UNIVERSITY OF THE
FREE STATE
UNIVERSITEIT VAN DIE
VRYSTAAT
YUNIVESITHI YA
FREISTATA

UFS

COMBINING DRILL-CORE HYPERSPECTRAL AND GEOCHEMICAL DATA BY DEEP LEARNING TO ENHANCE DRILL-CORE MINERAL MAPPING

by

Dzhavhelo Keystone MATIMBI

STUDENT NO: 2019100466

MSc dissertation submitted to the Faculty of Natural and Agricultural Sciences, University of the Free State, in fulfilment of the requirements for the degree of Masters.

Supervisor: Prof. E.J.M Carranza

Department of Geology

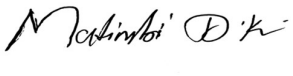
University of the Free State

Bloemfontein, South Africa

October 2025

DECLARATION

I, **Dzhavhelo Keystone Matimbi** declare that the dissertation, *Combining Drill-Core Hyperspectral and Geochemical Data by Deep Learning to Enhance Drill-Core Mineral Mapping*, submitted for qualification of Master's Degree majoring in Geology at the University of the Free State is my own independent work.

Student's signature:  Date: 23/09/2025

SUPERVISOR DECLARATION

I, Prof. Emmanuel John M. Carranza, approve the submitted MSc thesis of Mr. **Dzhavhelo Keystone Matimbi** for assessment and that the submitted work has not previously, either in part or in its entirety, been submitted to the examiners or moderators.

Signature: Date: October 6, 2025

Name: Prof. E.J.M Carranza

DEDICATION

This dissertation is dedicated to my parents Ms Matimbi Shoni Maria and Mr Maduwa Khathutshelo, and servants of God (Apostle A.E and Prophet M.S Mauna).

ACKNOWLEDGEMENT

Prior to everything, it is my desire to convey my sincerest gratitude to God Almighty for His steadfast love, kindness, and generosity that have guided me up till this moment. He gave me wisdom, knowledge, insight, revelation and intelligence to complete my studies. '*Even your smallest and humblest family will become as great as powerful nation. When the right time comes, I will make this happen quickly. I am the Lord'* (Isaiah 60:22).

My preference would be to give special thanks to my supervisor Prof. E.J.M Carranza, for giving me the direction and supporting me in raising the standard of this work by providing me with guidance and constructive suggestions. Thank you for always being available, motivating me to do better, and believing in me. Additional appreciation goes to my sponsor (Nelson Mandela Mining Precinct) for sponsoring me throughout my studies.

To my colleagues, Martha Amwaama, Nomonde Mabogo, Tshimangadzo Ndou, and Karabo Mahlapha, thank you for the intellectual encouragement, friendship, love, jokes and all other forms of support you showed me throughout my studies. It was not an easy academic journey, but I am humbly grateful because I got a chance to share experience with you.

To my beloved parents, Ms Shoni Maria Matimbi and Mr Khathutshelo Maduwa, and my sibling, thank you for your unwavering love, prayers and care. You gave me motivation when I needed it most. My sincere appreciation also goes to the servants of God (Apostle A.E and Prophet M.S Mauna) for guiding and praying for me throughout this journey. Thank you for being the best spiritual parents.

ABSTRACT

Drill-core mineral mapping is critical in orebody modelling, but traditional analysis based on manual geological logging is time-consuming, subjective, and prone to bias. Hyperspectral (HS) imaging provides a quick, high-resolution, and non-invasive alternative; however, traditional approaches to analyse HS data also rely on human interpretation, which limits efficiency and reliability. Recent advances in deep learning (DL) offer ways to handle these challenges by automatically extracting complex spatial and spectral features from massive datasets (e.g., HS data), eliminating the need for human feature engineering.

This study integrated drill-core HS data (in a form of reflectance spectra) and geochemical data (geochemical assay) with DL methods to enhance drill-core mineral mapping. The research was guided by three objectives: (i) quantifying uncertainty in drill-core mineral mapping using DL approaches, (ii) evaluating and comparing the performance of convolutional neural networks (CNNs) and recurrent neural networks (RNNs), specifically long-short term memory (LSTM) models, and (iii) developing a hybrid CNN-LSTM model to enhance mineral mapping.

The results demonstrated that all three models had high predictive performances with low MAEs and uncertainty. The hybrid CNN-LSTM model outperformed the stand-alone models, with lowest MAE (0.0258), followed by the LSTM model (MAE = 0.0278), and CNN model (MAE = 0.0312). The improved performance of the hybrid model demonstrated the benefit of integrating spatial–spectral feature extraction with sequential dependency modelling, whereas the relative strength of LSTM over CNN emphasizes the importance of sequential features.

