision indicate that the models do not capture all magnetopause crossings. This trade-off between precision and recall means that models may limit the completeness of the crossing dataset.

Data Availability: The models require consistent data coverage across all the variables used. If gaps exist in some variables, no crossings can be detected during these periods.

7.3. Magnetopause Boundary Crossing Spatial Distribution

The following spatial validation analyses are not performed using the MAE model presented in subsection 7.1.2, but rather the high F0.5* score model using ion thermal velocity, taken from Table 6.5. Due to the larger quantity of false positives in the high F0.5* score model, outliers are easier to spot and investigate. This gives more insight into the model behaviour. The spatial distribution is shown later with the MAE model presented in subsection 7.1.2 for comparison.

7.3.1. High F0.5* Score Model Spatial Validation

To validate the physical plausibility of the detected magnetopause boundary crossings, the spatial distribution of these crossings is analysed and compared to the labelled crossings during the same time period. For this, the high F0.5 score model using thermal velocity is used. Mostly because it was quicker to assign crossings, and some interesting findings came out of this. The locations of the detected crossings are additionally compared against established models of the magnetopause boundary, such as the Shue et al. (1998, [70]) model, which provides a theoretical framework for the expected shape and position of the magnetopause based on solar wind conditions like dynamic pressure and the Z-axis component of the magnetic field strength.

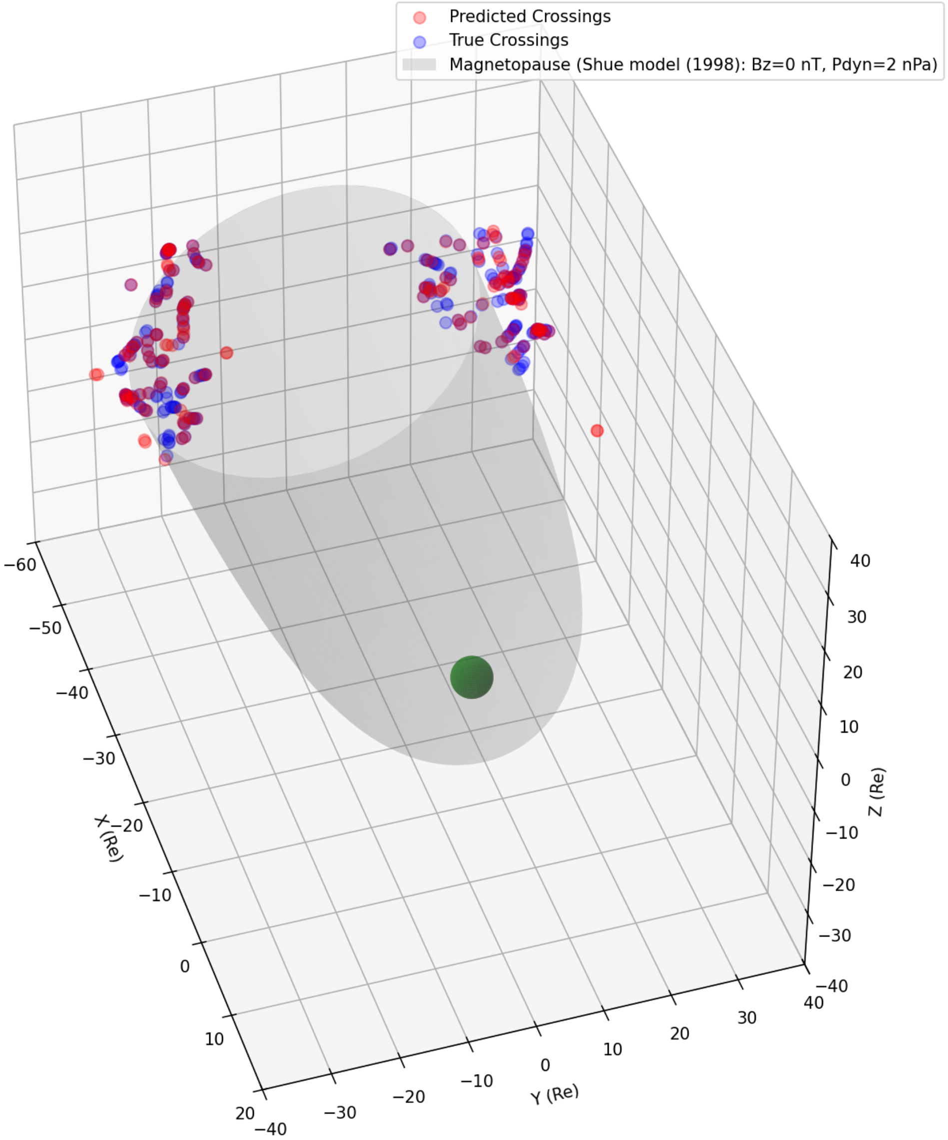


Figure 7.7: 3 Dimensional plot showing the spatial distribution of detected magnetopause crossings (red points) and labelled crossings (blue points) in relation to the Shue et al. (1998, [70]) magnetopause model (grey surface). The Earth is represented by the green sphere at the centre. The plot is in the Geocentric Solar Magnetospheric (GSM) coordinate system, with the Sun located along the positive X-axis.

Figure 7.7 shows the 3-dimensional spatial distribution of detected and labelled magnetopause crossings in relation to the Shue et al. (1998) magnetopause model. The detected crossings (red points) are generally clustered around the theoretical magnetopause boundary (grey surface), indicating that the model is effectively identifying crossings in physically plausible locations. However, some detected crossings are located further away from the model boundary, to the right of the figure. These will be investigated further in subsection 7.3.3.

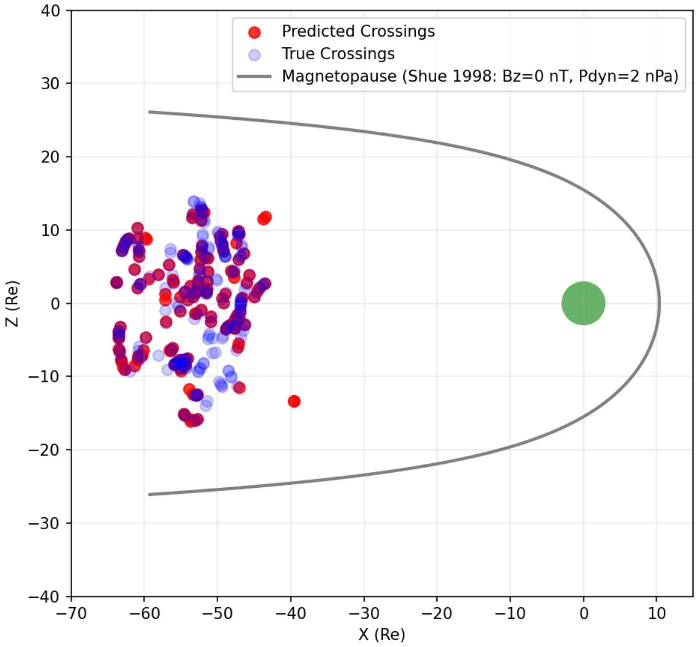


Figure 7.8: 2 Dimensional plot showing the spatial distribution in the X-Z plane of detected magnetopause crossings (red points) and labelled crossings (blue points) in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The Earth is represented by the green circle at the centre. The plot is in the Geocentric Solar Ecliptic (GSM) coordinate system, with the Sun located along the positive X-axis.

Both Figure 7.8 and Figure 7.9 show 2-dimensional projections of the spatial distribution of detected and labelled magnetopause crossings in relation to the Shue et al. (1998) magnetopause model. Again, the detected crossings (red points) are generally clustered around the theoretical magnetopause boundary (black line). The outliers can be easily spotted here, specifically apparent in the X-Y plane at near 40 and -40 R_E on the Y-axis. Both 2D plots show how most detected crossings are located very close to labelled crossings, often coinciding with them. Some detected crossings near -10 R_E on the Y-axis are also quite far away from labelled crossings, but still in plausible Y-axis coordinates. Further investigation is performed in subsection 7.3.3 to find out why these false positives/outliers occur.

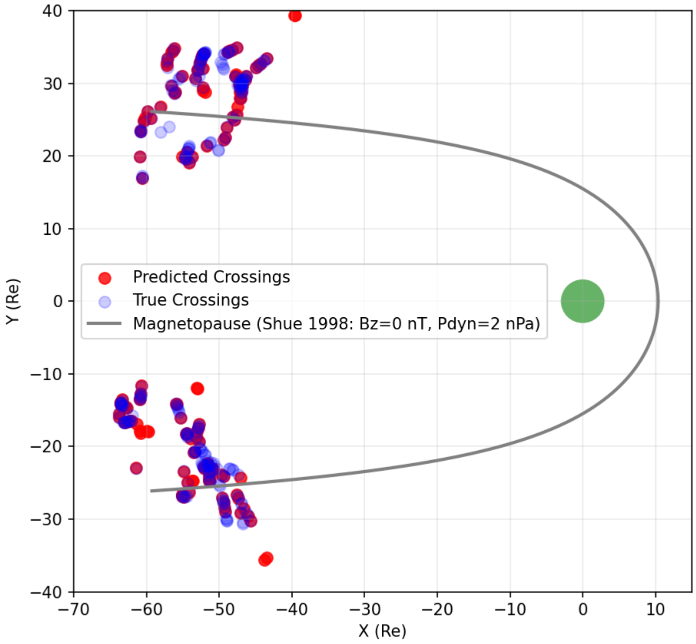


Figure 7.9: 2 Dimensional plot showing the spatial distribution in the X-Y plane of detected magnetopause crossings (red points) and labelled crossings (blue points) in relation to the Shue et al. (1998) magnetopause model (black line). The Earth is represented by the green circle at the centre. The plot is in the Geocentric Solar Ecliptic (GSM) coordinate system, with the Sun located along the positive X-axis.

7.3.2. MAE Model Spatial Validation

The spatial distribution of the main MAE model (using ion velocity and the X-component of the magnetic field) is very similar to the High F0.5* score model, with most detected crossings being located close to the labelled crossings and the Shue et al. (1998) magnetopause model. The results can be seen in Figure 7.10 and Figure 7.11. However, the extreme outliers seen in Figure 7.9 are not present here.

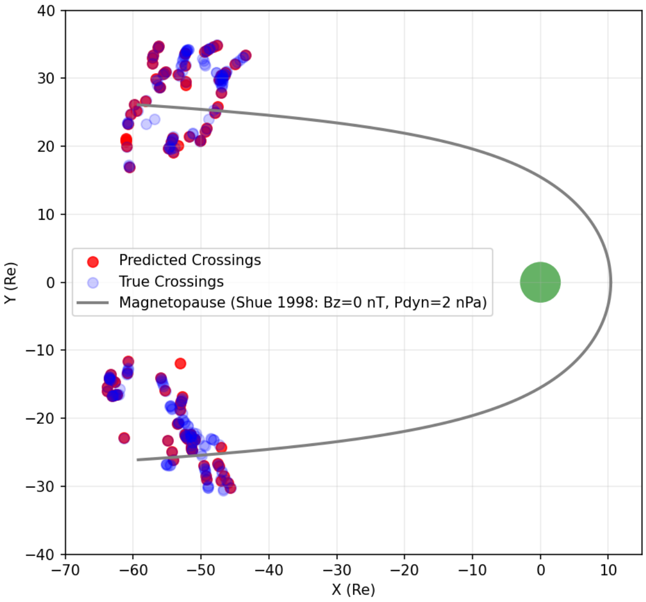


Figure 7.10: 2 Dimensional plot showing the spatial distribution of detected magnetopause crossings (red points) and labelled crossings (blue points) in relation to the Shue et al. (1998) magnetopause model (grey surface) on the X-Y plane. The Earth is represented by the green sphere at the centre. The plot is in the Geocentric Solar Ecliptic (GSM) coordinate system, with the Sun located along the positive X-axis.

When comparing Figure 7.11 to Figure 7.8, the spatial distribution is very similar, and the detected crossings of one model tend to coincide with the other.

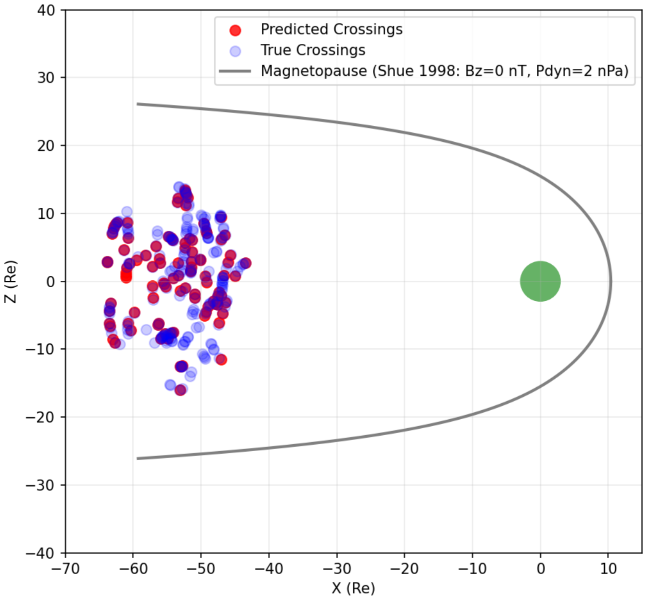


Figure 7.11: 2 Dimensional plot showing the spatial distribution of detected magnetopause crossings (red points) and labelled crossings (blue points) in relation to the Shue et al. (1998) magnetopause model (grey surface) on the X-Z plane. The Earth is represented by the green sphere at the centre. The plot is in the Geocentric Solar Ecliptic (GSM) coordinate system, with the Sun located along the positive X-axis.