This research demonstrated that integrating drill-core HS and geochemical data using DL methods can significantly enhance drill-core mineral mapping and mineral abundance estimation in drill-core datasets, while acquiring low MAEs.

Keywords: integrating; drill-core; hyperspectral; geochemical; deep-learning; enhance; mineral mapping.

TABLE OF CONTENTS

DECLARATION ii
SUPERVISOR DECLARATION iii
DEDICATION iv
ACKNOWLEDGEMENT v
ABSTRACT vi
TABLE OF FIGURES x
TABLES xi
LIST OF ABBREVIATIONS xii
LIST OF EQUATIONS xiii
CHAPTER ONE: INTRODUCTION 1
1.1. Background 1
1.2. Significance of Accurate Drill-Core Mineral Mapping 2
1.3. Enhanced HS Imaging for Mineral Mapping 3
1.4. The Role of Geochemical Data in Identifying Minerals 4
1.5. Integration of Deep Learning Models 6
1.6. Problem Statement 6
1.7. Research Aim and Objectives 7
1.8. Research Questions 8
1.9. Methodology 8
1.10. Thesis Structure 9
CHAPTER 2: LITERATURE REVIEW 11
2.1. Hyperspectral Imaging/Data in Mapping Minerals 12
2.2. Overview of Deep Learning Algorithms 14
2.2.1. Artificial Neural Networks: Convolutional Neural Networks and Recurrent Neural Networks 16
2.3. Ensemble Learning 18
2.4. Prior Research on Integration of Geochemical and Hyperspectral Data 20
2.5. Concluding Remarks 25
CHAPTER 3: METHODOLOGY 27
3.1. Case Study Area 27
3.2. Drill-Core Geochemical Data 28
3.3. Drill-Core Hyperspectral Data 31
3.3.1. Hyperspectral Data Pre-processing 31
3.4. Geochemical Data Pre-processing 33
3.4.1. Centred Log-Ratio Transformation 33

3.5. Model Design and Selection 38
3.5.1. Convolutional Neural Networks (CNNs) 39
3.5.2. Long-Short Term Memory (LSTM) 44
3.6. Concluding Remarks 47
CHAPTER 4: HYBRID CNN-LSTM IMPLEMENTATION AND MODEL TESTING 49
4.1. Hybrid CNN-LSTM Implementation 49
4.2. CNN, LSTM, and Hybrid CNN-LSTM Model Testing 52
4.3. Uncertainty Quantification 56
4.4. Mineral Abundance and Mineral Association Maps 59
4.6. Concluding Remarks 67
CHAPTER 5: DISCUSSION 69
CHAPTER 6: CONCLUSION, RECCOMENDATION AND FUTURE RESEARCH WORK 74
REFERENCES 77
APPENDICES 84
Appendix A: Hyperspectral Data Extraction 84
Appendix B: Hyperspectral Data Transformation 87
Appendix C: Geochemical Data Transformation 89
Appendix D: Model Training (CNN-LSTM) 94
Appendix E: Model Testing (CNN-LSTM) 100
Appendix F: Mineral Abundance and Mineral Association Maps (CNN-LSTM) 107

TABLE OF FIGURES

Figure 1.1. Flow diagram for the DL method to integrate HS and geochemical data for drill-core mineral mapping…….....……8
Figure 3.1. (A) Map of Western Australia showing the location of Rocklea Dome. (B) Rocklea Dome geologic map showing the Rocklea Dome channel iron deposit (CID) (from Haest et al. 2012) 25
Figure 3.2. (A) Rocklea Dome drill-holes (blue and red dots) and (B) Figure 3.3 showing drill-hole used in this study highlighted in red.....27
Figure 3.3. Data from drill-holes (in red colour) from Rocklea Dome were the ones used in this study..... 27
Figure 3.4. Mineral abundances per drill-hole.....28
Figure 3.5. (A) Rocklea dome extracted stacked spectra for RKC holes, with wavelength and reflectance arbitrary units and (B) PCA of the extracted reflectance spectra showing transformation of original high-dimensional HS data into a reduced set of uncorrelated components.....30
Figure 3.6. Frequency distributions of CLR-transformed element data from drill-core samples, highlighting differences in skewness and modality among the elements.....33
Figure 3.7. Correlation matrix of CLR-transformed geochemical data illustrating the pairwise correlation coefficient among the compositional variables, revealing potential interdependences.....34
Figure 3.8. PCA of CLR-transformed data.....35
Figure 3.9. CNN model consisting of input, convolution, pooling and fully connected layers.....37
Figure 3.10. Average training loss across folds and average MAE across folds for CNN model over 50 epochs.....40
Figure 3.11. LSTM structure showing the input, forget and output gates.....42
Figure 3.12. Average training loss across folds and average MAE across folds for LSTM model over 50 epochs.....43
Figure 4.1. Curves of average training and validation loss across folds, and average training and validation MAE across folds for hybrid CNN-LSTM model over 50 epochs.....48
Figure 4.2. Scatter-plots of the true vs predicted mineral abundances of (A) CNN model, (B) LSTM model, and (C) hybrid CNN-LSTM model.....53
Figure 4.3. Mineral abundance map created from the mineral abundance predictions which were predicted by the CNN model.....58
Figure 4.4. Mineral abundance map created from the mineral abundance predictions which were predicted by the LSTM model.....59
Figure 4.6. Mineral abundance map created from the mineral abundance predictions which were predicted by the hybrid CNN-LSTM model.....60
Figure 4.7. Mineral association maps based on mineral abundances predicted by (A) CNN model, (B) LSTM model, and (C) hybrid CNN-LSTM model.....61

TABLES

Table 3.1 Training loss with training MAE and validation loss with validation MAE during CNN model training on five-fold cross validation (epoch 50/50) 40
Table 3.2. Training loss with training MAE and validation loss with validation MAE during LSTM model training on five-fold cross validation (epoch 50/50) 44
Table 4.1. Training loss with training MAE and validation loss with validation MAE during CNN model training on five-fold cross validation.....46
Table 4.2. Mineral abundance MAEs and overall MAE during model testing of CNN, LSTM and a hybrid CNN-LSTM models.....52
Table 4.3: Performances of CNN, LSTM, and CNN-LSTM Models across mineral abundances based on R^2 and overall R^2 52
Table 4.5. Evaluation of model confidence and uncertainty metrics (CNN, LSTM, and hybrid CNN-LSTM) 56
Table 4.6. Average mineral abundance per cluster from the CNN model 60
Table 4.7. Average mineral abundance per cluster from the LSTM model 61
Table 4.8. Average mineral abundance per cluster from the hybrid CNN-LSTM model.....61
Table 5.1. Mineral abundance MAEs and overall MAE during model testing of CNN, LSTM and a hybrid CNN-LSTM models, and model uncertainty.....67

LIST OF ABBREVIATIONS

AdaBoost	Adaptive Boosting
ANN	Artificial Neural Networks
ALR	Addictive Log-Ratio
BNNs	Bayesian Neural Networks
CkSVM	Composite Kernel Support Vector Machine
CLR	Centred Log-Ratio
CNN	Convolutional Neural Networks
DL	Deep Learning
DT	Decision Tree
GA	Generic Algorithm
GBDT	Gradient Boosting Decision Tree
GPS	Geographical Positioning System
HDBSCAN	Hierarchical Density-Based Clustering
HS	Hyperspectral
HSS	Hierarchical Spatial-Spectral
ILR	Isometric Log-Ratio
KNN	K-Nearest Neighbours
LR	Logistic Regression
LSTM	Long-Short Term Memory
MAE	Mean Absolute Error
MC	Monte Carlo
ML	Machine Learning
PCA	Principal Component Analysis
R^2	Coefficient of determination
ReLU	Rectified Linear Unit
RF	Random Forest
RNN	Recurrent Neural Networks
SAM	Spectral Angle Mapper
SEM	Scanning Electron Microscopy
SSA	Sparrow Search Algorithm
SVM	Support Vector Machine

XGBoost	Extreme Gradient Boosting

LIST OF EQUATIONS

Equation 3.4.1.	Geometric Mean
Equation 3.5.1.	ReLU activation function
Equation 3.5.2.	Computational Process
Equation 4.2.1.	Mean Absolute Error calculation
Equation 4.3.1.	Epistemic Uncertainty calculations

CHAPTER ONE: INTRODUCTION

1.1. Background

Every year, a lot of drilling is done as part of exploration activities for the evaluation and discovery of ore deposits. Drill-core mineral mapping has consequently emerged as a crucial phase in the mineral exploration process (Tuşa et al., 2020). Geological data from drill-core samples are essential for describing possible mineral accumulations. Geologists typically visually examine drill cores, noting important details such as rock type, texture, alteration facies, ore-forming minerals, and structural features. Nevertheless, this procedure is laborious and heavily reliant on the knowledge of the geologists, which could result in inconsistency (Acosta et al., 2020). Drill-core geochemical data are frequently utilized to find elemental concentrations that act as stand-ins for commercially viable mineral deposits in order to improve mineral mapping.