Overall, the spatial validation shows that both models are effective at detecting magnetopause crossings in physically plausible locations, generally aligning well with established magnetopause models and labelled crossing data. When using the additional inhibition methods, a model aimed at high precision, extreme outliers are largely avoided compared to the high F0.5* score model.

7.3.3. Outlier Analysis

The outliers seen in the High F0.5* score model are used here, since they show the most extreme outliers. These false positives offer a chance to investigate what plasma conditions can lead to false detections, and what these events with similar plasma conditions are.

The first outliers to be discussed are the ones seen in Figure 7.9, at around -35 R_e on the Y-axis. These ”crossings” are detected at 26/06/2013 8:10 and 26/06/2013 8:25, very close to each other. Spectrogram, ion properties and GSM magnetic field data can be seen in Figure 7.12. In Figure 7.12, typical signatures of a magnetopause crossing are present in all used variables for the model. The magnetic field properties do show some special behaviour. The B_x component shows an initial decrease followed by a sharp increase, to settle back down. The B_y component shows this same pattern, just slightly shifted in time. Since the crossing occurs far from other crossings during this pass, and from the expected magnetopause location, other causes should be considered.

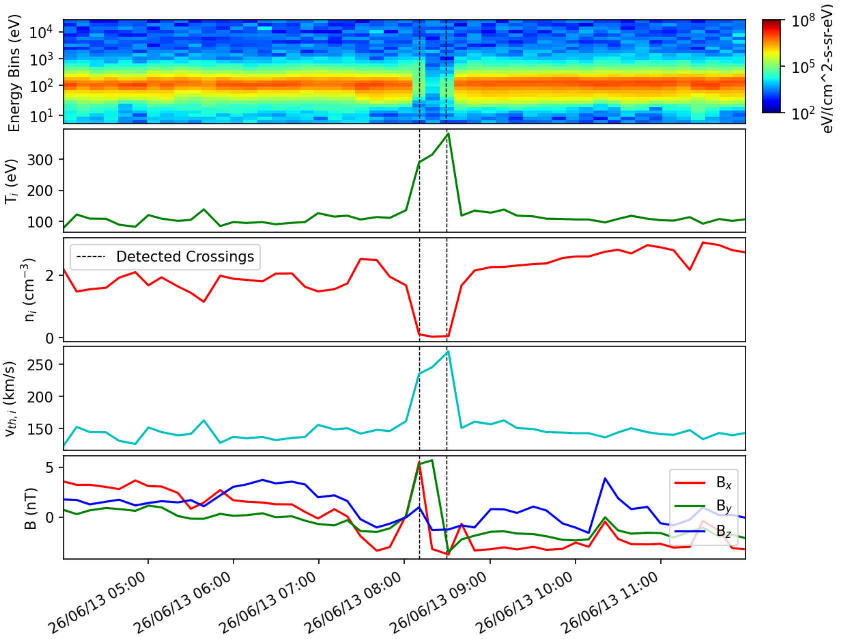


Figure 7.12: ARTEMIS P1 data showing features around the outlier crossings at 26/06/2013 8:10 and 26/06/2013 8:25. The top plot shows the spectrogram, the following plots show average ion temperature, density and velocity in that order. The final plot shows the magnetic field in GSM coordinates (B_x, B_y, B_z). Detected crossings are shown as vertical dashed black lines.

A more detailed view of the ion density and magnetic field can be seen in Figure 7.13, where data at a higher resolution is used. Both B_x and B_y show a bipolar signature, fluctuating in a sinusoidal manner, while B_z stays mostly constant. The magnetic field magnitude also shows a clear increase between the detected ”crossings”. During this, a strong decrease in ion density is seen as well. These sudden changes are often seen in ARTEMIS data when the spacecraft crosses the lunar wake [74]. The lunar wake is a plasma void created on the nightside of the Moon as it obstructs the solar wind flow. As ARTEMIS crosses into and out of this wake region, it encounters abrupt changes in plasma density and magnetic field characteristics, which can mimic magnetopause crossing signatures. This explains why the model detected these outliers as magnetopause crossings, despite their location being far from the expected magnetopause boundary.

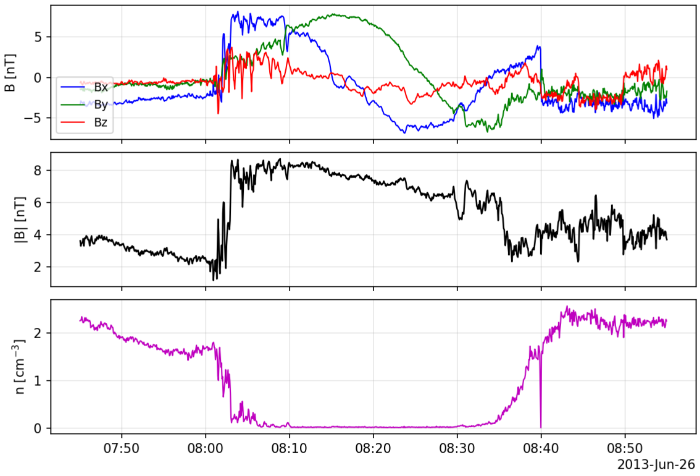


Figure 7.13: Close-up of ARTEMIS P1 data showing ion density and magnetic field around the outlier crossings at 26/06/2013 8:10 and 26/06/2013 8:25 at high resolution. The top plot shows the magnetic field in GSM coordinates (B_x, B_y, B_z), the middle plot shows magnetic field magnitude, and the bottom shows average ion density.

Nevertheless, the magnetic field properties in this wake seem interesting. A maximum variance analysis (MVA) was performed on the magnetic field data around the crossing to determine the 'shape' of the magnetic structure. The MVA analysis changes the coordinate system of the magnetic field data to a new system defined by the maximum, intermediate and minimum variance directions of the magnetic field. The minimum variance direction is often interpreted as the normal direction of a current sheet or boundary layer. The MVA results can be seen in Figure 7.15 and Figure 7.14.

The B_L and B_M components in Figure 7.14 show the magnetic field components in the maximum and intermediate variance directions, respectively. Around the time of the crossing, shown by the red dots in Figure 7.14, a strong rotation in the B_L and B_M components is seen.

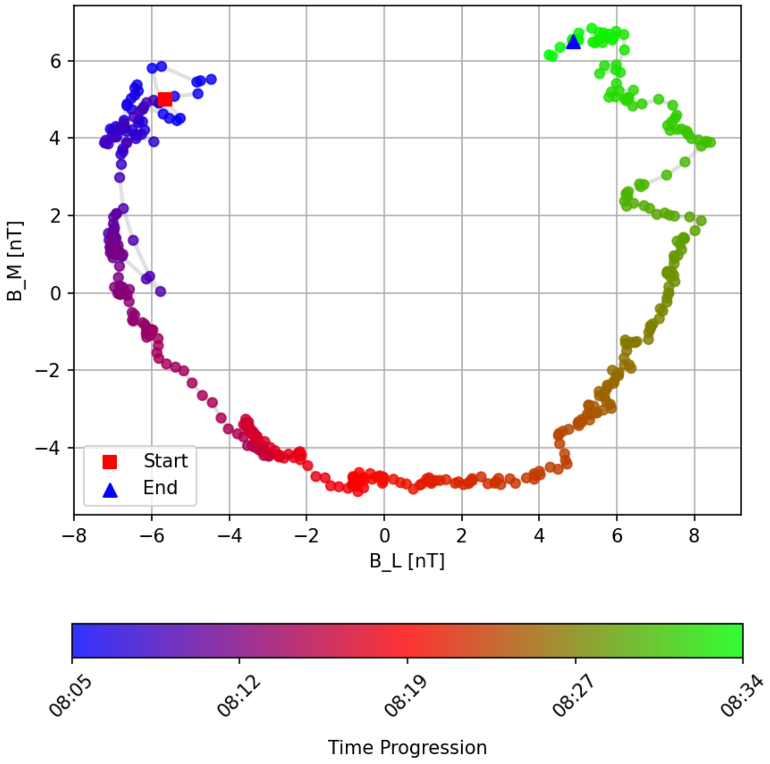


Figure 7.14: B_L and B_M components from MVA analysis for the outlier crossing at 26/06/2013 8:10 and 26/06/2013 8:25. A colorscale is used to show the time progression.

The B_n component in Figure 7.15 shows the magnetic field component in this minimum variance direction. It stays mostly constant around zero, fluctuating slightly. The fluctuation is stronger at the start, around 8:05, where some stronger instability in the magnetic field components is observed. Afterwards, the B_n stabilises close to zero, ticking -1 to 1 nT.

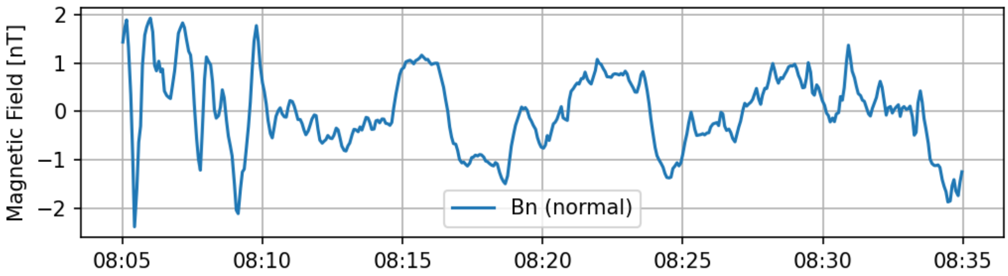


Figure 7.15: B_n component from MVA analysis for the outlier crossing at 26/06/2013 8:10 and 26/06/2013 8:25.

To prevent lunar wake crossings from being detected as magnetopause crossings, a post-processing step is added to check spacecraft coordinates against the Moon's position in the HEE (heliocentric Earth ecliptic) coordinate system with respect to the solar wind. If the spacecraft is in the lunar wake region, here defined by being behind the moon from the solar wind's perspective with a 10% margin added to the lunar radius, detected crossings are discarded.

Another outlier within the validation set is within the magnetotail at 0 R_e on the Z-axis, and around -10 R_e on the Y-axis. Similarly to the previous outlier, two crossings are detected closely in time, at 26/05/2013 01:09 and 26/05/2013 01:19. These can be seen in Figure 7.16. Again, typical signatures of a magnetopause crossing are present in all the variables used for the model.

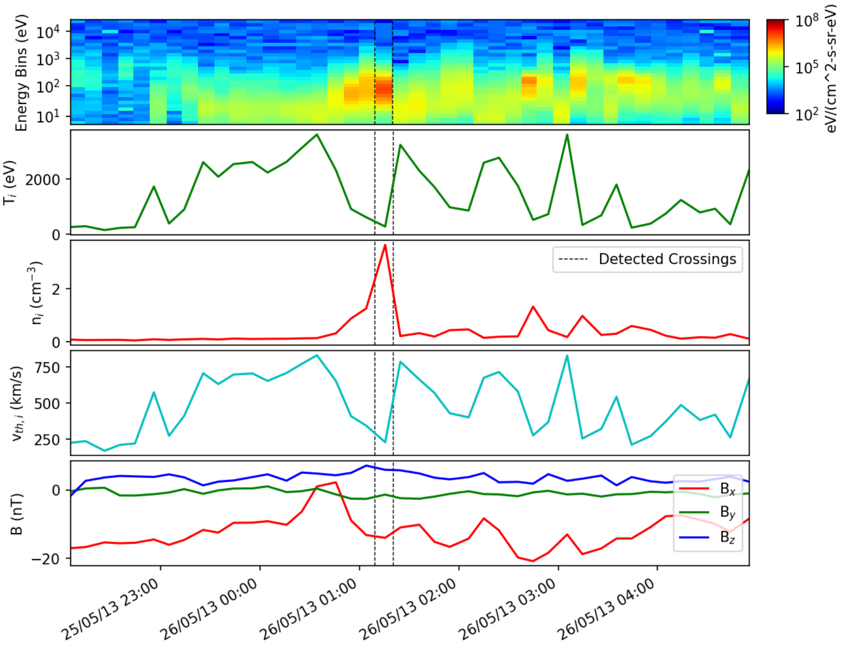


Figure 7.16: ARTEMIS P1 data showing features around the outlier crossings at 26/05/2013 01:09 and 26/05/2013 01:19. The top plot shows the spectrogram, the following plots show average ion temperature, density and velocity in that order. The final plot shows the magnetic field in GSM coordinates (B_x, B_y, B_z). Detected crossings are shown as vertical dashed black lines.

To see whether these detected crossings are indeed actual crossings, a more detailed view of the ion density, magnetic field and ion velocity is taken in Figure 7.17. Most parameters show typical signatures of a magnetopause crossing, higher ion density, lower ion temperature and velocity. However, ion velocities in the negative X direction, while also a sign of magnetosheath plasma, can be caused by another phenomenon within the magnetotail.

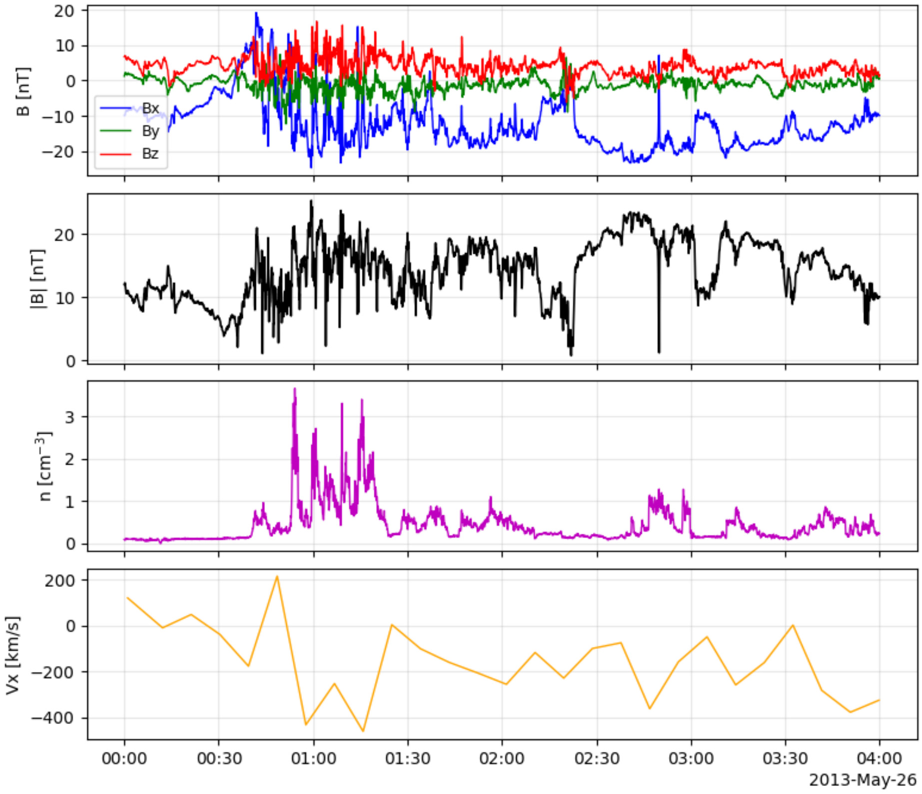


Figure 7.17: ARTEMIS P1 data showing features around the outlier crossings at 26/05/2013 01:09 and 26/05/2013 01:19. The top plot shows magnetic field in GSM coordinates (B_x, B_y, B_z), with the one below showing magnetic field magnitude. The middle plot shows average ion density. The second-to-last plot shows ion temperature, and the bottom plot shows ion velocity in the X direction (no higher sampling rate available for this time range).

Since the detected crossing is close to $0\,R_E$ on the Z-axis, it will likely be close to the plasma and current sheet. Bursty bulk flows (BBFs) are fast, transient flows within the magnetotail plasma sheet, directed earthward or tailward along the X-axis [51]. They are often associated with magnetic reconnection events in the magnetotail and can lead to significant changes in plasma properties, including ion velocity, density, and temperature [43]. Typical indicators of BBFs are a high velocity along the X-axis, which is seen from 1:00 to 1:30, making it a tailward BBF candidate, an increase in density, also seen in this timeframe, and more unstable magnetic fields [62]. However, care should be taken, as the magnetosheath also commonly has high velocities on the X-axis, with a higher density. A key discriminator between a magnetopause crossing and a BBF is the magnetic field behaviour. In a magnetopause crossing, a significant change in the magnetic field direction and magnitude is expected; more specifically, the X-axis component is very pronounced within the magnetotail. As can be seen in Figure 7.17, the magnetic field still shows a strong B_x component during a very unstable magnetic field. Notably, just before the highest density increases, a small bump in density accompanied by a positive V_x might indicate an occurring earthward BBF. This is plausible as discussed by Chong et al. (2021, [14]). This indicates that the detected crossings are likely not magnetopause crossings, but rather detected bursty bulk flows within the magnetotail plasma sheet. Furthermore, geomagnetic storm activity during this time period was high, with geomagnetic storms occuring on 25/05/2013 [56].

7.4. Detection of Non-Labelled Time Range Crossings

Now that the model has been validated against labelled data, it can be used to detect magnetopause crossings in unlabelled data. For this, the main MAE model is used. The model is applied to all ARTEMIS P1 and P2 data from 2013-07 until 2025-09, and the spatial distribution of detected crossings is analysed. In total, 3072 crossings were detected by the model, 1476 for ARTEMIS P1, and 1596 for ARTEMIS P2. The results can be seen in Figure 7.18 and Figure 7.19. Here, the crossings are binned into a 2D histogram with a bin size of 2 R_E in both directions, and the colour indicates the number of crossings detected within each bin. The Shue et al. (1998, [70]) magnetopause model is again shown for reference.

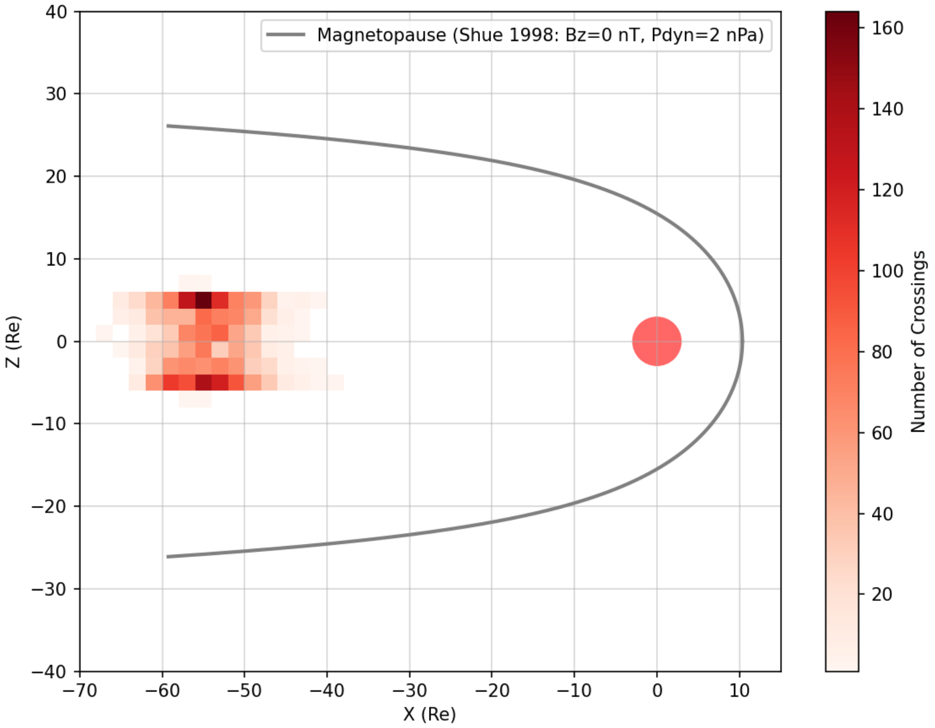


Figure 7.18: 2 Dimensional histogram showing the spatial distribution in the X-Z plane of detected magnetopause crossings from 2013-07 to 2025-08 in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The plot is in the Geocentric Solar Ecliptic (GSE) coordinate system.

An immediate observation one can make from Figure 7.18 and Figure 7.19 is that the density of crossigns is very high close to the expected magnetopause location, and decreases further away from it. Additionally, a shift towards the positive Y-axis is seen, which is expected due to the aberration effect caused by Earth's orbital motion around the Sun. Some outliers are also present further from the expected magnetopause location in small amounts.

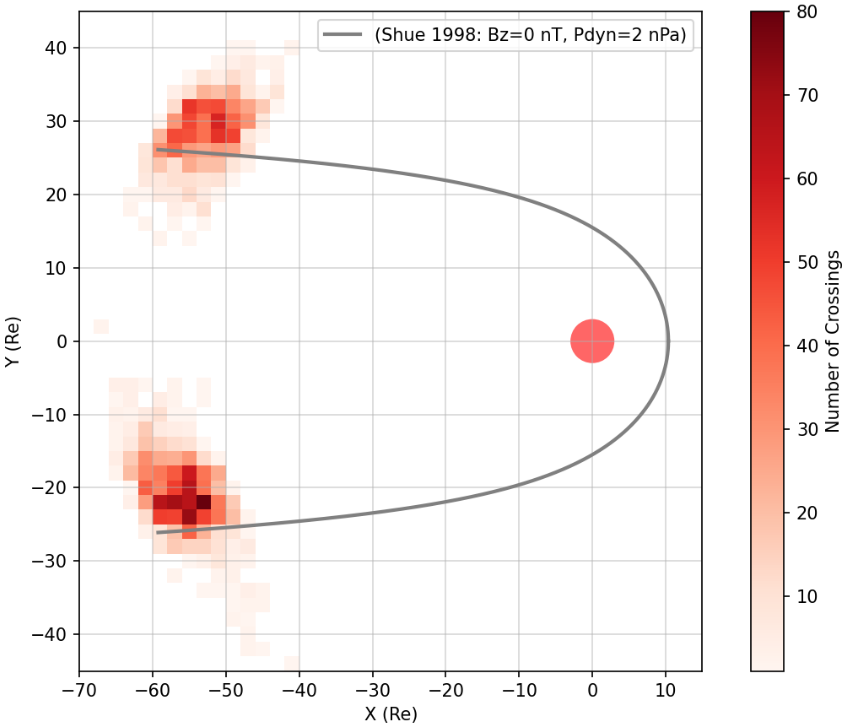


Figure 7.19: 2 Dimensional histogram showing the spatial distribution in the X-Y plane of detected magnetopause crossings from 2013-07 to 2025-08 in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The plot is in the Geocentric Solar Ecliptic (GSE) coordinate system.

To correct for both Earth's orbital motion around the Sun, the solar wind aberration, and Earth's dipole direction, a correction to the aberrated Solar Wind corrected Geocentric Solar Magnetospheric (aSWGSM) coordinate system is performed. The solar wind data was taken from the OMNI database [53]. The correction was performed on each crossing separately. First, the solar wind velocity vector from 15 minutes before the crossing is taken. Depending on that velocity, the time delay caused by solar wind travel time from the bow shock to the crossings' location is calculated. Using that time, data from the OMNI database is averaged from an hour before the crossing to the calculated time. This average solar wind velocity vector is then used to calculate the aberration angle, which is then used to rotate the crossing's coordinates into the aSWGSM coordinate system. The coordinate transformation is described in Appendix C.

The spatial distribution can be seen in Figure 7.20. Here, the shift towards the positive Y-axis is no longer present, and the distribution is more symmetric around the expected magnetopause location. The bulk of crossings is more densely packed around the expected magnetopause location, with fewer outliers further away from it.

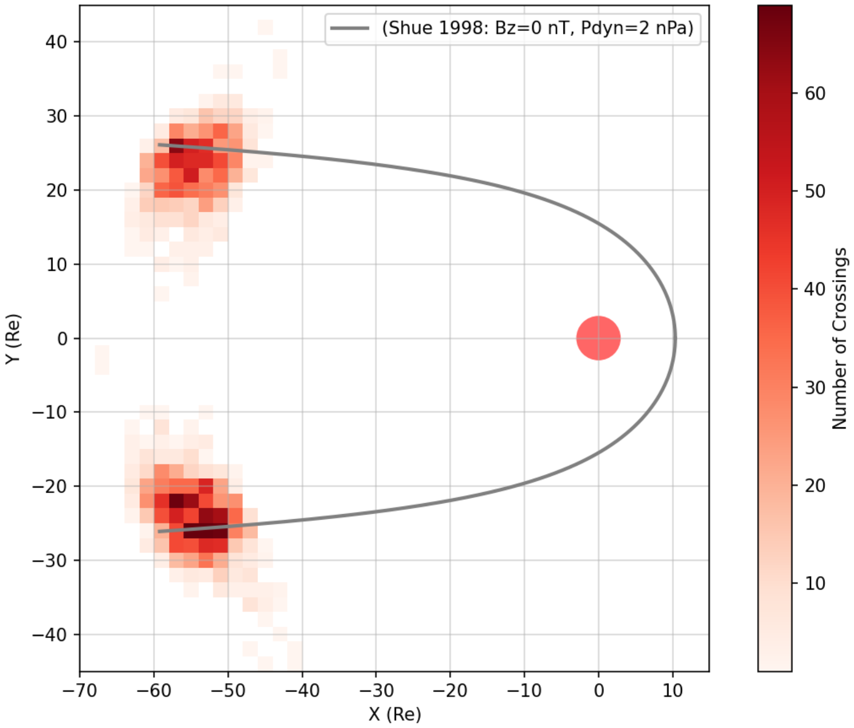


Figure 7.20: 2 Dimensional histogram showing the spatial distribution in the X-Y plane of detected magnetopause crossings from 2013-07 to 2025-08 in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The plot is in the aberration Solar Wind corrected Geocentric Solar Magnetospheric (aSWGSM) coordinate system.

After the removal of outliers, as described in section 7.5, the spatial distribution can be seen in Figure 7.21 and Figure 7.22. Here, the outliers further from the expected magnetopause location are significantly reduced.

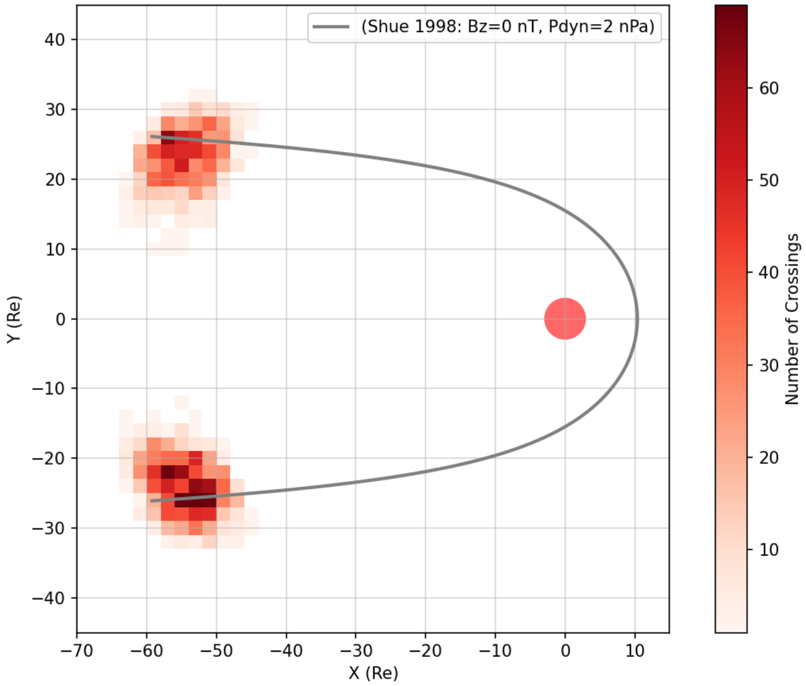


Figure 7.21: 2 Dimensional histogram showing the spatial distribution in the X-Y plane of detected magnetopause crossings from 2013-07 to 2025-08 in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The plot is in the aberration Solar Wind corrected Geocentric Solar Magnetospheric (aSWGSM) coordinate system. Outliers are removed.

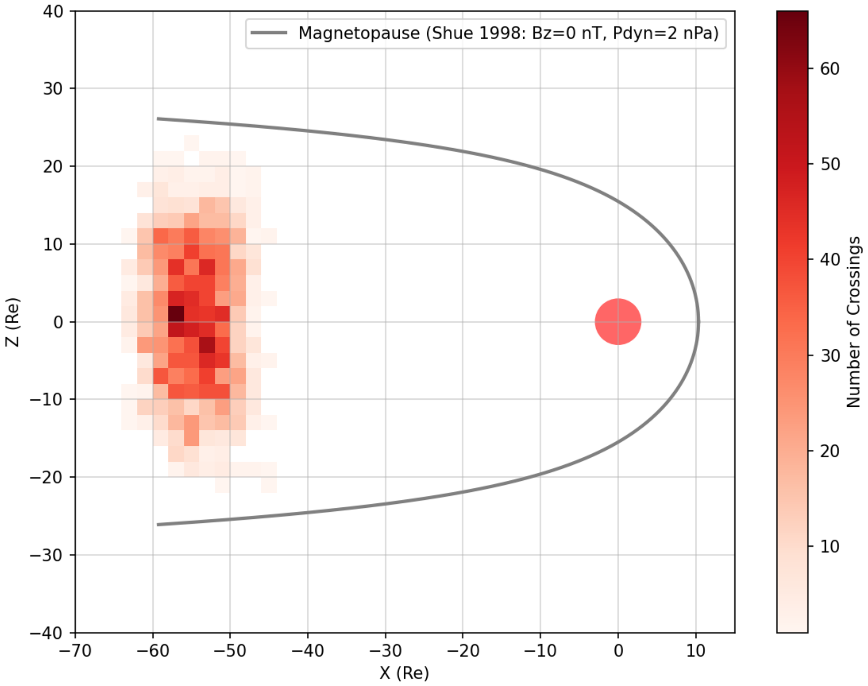


Figure 7.22: 2 Dimensional histogram showing the spatial distribution in the X-Z plane of detected magnetopause crossings from 2013-07 to 2025-08 in relation to the Shue et al. (1998, [70]) magnetopause model (black line). The plot is in the aberration Solar Wind corrected Geocentric Solar Magnetospheric (aSWGSM) coordinate system. Outliers are removed.

From Figure 7.22, it can also be observed that the distribution of detected crossings is significantly more spread out along the Z-axis in the aSWGSM coordinate system, compared to the uncorrected GSE coordinate system in Figure 7.18. Earth's dipole tilt causes spreading out as it can rotate the GSE coordinate system about the X-axis by up to 33 degrees, depending on the time of year and day Kubyshkina et al. (2015, [33]).

7.4.1. Spatial Distribution Over Time and Solar Cycle

The spatial distribution can also be plotted over time to see if any general trends can be observed. This is done for the main MAE transformer model. The distance from the aSWGSM X-axis to each detected crossing is calculated, and the yearly average distance is plotted over time. The distribution does not contain outliers as they were previously identified and thus uses the crossing distribution seen in Figure 7.22. Solar cycle 24's maximum occurred in April 2014, and the subsequent minimum in December 2019 [72]. Figure 7.23, seems to suggest that the magnetopause at lunar-distance is anti-correlated with the solar cycle, with the magnetopause at lunar-distance being further away from Earth during solar minimum, and closer during solar maximum.

Figure 7.23 also shows the predicted magnetopause standoff distance from the Shue et al. (1998, [70]) model, using average solar wind dynamic pressure and IMF B_z for each year, taken from the OMNI database [52]. The model was used at a distance of -54 R_e along the aSWGSM X-axis, which corresponds with the average distance of magnetopause crossings along this axis from Earth. The predicted magnetopause standoff distance is also calculated based on the solar wind dynamic pressure, from an hour before to the moment of each detected crossing, also taken from the OMNI database [53]. This resulted in a better match to the detected crossings.

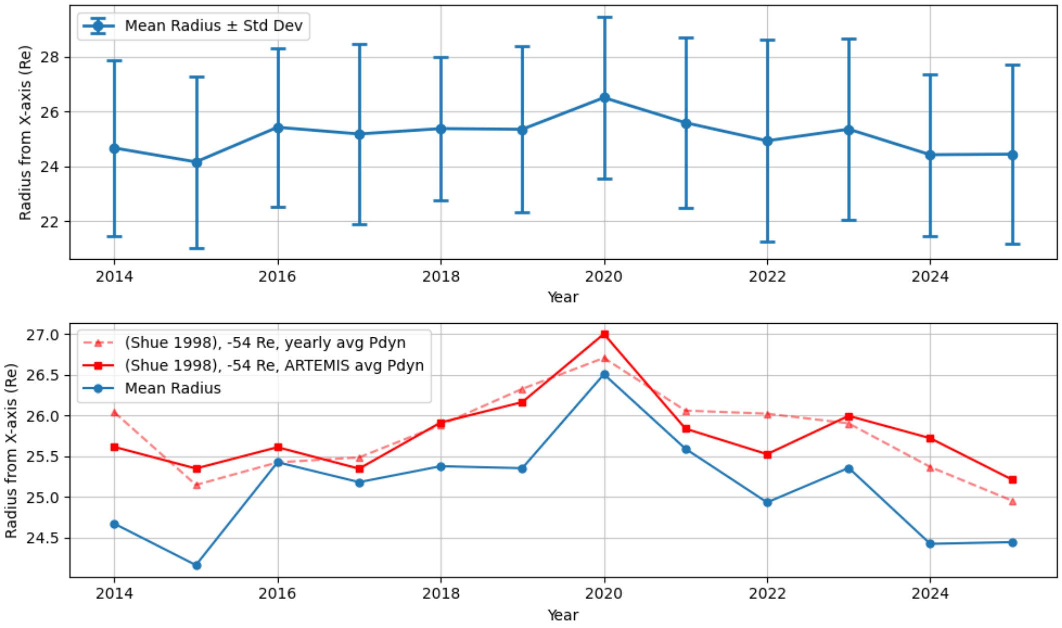


Figure 7.23: The yearly average radius from the aSWGSM X-axis to the detected magnetopause crossings from 2013-07 to 2025-08. The top plot shows the radius with standard deviation. The bottom plot shows the same data without standard deviation, and compared to the yearly average predicted magnetopause standoff distance from the Shue et al. (1998, [70]) model based on average solar wind dynamic pressure (overall yearly and ARTEMIS average) and IMF B_z for each year.

Table 7.4 shows the yearly statistics of the detected crossings used in Figure 7.23, including mean radius from the aSWGSM X-axis, standard deviation and crossing count per year.

The yearly distribution for Figure 7.23 can be seen in Table 7.4.

Table 7.4: Yearly Statistics

Year	Mean Radius [R_e]	Std [R_e]	Crossing Count
2014	24.67	3.21	235
2015	24.16	3.13	299
2016	25.43	2.89	221
2017	25.18	3.29	287
2018	25.38	2.60	212
2019	25.35	3.02	249
2020	26.51	2.96	254
2021	25.59	3.09	261
2022	24.93	3.68	237
2023	25.36	3.30	222
2024	24.42	2.95	174
2025	24.44	3.28	150

Figure 7.23 matches the Shue et al. (1998, [70]) model in trend quite well, but the average radius is typically about half an Earth radius lower, and the solar minima seem to be more exaggerated. The standard deviation of the yearly average distance is also quite high, indicating a large spread in detected crossings. The magnetotail is a highly dynamic region, and the magnetopause location can vary significantly and quickly due to changes in solar wind conditions. Additionally, other events, like CMEs and geomagnetic storms, as will be discussed in section 7.5, can also influence the detected crossings, and thus the yearly average distance. This is partially mitigated in the plots seen here, as strong outliers have been removed.

7.5. Outlier Analysis of Detected Crossings

To find outliers of the analysis, the average distance in Earth radii to the X-axis of the aSWGSM coordinate system is calculated. Crossings are considered outliers when they are further than two standard deviations from the mean. The mean was calculated to be 25.22 R_e and the standard deviation 3.96 R_e across all years of data.

The outward outliers (outliers that are exceptionally far in the usual magnetosheath region) can be seen in Figure 7.24. The figure also shows the sunspot number across the years.

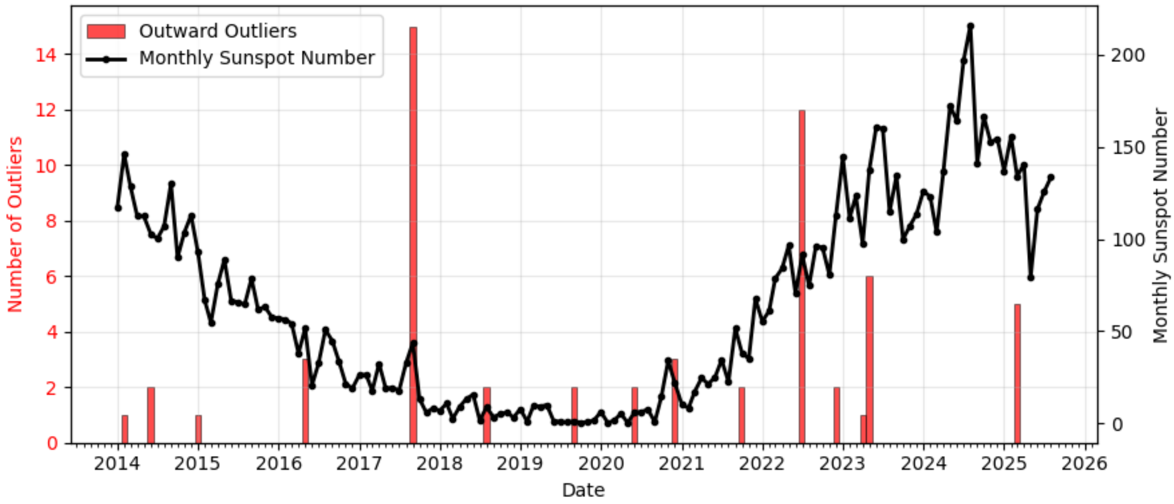


Figure 7.24: Outward outliers and sunspot number per month from 2013-08 until 2025-08.

Outward outliers are fairly rare, with mostly only a couple every year, with two exceptions. In 2017 and 2022, a large number of outliers were concentrated in a single month, 15 and 12, respectively. The outliers do not seem to follow a clear pattern compared to the solar cycle.

In 2017, a large number of outward outliers occurred around September 8th and 9th. This corresponds to the Coronal Mass Ejection hitting Earth's magnetic field caused by an X9 solar flare [56]. Figure 7.25 shows the data during that month.

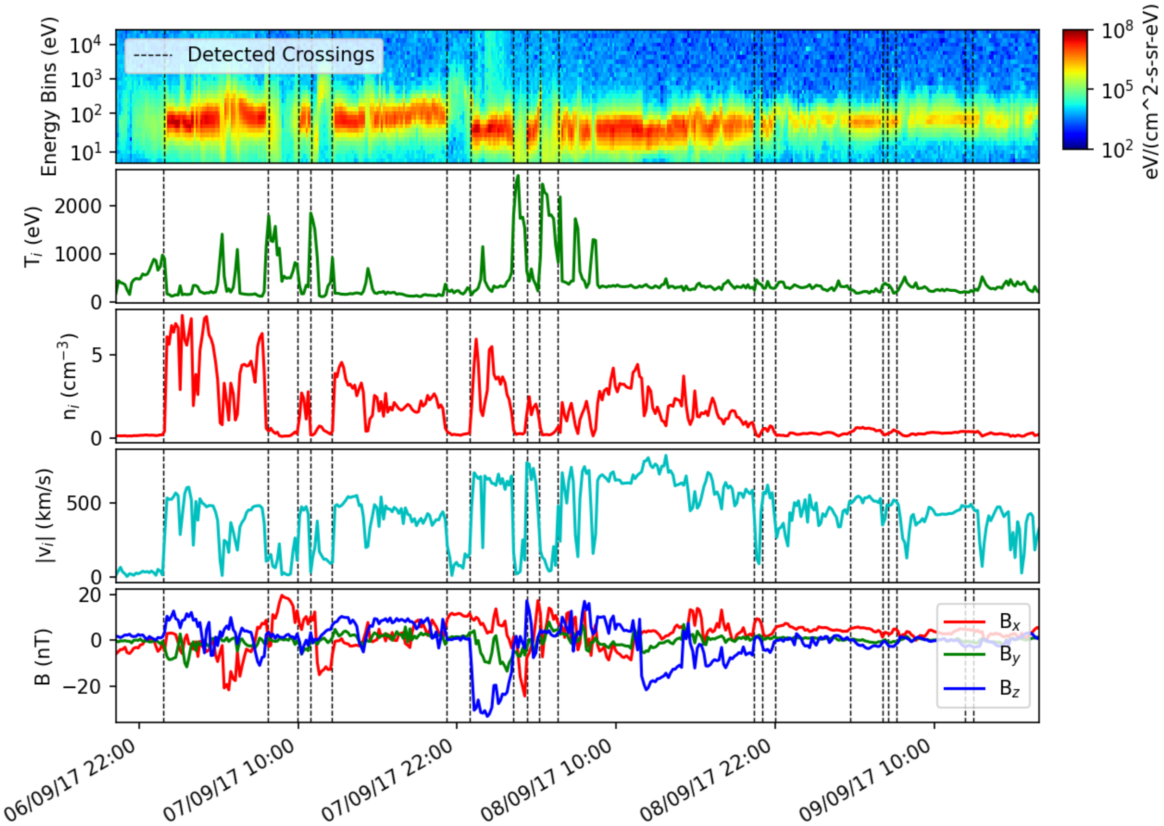


Figure 7.25: Outward outliers from September 2017.

Here, the spacecraft exits the magnetotail for the first time around midnight between the 6th and 7th of November. This first exit is detected due to a large increase in density, velocity and ion energy flux. Afterwards, the spacecraft enters and exits the magnetotail a few more times. Deep into the usual magnetosheath, at the end of the 8th of September, the ion density and energy flux levels drop significantly, way lower than usual levels in the magnetosheath. Furthermore, the magnetic field vector and the plasma moments data correspond with patterns of CMEs hitting as discussed in Werner et al. (2019, [79]). In Werner et al. (2019, [79]), the same drop in density is seen from the Wind spacecraft, attributed to the wake of the CME [80]. This significant drop in energy flux and density mimics plasma moment patterns seen in the magnetotail.

In 2022, most outward outliers occurred in July. Similar to what happened in 2017, the first initial exit is followed by a lot more entries and exits. During these exits and entries, proton density and energy flux stay relatively low, and proton velocity fluctuates heavily. The strong difference with usual magnetosheath values can be seen when comparing these values with data on the right-hand side of the plot.

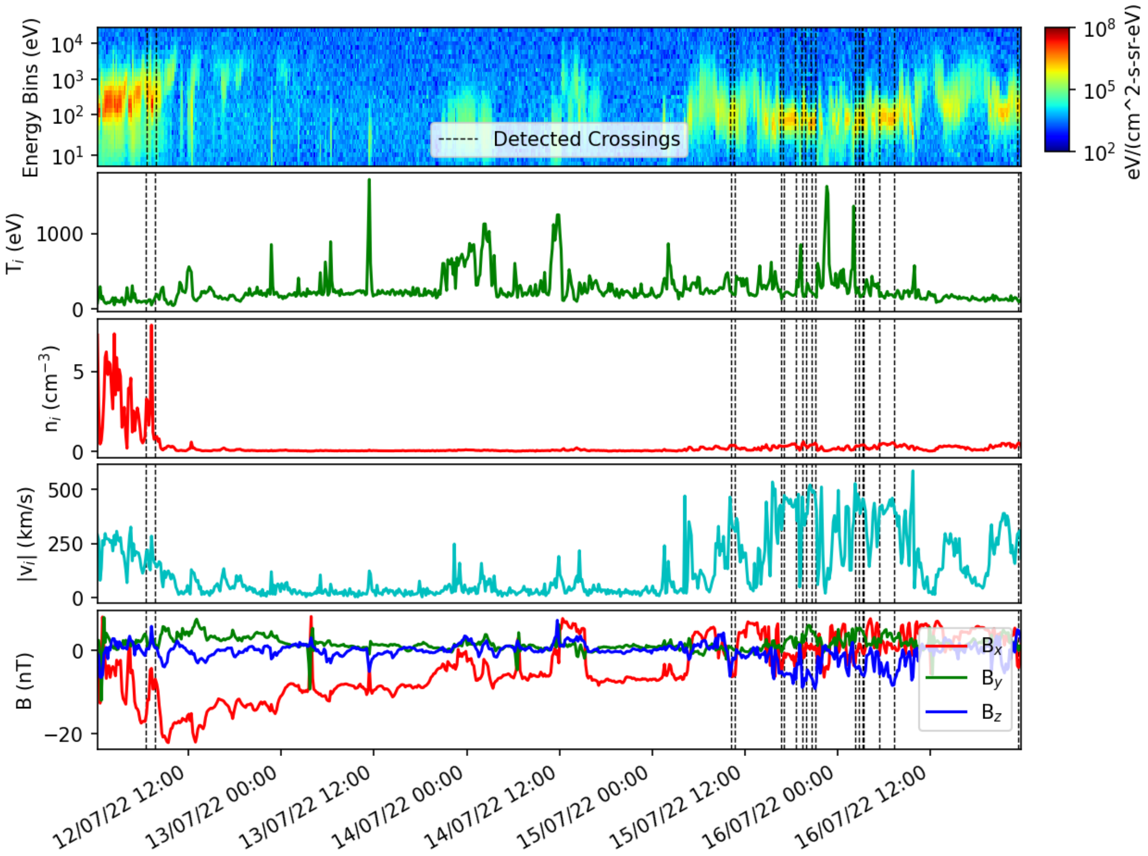


Figure 7.26: Outward outliers from July 2022.

These observations of deviating plasma moments from usual magnetosheath conditions could be explained by Figure 7.27. During the magnetotail pass from July 2022, the solar wind proton density heavily decreased, similarly to the wake of the CME from 2017, likely causing the false positives. This was also seen in data by the L1 Lagrange-point-based spacecrafts ([41, 50]).

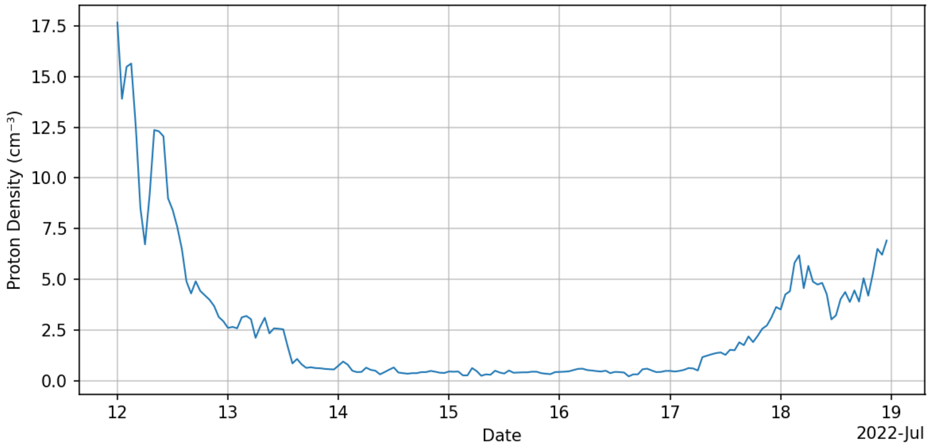


Figure 7.27: Proton density as measured by the Wind spacecraft from July 12th to July 19th 2022 [50].

The strong solar wind density decreases have previously been related to magnetosphere expansion, possibly explaining the extreme outlier detection [28].

The inward outliers (outliers that are exceptionally far in the usual magnetotail region) can be seen in Figure 7.28

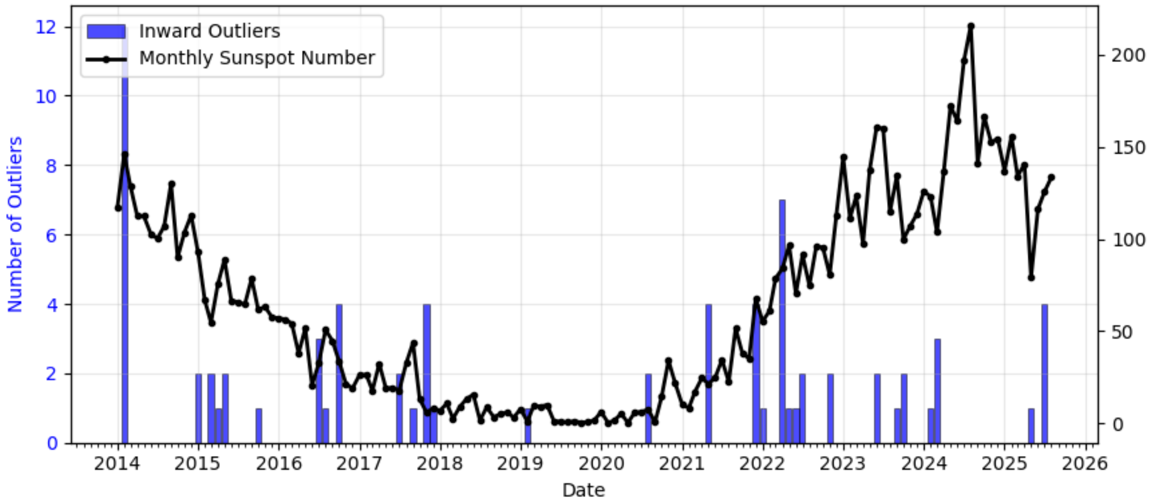


Figure 7.28: Outward outliers and sunspot number per month from 2013-08 until 2025-08.

These follow a path more aligned with the solar cycle. In the stretch of lowest solar cycle activity, from 2018 to 2021, only three inward outliers were detected. During the peaks of solar cycles 24 and 25, the amount per year is significantly higher. One standout inward outlier period is February 2014. Here, 12 inward outliers were detected. Most of these (9 out of 12) occurred on February 15th. This corresponds to a strong solar wind pressure increase caused by a CME hitting Earth's magnetic field at 13:00 UT [56]. High solar wind pressure is known to compress the magnetosphere, causing more inward magnetopause crossing detections. Additionally, strong CMEs can cause injections of high-density plasma into the plasma sheet [12]. This can cause false positive detections of magnetopause crossings, as the model puts high importance on ion density and ion energy flux.

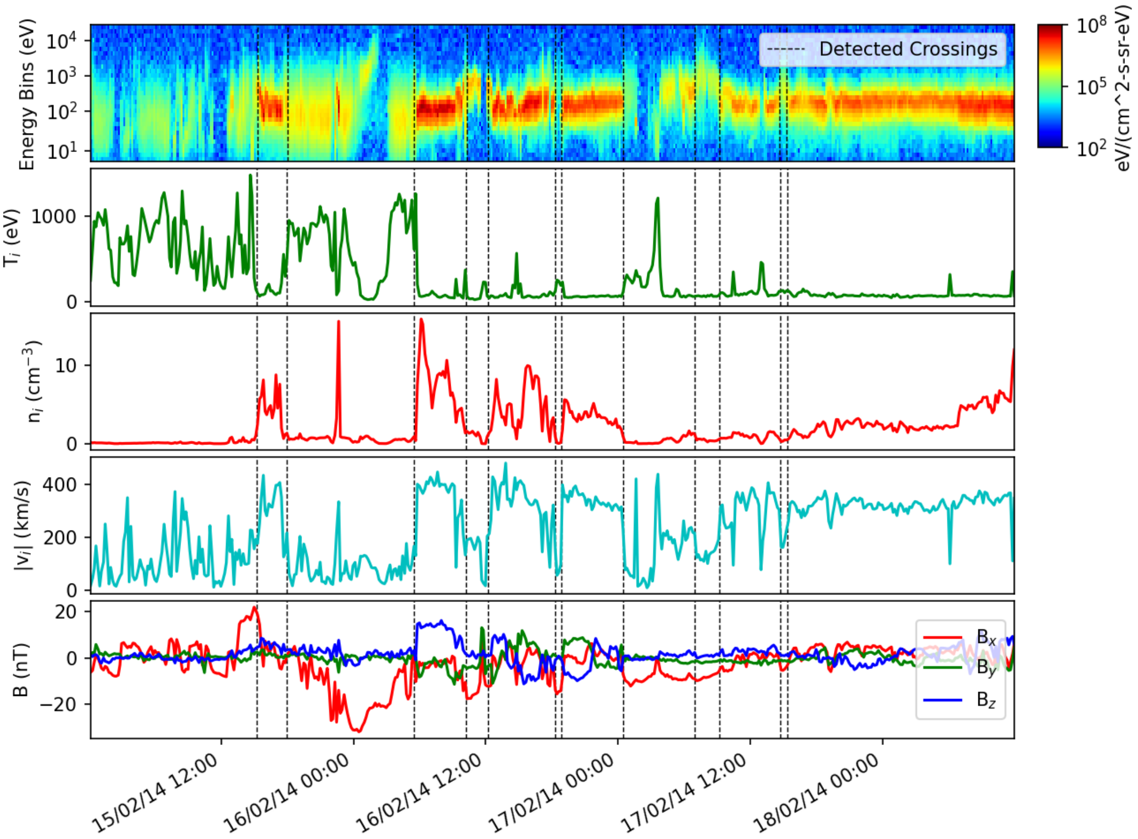


Figure 7.29: Inward outliers from February 2014 data.

Figure 7.29 shows the plasma moments and magnetic field data from this period. The first nine detected crossings were outliers, being detected at a radius from 10 to 15 R_e from the aSWGSM X-axis. During these crossings, ion density and ion energy flux are significantly higher than usual magnetotail values, mimicking magnetosheath plasma. The initial two crossings show a magnetic field reversal, from positive to negative B_x, indicating a possible crossing of the current sheet, and not a magnetopause crossing. However, the following crossings do not show this clear field reversal, and it is harder to attribute them to specific phenomena. Still, the elevated ion density and energy flux are likely caused by a combination of the compressed magnetosphere and injections of plasma into the plasma sheet caused by the CME.

The strongest inward outliers were detected on March 25th 2024. Here, 2 outliers at a radius of less than 5 R_e from the aSWGSM X-axis were detected. These correspond with a CME that struck Earth's magnetic field on March 24th 2024, opening a crack in Earth's magnetosphere, causing the strongest geomagnetic storm since September 2017 [56].

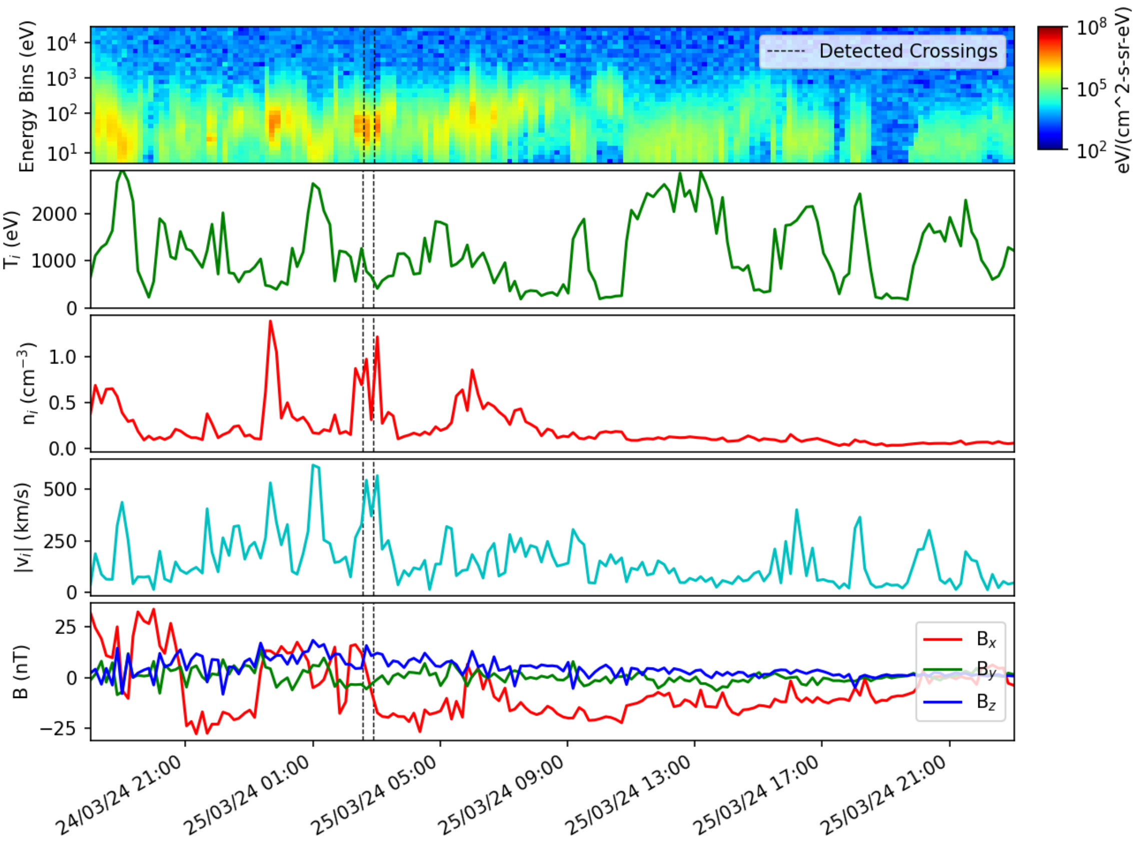


Figure 7.30: Inward outliers from March 2024 data.

At the two detected crossings' times, ion density and ion energy flux are again significantly higher than usual magnetotail values, mimicking magnetosheath plasma. Ion velocity also increases significantly, reaching values of up to 500 km/s, more typical for magnetosheath plasma. Some earlier spikes are also seen, but are likely not detected due to their shorter duration.

With most of the extreme outliers discussed, the overall distribution of distances from the aSWGSM X-axis for all detected crossings can be seen in Figure 7.31. Here, the logarithm of the solar wind dynamic pressure from the OMNI database is used as the X-axis [53]. A clear trend is seen, with higher solar wind dynamic pressures causing the magnetopause to be closer to Earth, and thus detected crossings being closer to the aSWGSM X-axis. Four events are also marked, corresponding to the strong CME events discussed earlier, showing their position in the distribution. The solar wind dynamic pressures are found by taking the average value of the hour before the detected crossings from the OMNI database. The OMNI database solar wind variables are time shifted to the bow shock nose, and the solar wind typically propagates to the magnetotail in around 20 minutes at a velocity of 400 km/s [17]. This makes the one-hour average a reasonable approximation.

The September 2017 CME crossings on the plot do not reflect accurate solar wind conditions, as they used daily averages for solar wind dynamic pressure, due to missing data at the times of the crossings, making them appear at much higher solar wind dynamic pressures than they actually were. Nevertheless, the other three events fit well within the overall trend.

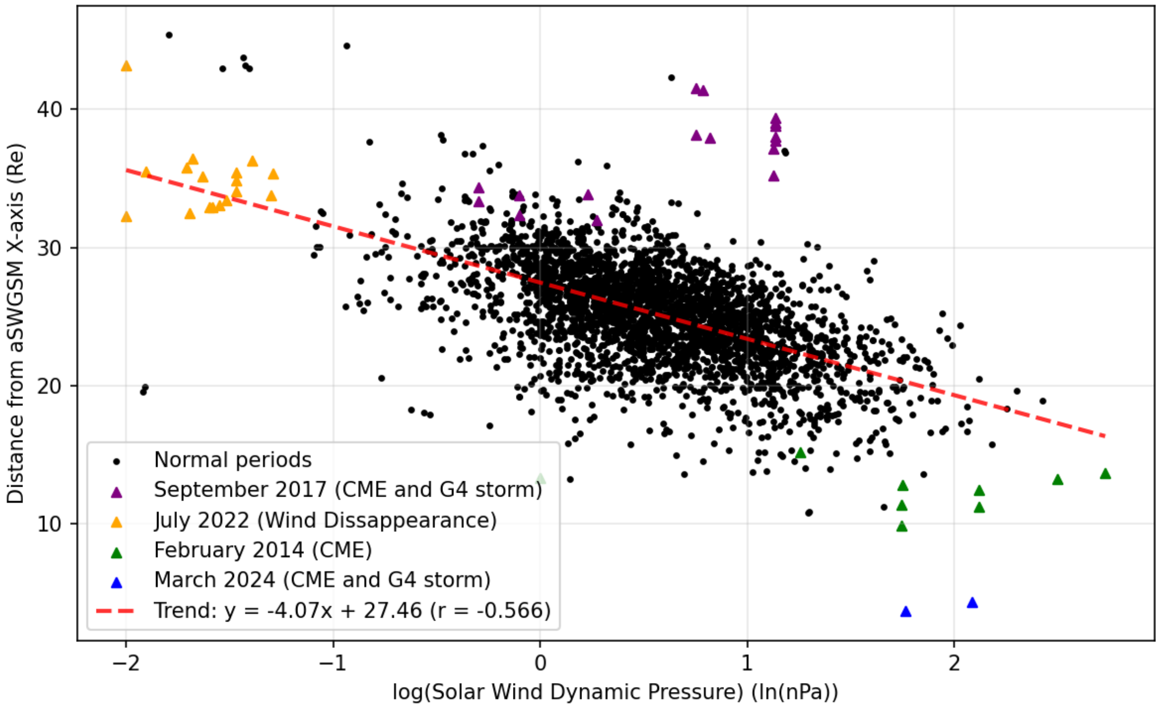


Figure 7.31: Distribution of distances from aSWGSM X-axis for detected crossings from 2013-07 to 2025-08, compared to the logarithm of the solar wind dynamic pressure from the OMNI database [53].

The overall trend seen in Figure 7.31 matches well with expectations, as higher solar wind dynamic pressures compress the magnetosphere. From the data shown in Figure 7.31, a correlation coefficient of -0.570 is calculated between the logarithm of the solar wind dynamic pressure and radius of crossings, showing a moderately strong negative correlation between solar wind dynamic pressure and distance from the aSWGSM X-axis to detected crossings.

7.6. Summary and Conclusion

This chapter has presented a comprehensive evaluation of the MAE transformer for automated magnetopause crossing detection. The MAE transformer outperformed gradient boosted classifiers, specifically in terms of precision and temporal accuracy, demonstrating its effectiveness in identifying true magnetopause crossings while minimising false positives.

Spatial validation showed that most detected crossings match well with established magnetopause models and labelled data. Most outliers are likely caused by real physical events, such as coronal mass ejections and geomagnetic storms. The application of the model to unlabelled data produced physically plausible results.

The spatial analysis showed that the average magnetopause distance at lunar orbit is larger during solar minimum and smaller during solar maximum, matching trends from the Shue et al. (1998, [70]) model for yearly averaged solar wind conditions. A more detailed analysis accounting for spacecraft orbit and exacter solar wind conditions could refine this finding further.

Overall, the results demonstrate the capability of machine learning models to effectively detect magnetopause crossings in spacecraft data at a large scale, detecting around 3000 crossings from 10 years of ARTEMIS data.

8

Conclusion and Future Work

This chapter summarises the key findings of the thesis, answers the research questions, and discusses the limitations and directions for future work.

8.1. Conclusion

This thesis set out to design and evaluate automatic machine learning methods for detecting magnetopause crossings in spacecraft data at lunar distance, and perform a preliminary analysis on the results. The main motivation was based on the need for automated methods to handle the large volumes of data being collected, detect crossings accurately, and the scientific interest in studying the magnetopause dynamics at lunar distance.

Two models were developed with this goal in mind. The first was a gradient boosted decision tree (GBDT) that used spectrogram statistics and plasma moments. This provided a good baseline. The second was a masked autoencoder (MAE) transformer. This model used reconstruction-based crossing detection on spectrogram energy flux data and other plasma moments, allowing it to analyse more complex parts of the data.

The results show that both approaches can detect magnetopause crossings effectively. The GBDT baseline achieved good precision at moderate recall rates, while the MAE transformer, using a gate thresholding method, achieved even higher precision and better temporal accuracy, while maintaining similar recall rates. In total, around 3000 crossings were detected from 2013-08 to 2025-08 in ARTEMIS P1 and P2 data. The spatial distribution of these crossings matched well with empirical magnetopause models, validating the detection methods. Outliers were shown to be caused by solar and geomagnetic activity.

8.2. Answers to the Research Questions

MRQ: How can machine learning be used to detect magnetopause crossings at lunar distance? Machine learning can be used to detect magnetopause crossings by using the differences between plasma and magnetic field properties on either side of the magnetopause. By training models on labelled data, machine learning algorithms can learn to identify the characteristic signatures of these crossings, such as changes in ion energy flux and density. Both the GBDT and MAE transformer models developed in this thesis demonstrated that machine learning can effectively identify these transitions with high precision and temporal accuracy, enabling automated detection of magnetopause crossings in large datasets. This confirms the hypothesis stated in subsection 3.3.1.

SRQ-1: What preprocessing steps are required to convert raw or region-labelled spacecraft data into a form suitable for boundary crossing detection using machine learning models?
Key steps include cleaning data, normalising features, and creating time-windowed inputs. Spectrograms and plasma moments were good features. This process is discussed in section 5.2

SRQ-2: What performance metrics best capture the performance of the magnetopause crossing detection?
Precision, recall, F-scores, and temporal error metrics captured different parts of performance well. Due to the importance of avoiding false positives in this application, precision had a high priority. Due to this, $F_{0.5}$ was particularly useful to emphasise precision while keeping a moderate recall.

SRQ-3: How can a magnetopause crossing ML model be evaluated and verified with little verification data?
Validation was achieved with cross-validation, consistency checks against empirical models, and sensitivity studies of preprocessing and post-processing choices.

SRQ-4: How do different machine learning models, and more traditional methods, compare for magnetopause crossing detection
Many models were considered. The MAE Transformer was chosen due to its high performance in similar tasks, its resistance against long-term distribution shifts in plasma data, and its possibility to train in a self-supervised manner. Compared to the other developed model, the GBDT, it offered better precision, F0.5 score, and better temporal accuracy when validating against labelled data.

SRQ-5: What patterns do machine learning models detect in the data to identify magnetopause crossings?
Transitions in ion energy flux and ion density showed to strongest correlation to detected crossings based on a qualitative assessment. These features showed clear changes at the crossing times, with both energy flux and density dropping when crossing from the magnetosheath to the magnetosphere, and increasing when crossing back.

SRQ-6: What patterns can be found in the detected crossings using the machine learning model, and how does it compare to the solar cycle?
Detected crossings are close to empirical predictions of the magnetopause location based on solar wind conditions. The yearly average radius of detected crossings follows the solar cycle, with a higher radius during solar minimum and a lower radius during solar maximum. This trend is more accurate when taking into account upstream solar wind conditions. Inward outliers are more common during solar maximum, and less common during solar minimum, while outward outliers' occurrence seems more affected by other phenomena. Within the magnetopause crossings, a strong correlation is observed between solar wind dynamic pressure and the average radius from the aSWGSM x-axis of the detected crossings, consistent with established magnetospheric physics.

SRQ-7: What are outliers (if any) in the detected crossings caused by?
Outliers are often caused by solar events influencing the magnetosphere, such as CMEs or extremely low solar wind pressure. These events can either cause the magnetopause to move significantly inward or outward, leading to crossings that deviate from typical locations,

or they can create similar plasma conditions within the magnetosheath or magnetosphere (e.g., increased density and energy flux in the current sheet) that mimic magnetopause crossings, resulting in false positives.

8.3. Future Work and Recommendations

Several directions for future research follow from this work.

8.3.1. Sophisticated Statistical Analyses

Building further on the analyses done in this study, the dataset could be used to study how the magnetopause position varies with solar wind conditions precisely, including its variation due to IMF conditions and geomagnetic activity, by relating the crossing times with upstream solar wind data and magnetosphere conditions. This could be performed using machine learning regression methods to predict the magnetopause position based on solar wind parameters, using the detected crossings as training data.

8.3.2. Event Disambiguation

Detected crossings may include false positives due to geomagnetic activity or other phenomena. Future work could focus on developing methods to disambiguate true magnetopause crossings from other events in a consistent manner, which could improve both the reliability of the detection methods and the quality of the resulting analysis.

8.3.3. Operational Applications

This thesis focused on using archived data to detect crossings, but the light computational load of both models means they could possibly be adopted for real-time applications. This could be useful to trigger spacecraft instruments to capture high-resolution data during crossings, or to inform space weather forecasting models.

In summary, this thesis has demonstrated that machine learning methods can detect magnetopause crossings automatically with decent temporal precision. With further improvements in validation, statistical analyses, and event disambiguation, these methods can become powerful tools for studying the dynamic magnetotail environment.

References

[1] I. G. Akay, Z. Kaymaz, and D. G. Sibeck. “Magnetotail boundary crossings at lunar distances: ARTEMIS observations”. In: *Journal of Atmospheric and Solar-Terrestrial Physics* 182 (2019), pp. 45–60. ISSN: 1364-6826. DOI: `https://doi.org/10.1016/j.jastp.2018.11.002`.

[2] M. R. Argall et al. “MMS SITL Ground Loop: Automating the Burst Data Selection Process”. In: *Frontiers in Astronomy and Space Sciences* 7 (2020), p. 54. DOI: `10.3389/fspas.2020.00054`.

[3] H. U. Auster et al. “The THEMIS Fluxgate Magnetometer”. In: *The THEMIS Mission*. Ed. by J. L. Burch and V. Angelopoulos. New York, NY: Springer New York, 2009, pp. 235–264. DOI: `10.1007/978-0-387-89820-9_11`. URL: `https://doi.org/10.1007/978-0-387-89820-9_11`.

[4] Menouar Azib et al. “Event Detection in Time Series: Universal Deep Learning Approach”. In: *arXiv preprint* arXiv:2311.15654 (2023). URL: `https://arxiv.org/abs/2311.15654`.

[5] Fran Bagenal. “Planetary Magnetospheres”. In: *Encyclopedia of the Solar System (Third Edition)*. Ed. by Bror Levine. Elsevier, 2013, pp. 251–263. DOI: `10.1016/B978-0-12-415845-0.00007-4`.

[6] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. “An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling”. In: *Advances in Neural Information Processing Systems (NeurIPS)*. arXiv:1803.01271. 2018, a–b.

[7] Shaojie Bai, J. Zico Kolter, and Vladlen Koltun. *An Empirical Evaluation of Generic Convolutional and Recurrent Networks for Sequence Modeling*. 2018. arXiv: `1803.01271 [cs.LG]`. URL: `https://arxiv.org/abs/1803.01271`.

[8] S. J. Bame et al. “Characteristics of the plasma sheet in the Earth’s magnetotail”. In: *Journal of Geophysical Research (1896-1977)* 72.1 (1967), pp. 113–129. DOI: `https://doi.org/10.1029/JZ072i001p00113`.

[9] Wolfgang Baumjohann and Rudolf Treumann. *Basic Space Plasma Physics - Revised Edition*. Mar. 2012. ISBN: 978-1-84816-894-7. DOI: `10.1142/P850`.

[10] Paul Murray Bellan. *Fundamentals of Plasma Physics*. Cambridge, UK: Cambridge University Press, 2006. ISBN: 0-521-52800-3.

[11] John Bonnell et al. “The electric field instrument (EFI) for THEMIS”. In: *Space Science Reviews* 141 (Dec. 2008), pp. 303–341. DOI: `10.1007/s11214-008-9469-2`.

[12] Joseph E. Borovsky, Michelle F. Thomsen, and Richard C. Elphic. “The driving of the plasma sheet by the solar wind”. In: *Journal of Geophysical Research: Space Physics* 103.A8 (1998), pp. 17617–17639. DOI: `10.1029/97JA02986`.

[13] I. K. Cheng, N. Achilleos, and A. Smith. “Automated bow shock and magnetopause boundary detection with Cassini using threshold and deep learning methods”. In: *Frontiers in Astronomy and Space Sciences* 9 (2022), p. 1016453. DOI: `10.3389/fspas.2022.1016453`.

[14] G. S. Chong et al. “Tailward flows in the vicinity of fast earthward flows”. In: *Journal of Geophysical Research: Space Physics* 126 (2021), e2020JA028978. DOI: `10.1029/2020JA028978`.

[15] A. V. Dmitriev, J. K. Chao, and D. J. Wu. “Comparative study of bow shock models using Wind and Geotail observations”. In: *Journal of Geophysical Research: Space Physics* 108.A12 (2003). DOI: `https://doi.org/10.1029/2003JA010027`.

[16] J. W. Dungey. “Interplanetary Magnetic Field and the Auroral Zones”. In: *Phys. Rev. Lett.* 6 (2 1961), pp. 47–48. DOI: `10.1103/PhysRevLett.6.47`.

[17] U. Feldman et al. “On the sources of fast and slow solar wind”. In: *Journal of Geophysical Research* 110.A7 (2005). “The slow speed solar wind … is characterized by its velocity of □400 km s□¹” (abstract), A07109. DOI: `10.1029/2004JA010918`.

[18] S. Fuselier et al. “Global-Scale Processes and Effects of Magnetic Reconnection on the Geospace Environment”. In: *Space Science Reviews* 220 (Apr. 2024). DOI: `10.1007/s11214-024-01067-0`.

[19] M. Gupta et al. “Real-time Change-Point Detection: A deep neural network autoencoder approach”. In: *Applied Soft Computing* (2022). uses autoencoder for quick change detection in streaming data. DOI: `10.1016/j.asoc.2022.109153`.

[20] Gousia Habib and Shaima Qureshi. “Optimization and acceleration of convolutional neural networks: A survey”. In: *Journal of King Saud University - Computer and Information Sciences* 34.7 (2022), pp. 4244–4268. ISSN: 1319-1578. DOI: `https://doi.org/10.1016/j.jksuci.2020.10.004`.

[21] M.A. Hapgood. “Space physics coordinate transformations: A user guide”. In: *Planetary and Space Science* 40.5 (1992), pp. 711–717. ISSN: 0032-0633. DOI: `https://doi.org/10.1016/0032-0633(92)90012-D`.

[22] Simon Hawkins et al. “Outlier Detection Using Replicator Neural Networks”. In: *Proceedings of the 4th International Conference on Data Warehousing and Knowledge Discovery (DaWaK)*. Vol. 2454. Lecture Notes in Computer Science. Springer, 2002, pp. 170–180. ISBN: 3-540-44123-9. DOI: `10.1007/3-540-46145-0_17`.

[23] Kaiming He et al. “Masked Autoencoders Are Scalable Vision Learners”. In: *CoRR* abs/2111.06377 (2021). arXiv: `2111.06377`. URL: `https://arxiv.org/abs/2111.06377`.

[24] Po-Han K. Hou and Sung-Chi C. Hsieh. “Regression-Based Physics-Informed Neural Networks for Magnetopause Tracking”. In: *arXiv preprint* arXiv:2306.09621 (2025). URL: `https://arxiv.org/abs/2306.09621`.

[25] Joseph H. Howe Herbert C.; Binsack. “Explorer 33 and 35 plasma observations of magnetosheath flow”. In: *Journal of Geophysical Research: Atmospheres 1972-jul 01 vol. 77 iss. 19* 77 (19 1972). DOI: `10.1029/ja077i019p03334`.

[26] Hughes. “The Magnetopause, Magnetotail, and Magnetic Reconnection”. In: *Introduction to Space Physics*. Ed. by M. G. Kivelson and C. T. Russell. Cambridge University Press, 1995, pp. 227–285.

[27] A. J. Hundhausen. “The Solar Wind”. In: *Introduction to Space Physics*. Ed. by M. G. Kivelson and C. T. Russell. Cambridge University Press, 1995, pp. 91–128.

[28] P. Janardhan et al. “Resolving the enigmatic solar wind disappearance event of 11 May 1999”. In: *Journal of Geophysical Research: Space Physics* 110.A08 (2005), A08101. DOI: `10.1029/2004JA010535`.

[29] Sahib Julka, Rodion Ishmukhametov, and Michael Granitzer. “Deep Active Learning with Concept Drifts for Detection of Mercury’s Bow Shock and Magnetopause Crossings”. In: *Machine Learning and Knowledge Discovery in Databases. LOD 2023*. Vol. 14505. Lecture Notes in Computer Science. Springer, 2024, pp. 390–404. DOI: `10.1007/978-3-031-53969-5_29`.

[30] Sahib Julka et al. “Deep Active Learning for Detection of Mercury’s Bow Shock and Magnetopause Crossings”. In: *Proceedings of ECML PKDD 2022*. Vol. 13716. Lecture Notes in Artificial Intelligence. Springer, 2023, pp. 452–467. DOI: `10.1007/978-3-031-26412-2_28`.

[31] Mahmoud Khalil, Ahmad Khalil, and Alioune Ngom. *A Comprehensive Study of Vision Transformers in Image Classification Tasks*. 2023. arXiv: `2312.01232 [cs.CV]`. URL: `https://arxiv.org/abs/2312.01232`.

[32] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. 2017. arXiv: `1412.6980 [cs.LG]`. URL: `https://arxiv.org/abs/1412.6980`.

[33] M. Kubyshkina et al. “Further evidence for the role of magnetotail current shape in substorm initiation”. In: *Earth, Planets and Space* 67.139 (2015). DOI: `10.1186/s40623-015-0304-1`.

[34] S. N. Kuznetsov and A. V. Suvorova. “An Empirical Model of the Magnetopause for Broad Ranges of Solar Wind Pressure and BZ IMF”. In: *Polar Cap Boundary Phenomena*. Ed. by J. Moen, A. Egeland, and M. Lockwood. Dordrecht: Springer Netherlands, 1998, pp. 51–61. DOI: `10.1007/978-94-011-5214-3_5`.

[35] Tianyang Lin et al. “A survey of transformers”. In: *AI Open* 3 (2022), pp. 111–132. ISSN: 2666-6510. DOI: `https://doi.org/10.1016/j.aiopen.2022.10.001`.

[36] Imke De Pater; Jack Jonathan Lissauer. *Planetary sciences*. Cambridge University Press, 2001. ISBN: 0521482194; 9780521482196.

[37] Yong Liu et al. *Timer: Generative Pre-trained Transformers Are Large Time Series Models*. 2024. arXiv: `2402.02368 [cs.LG]`. URL: `https://arxiv.org/abs/2402.02368`.

[38] X. Ma et al. “Statistical Study of Solar Wind, Magnetosheath, and Magnetotail Plasma and Field Properties: 12+ Years of THEMIS Observations and MHD Simulations”. In: *arXiv preprint arXiv:2202.01879* (2022). URL: `https://arxiv.org/abs/2202.01879`.

[39] Xuanye Ma et al. “Statistical Study of Solar Wind, Magnetosheath, and Magnetotail Plasma and Field Properties: 12+ Years of THEMIS Observations and MHD Simulations”. In: *Journal of Geophysical Research: Space Physics* 125.10 (2020). DOI: `10.1029/2020JA028209`.

[40] Pankaj Malhotra et al. “LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection”. In: *CoRR* abs/1607.00148 (2016). arXiv: `1607.00148`. URL: `http://arxiv.org/abs/1607.00148`.

[41] D. J. McComas et al. *ACE Solar Wind Electron, Proton, and Alpha Monitor (SWEPAM) Plasma Moments, Level 2 (H2), 1 h Data*. `https://cdaweb.gsfc.nasa.gov/cgi-bin/eval2.cgi?dataset=AC_H2_SWE&index=sp_phys`. Data set. Space Physics Data Facility. DOI: 10.48322/9w01-2555. Accessed: 2025-10-16. 2022. DOI: `10.48322/9w01-2555`.

[42] J. P. McFadden et al. “The THEMIS ESA Plasma Instrument and In-flight Calibration”. In: *Space Science Reviews* 141 (2008), pp. 277–302. DOI: `10.1007/s11214-008-9440-2`. URL: `https://doi.org/10.1007/s11214-008-9440-2`.

[43] V. G. Merkin et al. “Contribution of Bursty Bulk Flows to the Global Dipolarization of the Magnetotail during an Isolated Substorm”. In: *Journal of Geophysical Research: Space Physics* 124 (2019), pp. 8647–8668. DOI: `10.1029/2019JA026872`.

[44] NASA Coordinated Data Analysis Web. *Coordinated Data Analysis Web (CDAWeb)*. Accessed: 2025-03-31. 2024. URL: `https://cdaweb.gsfc.nasa.gov/`.

[45] NASA Magnetopause Database. *Magnetopause and Boundary Crossings - OMNIWeb*. `https://omniweb.gsfc.nasa.gov/ftpbrowser/magnetopause/Intro.html`. Accessed: 2025-03-31. 2024.

[46] NASA Satellite Situation Center. *Satellite Situation Center Web (SSCWeb)*. `https://sscweb.gsfc.nasa.gov/`. Accessed: 2025-03-31. 2025.

[47] N. F. Ness et al. “Early results from the magnetic field experiment on lunar Explorer 35”. In: *Journal of Geophysical Research (1896-1977)* 72.23 (1967), pp. 5769–5778. DOI: `https://doi.org/10.1029/JZ072i023p05769`.

[48] G. Nguyen et al. “Massive Multi-Mission Statistical Study and Analytical Modeling of the Earth’s Magnetopause: 1. A Gradient Boosting Based Automatic Detection of Near-Earth Regions”. In: *Journal of Geophysical Research: Space Physics* 127.1 (2022). DOI: `https://doi.org/10.1029/2021JA029773`.

[49] G. Nguyen et al. “Massive Multi-Mission Statistical Study and Analytical Modeling of the Earth’s Magnetopause: 2. Shape and Location”. In: *Journal of Geophysical Research: Space Physics* 127.1 (2022). DOI: `10.1029/2021JA029774`.

[50] K. W. Ogilvie et al. *WIND Spacecraft Solar Wind Data (via CDAWeb)*. `https://cdaweb.gsfc.nasa.gov/`. Data from WIND spacecraft, via the NASA CDAWeb SPDF archive. Accessed: 2025-10-16. 2025.

[51] S. Ohtani, M. Shay, and T. Mukai. “Temporal structure of the fast convective flow in the plasma sheet: Comparison between observations and two-fluid simulations”. In: *Journal of Geophysical Research: Space Physics* 109 (2004), A03210. DOI: `10.1029/2003JA010342`.

[52] Natalia E. Papitashvili and Joseph H. King. *OMNI 27-Day Data*. Accessed on (DATE). NASA Space Physics Data Facility, 2020. DOI: `10.48322/gbpg-5r77`. URL: `https://doi.org/10.48322/gbpg-5r77`.

[53] Natalia E. Papitashvili and Joseph H. King. *OMNI 5-min Data*. Accessed on (DATE). NASA Space Physics Data Facility, 2020. DOI: `10.48322/gbpg-5r77`. URL: `https://doi.org/10.48322/gbpg-5r77`.

[54] E. N. Parker. “Dynamics of the Interplanetary Gas and Magnetic Fields”. In: *The Astrophysical Journal* 128 (Nov. 1958), p. 664. DOI: `10.1086/146579`. URL: `https://ui.adsabs.harvard.edu/abs/1958ApJ...128..664P`.

[55] C. T. Petrinec S. M.; Russell. “Near-Earth magnetotail shape and size as determined from the magnetopause flaring angle”. In: *Journal of Geophysical Research: Atmospheres 1996-jan 01 vol. 101 iss. A1* 101 (A1 1996). DOI: `10.1029/95ja02834`.

[56] Tony Phillips. *SpaceWeather.com: News and Information about Meteor Showers, Solar Flares, Geomagnetic Storms, and Auroras*. `https://www.spaceweather.com/`. Accessed: 2025-10-13. 2025.

[57] Eric R. Priest and Terry G. Forbes. *Magnetic Reconnection: MHD Theory and Applications*. 1st ed. Cambridge, UK: Cambridge University Press, 2000. ISBN: 0-521-48179-1.

[58] Hendrik Purwins et al. “Deep learning for audio signal processing”. In: *IEEE Journal of Selected Topics in Signal Processing* 13.2 (2019), pp. 206–219. DOI: `10.1109/JSTSP.2019.2908700`.

[59] Tao Qin et al. “Physics-informed recurrent neural networks for dynamical system modeling”. In: *Advanced Engineering Informatics* 55 (2023), p. 101877. ISSN: 0959-1524. DOI: `10.1016/j.aei.2023.101877`.

[60] Katie Raymer. *Geotail magnetopause crossing database 1996 - 2015*. Version 1. Zenodo, Mar. 2020. DOI: `10.5281/zenodo.3719411`. URL: `https://doi.org/10.5281/zenodo.3719411`.

[61] David G. Roelof Edmond C.; Sibeck. “Magnetopause Shape as a Bivariate Function of Interplanetary Magnetic Field B z and Solar Wind Dynamic Pressure”. In: *Journal of Geophysical Research: Atmospheres vol. 98 iss. A12* 98 (A12 1993). DOI: `10.1029/93ja02362`. URL: `libgen.li/file.php?md5=3f84446f88995a4b4fb8895b3ceea723`.

[62] A. Runov and et al. “A THEMIS multicase study of dipolarization fronts in the near-Earth magnetotail”. In: *Journal of Geophysical Research: Space Physics* 116 (2011), A05215. DOI: `10.1029/2010JA016316`.

[63] C. T. Russell and R. C. Elphic. “Initial ISEE Magnetometer Results: Magnetopause Observations (Article published in the special issues: Advances in Magnetospheric Physics with GEOS-1 and ISEE-1 and 2.)” In: *Space Science Reviews* 22.6 (1978), pp. 681–715. DOI: `10.1007/BF00212619`.

[64] Stefania Russo et al. “Anomaly Detection using Deep Autoencoders for in-situ Wastewater Systems Monitoring Data”. In: *arXiv preprint arXiv:2002.03843* (2020). URL: `https://arxiv.org/abs/2002.03843`.

[65] Karel Schrijver et al. *Principles Of Heliophysics: a textbook on the universal processes behind planetary habitability*. 2024. arXiv: `1910.14022 [astro-ph.SR]`.

[66] K. Seki et al. “A Review of General Physical and Chemical Processes Related to Plasma Sources and Losses for Solar System Magnetospheres”. In: *Space Science Reviews* 192 (Aug. 2015). DOI: `10.1007/s11214-015-0170-y`.

[67] A. Sen et al. “Input–Output Optics as a Causal Time Series Mapping”. In: *Physical Review Research* 7.2 (2025), p. 023015. DOI: `10.1103/PhysRevResearch.7.023015`.

[68] Yuge Shi et al. “How Robust is Unsupervised Representation Learning to Distribution Shift?” In: *Proceedings of the 39th International Conference on Machine Learning (ICML)*. 2022.

[69] J.-H. Shue et al. “A new functional form to study the solar wind control of the magnetopause size and shape”. In: *Journal of Geophysical Research* 102.A5 (1997), pp. 9497–9511. DOI: `10.1029/97JA00196`.

[70] J.-H. Shue et al. “Magnetopause location under extreme solar wind conditions”. In: *Journal of Geophysical Research: Space Physics* 103.A8 (1998), pp. 17691–17700. DOI: `10.1029/98ja01103`.

[71] D. G. Sibeck, R. E. Lopez, and E. C. Roelof. “Solar wind control of the magnetopause shape, location, and motion”. In: *Journal of Geophysical Research: Space Physics* 96 (A4 1991), pp. 5489–5495. DOI: `10.1029/90ja02464`.

[72] Solar Influences Data Analysis Center (SIDC). *The solar cycle minimum has passed, solar cycle 25 is on the rise*. Accessed: 2025-09-28. 2020. URL: `https://www.sidc.be/SILSO/solar-cycle-minimum-passed-december-2019`.

[73] Frances A Staples et al. *THEMIS Magnetopause Crossing Database*. Version 1. Zenodo, Mar. 2020. DOI: `10.5281/zenodo.3700504`. URL: `https://doi.org/10.5281/zenodo.3700504`.

[74] J. B. Tao et al. “Kinetic instabilities in the lunar wake: ARTEMIS observations”. In: *Journal of Geophysical Research: Space Physics* 117.A03 (2012), A03106. DOI: `10.1029/2011JA017364`.

[75] M. Thill et al. “Temporal convolutional autoencoder for unsupervised anomaly detection in time series data”. In: *Future Generation Computer Systems* (2021). uses convolutional autoencoder in time series anomaly / change detection.

[76] B.T. Tsurutani et al. “A review of the ISEE-3 Geotail magnetic field results”. In: *Planetary and Space Science* 34.10 (1986), pp. 931–960. ISSN: 0032-0633. DOI: `https://doi.org/10.1016/0032-0633(86)90004-8`.

[77] Ashish Vaswani et al. “Attention Is All You Need”. In: *CoRR* abs/1706.03762 (2017). arXiv: `1706.03762`. URL: `http://arxiv.org/abs/1706.03762`.

[78] J. J. Wang et al. “A Deep Learning Framework for Constitutive Modeling Based on Temporal Convolutional Networks”. In: *Journal of the Mechanics and Physics of Solids* (2022). uses TCN for history dependence in material modeling.

[79] A. L. E. Werner et al. “Modeling the Multiple CME Interaction Event on 6–9 September 2017 with WSA-ENLIL+Cone”. In: *Space Weather* 17.2 (2019), pp. 357–369. DOI: `10.1029/2018SW001993`.

[80] *WIND Spacecraft*. `https://wind.nasa.gov/`. NASA / WIND mission, spin-stabilized spacecraft launched 1 November 1994. 1994.

[81] Jiehui Xu et al. “Anomaly Transformer: Time Series Anomaly Detection with Association Discrepancy”. In: *CoRR* abs/2110.02642 (2021). arXiv: `2110.02642`. URL: `https://arxiv.org/abs/2110.02642`.

[82] Daksha Yadav, Sabrina Zhang, and Tom Jin. “Transformer based anomaly detection on multivariate time series subledger data”. In: (2023). URL: `https://www.amazon.science/publications/transformer-based-anomaly-detection-on-multivariate-time-series-subledger-data`.

[83] Zahra Zamanzadeh Darban et al. “Deep Learning for Time Series Anomaly Detection: A Survey”. In: *ACM Computing Surveys* 57.1 (Oct. 2024), pp. 1–42. ISSN: 1557-7341. DOI: `10.1145/3691338`. URL: `http://dx.doi.org/10.1145/3691338`.

[84] Aston Zhang et al. *Dive into Deep Learning*. Cambridge University Press, 2021. URL: `https://d2l.ai`.

[85] Haoyi Zhou et al. “Informer: Beyond Efficient Transformer for Long Sequence Time-Series Forecasting”. In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 12. 2021, pp. 11106–11115.

[86] Weizhi Zhu, Xiaoyu Wang, and Ashesh Chattopadhyay. *Physics-constrained LSTM for learning chaotic dynamics*. 2023. arXiv: `2302.10779 [physics.comp-ph]`. URL: `https://arxiv.org/abs/2302.10779`.

A

Code Availability

The source code used for data preprocessing, model training, and analysis is available on GitHub: `https://github.com/ianmaes/Magnetotail-Boundary-with-ML`. Here two demos are available to illustrate how to use the code for training and inference. One demo covers the extraction of data from NASA Coordinated Data Analysis Web (2024, [44]) using pyspedas using custom classes defined in the repository, and to use this data for inference using a pretrained model. The second demo illustrates how to train a new model from scratch using the provided data preprocessing and model training scripts.

B

System Requirements

Table B.1: System Requirements

ID	**Requirement Description**
SR-1	The system shall accept in-situ spacecraft data as input, including magnetic field vector components (B_x, B_y, B_z), plasma density, plasma temperature, plasma velocity, ion and electron spectrograms, and spacecraft position, and TBD other parameters.
SR-2	The system shall automatically preprocess input data to ensure compatibility for magnetopause crossing detection.
SR-3	The system shall output timestamps of detected magnetopause crossings within the input time period.
SR-4	The system shall pinpoint each crossing with a maximum error of 20 minutes, 90% of the time.
SR-5	The system shall process input data for magnetopause crossing detection within an hour.
SR-6	The system shall be designed to process Earth-based data, specifically focusing on the ARTEMIS mission dataset.
SR-7	The system shall allow for integration of output data with downstream analysis tools or databases.
SR-8	The system shall provide an interface to ingest input data from the NASA Coordinated Data Analysis Web (CDAWeb) database and other TBD databases.
SR-9	The system shall be designed to operate on a single computer equipped with modern GPU and CPU capabilities.
SR-10	The system (specifically, the machine learning model) shall process multidimensional time series data, including 1D and 2D data.
SR-11	The system has a precision of at least 75% when evaluated against a labelled dataset of known magnetopause crossings, and based on the temperal accuracy requirement SR-4.

C

Coordinate Transformation

GSE $\to$ aSWGSM coordinate transformation

The transformation from GSE (Geocentric Solar Ecliptic) to aSWGSM (aberrated Solar Wind Geocentric Solar Magnetospheric) coordinates involves sequential rotations to account for Earth's orbital motion and magnetic dipole alignment.

The first step applies an aberration correction to account for the relative motion between Earth and the solar wind. This aberration effect arises from several sources, including Earth's orbital motion around the Sun, but more generally from the velocity difference between the observer (Earth) and the solar wind frame. The correction involves rotations about both the Y-axis (angle β, accounting for orbital eccentricity and other velocity components) and Z-axis (angle α, for the primary aberration effect, typically around 4°):

$$\mathbf{R}_{\text{aberration}} = \begin{pmatrix} \cos(\alpha)\cos(\beta) & -\sin(\alpha) & \cos(\alpha)\sin(\beta) \\ \sin(\alpha)\cos(\beta) & \cos(\alpha) & \sin(\alpha)\sin(\beta) \\ -\sin(\beta) & 0 & \cos(\beta) \end{pmatrix} \tag{C.1}$$

This transforms the position vector to aberrated coordinates: $\mathbf{r}_{\text{aSWGSE}} = \mathbf{R}^T_{\text{aberration}}\mathbf{r}_{\text{GSE}}$.

The second step aligns the coordinate system with Earth's magnetic dipole. We first rotate the dipole direction vector from GSE to aSWGSE coordinates:

$$\mathbf{d}_{\text{aSWGSE}} = \mathbf{R}^T_{\text{aberration}}\mathbf{d}_{\text{GSE}} \tag{C.2}$$

Then project this onto the aSWGSE YZ-plane by setting the X-component to zero $\mathbf{d}_{\text{proj}} = (0, d_y, d_z)$ The rotation angle θ about the X-axis is calculated from the normalised projected dipole components $\hat{\mathbf{d}}_{\text{proj}} = \frac{\mathbf{d}_{\text{proj}}}{|\mathbf{d}_{\text{proj}}|}$:

$$\cos(\theta) = \hat{d}_z \tag{C.3}$$

$$\sin(\theta) = \hat{d}_y \tag{C.4}$$

The dipole alignment rotation matrix is:

$$\mathbf{R}_{\text{dipole}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{pmatrix} \tag{C.5}$$

The complete transformation combines both rotations sequentially:

$$\mathbf{r}_{\text{aSWGSM}} = \mathbf{R}_{\text{dipole}} \mathbf{R}^{T}_{\text{aberration}} \mathbf{r}_{\text{GSE}} \tag{C.6}$$

This process ensures the final aSWGSM coordinate system accounts for both orbital aberration and time-varying magnetic dipole orientation.