Hyperspectral (HS) imaging has become a powerful tool in the mining sector in recent years, enabling rapid, non-invasive scanning of drill-core samples. Modern HS systems can process a full core tray of approximately five metres of core in seconds, generating detailed mineralogical information (Tuşa et al., 2020). However, the large volume and high dimensionality of HS drill-core datasets make manual interpretation inefficient and subjective, particularly when data are processed in segmented intervals along the core (Zhou et al., 2025). The large quantity of spectral bands, minute spectral variations across minerals, and the requirement to continuously connect mineralogical variations along the core provide challenges that might overwhelm human analysers and cause inconsistencies (Bioucas-Dias et al., 2013).

These characteristics create a strong need for automated, data-driven analytical approaches. Machine learning (ML) methods have therefore increasingly been adopted to address the analytical challenges posed by HS drill-core data. Supervised ML algorithms can efficiently exploit spectral patterns to improve mineral identification accuracy and consistency (Contreras Acosta et al., 2021). Nevertheless, drill-core applications remain constrained by difficulties in recognising subtle mineral classes and by the limited availability of representative labelled samples for model training (Acosta et al., 2020). Consequently, integrating HS imaging with ML techniques offers a pathway toward more objective, scalable, and reproducible drill-core mineral mapping, improving ore deposit characterisation while reducing reliance on subjective visual interpretation.

1.2. Significance of Accurate Drill-Core Mineral Mapping

Determining mineralization and alteration paragenesis, identifying the distribution and type of ore minerals, and comprehending how mineralogy affects ore processing all depend on an understanding of mineral distribution within a drill-core (Barker et al., 2021). Because drill-cores offer the material base that accurately depicts the conditions underground, their extraction is an essential stage in the exploration of mineral deposits. When skilled geologists visually log drill-cores, they often extract geological information such as lithology, mineralogy, mineral assemblages, alteration patterns, structures, and mineralization zones (De la Rosa et al., 2021). In addition, mineral mapping helps to identify various types of minerals present in drill-core samples, enabling the location and measurement of features like veins, which can be used to determine a value of the sample and various conditions it has encountered over time (Fabre et al., 2018). Drill-core mineral mapping is used to determine whether an area is valuable for

mining, whether minerals are abundant at the site, and what potential challenges might arise from future exploration.

1.3. Enhanced HS Imaging for Mineral Mapping

In the past decade, HS imaging has become an innovative method for quick, non-invasive, and non-destructive examination of drill-core samples, especially for mineral mapping (Contreras Acosta et al., 2021). This method permits accurate identification and spatial mineral mapping based on their reflectance or emissivity signatures by capturing detailed spectral information across a broad range of wavelength (Kruse et al., 2012). The integration of resolution enhancement techniques, especially supervised mineral mapping procedures, is one of the major developments in HS imaging. More precise and explicit identification of mineral assemblages is made possible by these improvements, which enable more thorough analysis of the matrix and vein structured within the cores (Contreras Acosta et al., 2021).

Despite its benefits, HS imaging has drawbacks that can compromise mapping accuracy and computing efficiency, such as high data dimensionality and noise susceptibility (Van der Meer et al., 2012). In order to address these problems, current research has investigated spatial enhancement methods such as combing high-resolution red, green, and blue imagery with HS data. Fine-scale geological features like minor intrusions and mineralized zones can be identified more easily due to this enhanced integration of spectral and spatial resolution (Duan et al., 2020).

Comprehensive information regarding physical characteristics and chemical composition of geological materials can also be obtained by HS imaging (Krupnik et al., 2019). More accurate quality evaluations are made possible by subtle changes in spectral patterns that can

demonstrate elemental substitutions or mineralogical variances, such as variations in the amount of magnesium or silica. Engineers can pinpoint high-grade areas, improve blasting methods, and reduce unnecessary material transportation by incorporating HS imaging into mining operations. Consequently, this enhances overall profitability and resource efficiency (Kirsch et al., 2018).

1.4. The Role of Geochemical Data in Identifying Minerals

Understanding the geological processes that have shaped our planet requires the examination of elements and compounds found in rocks, soil, or water. It offers important hints regarding the types of minerals present, where they are distributed, and any potential changes that have taken place throughout time. Alteration are specific chemical changes that occur in the surrounding rocks during the formation of a mineral deposit and if properly interpreted, these changes leave behind unique signals that may help identify mineral-rich areas (Valls, 2023).

According to Grunsky et al. (2017), geochemical data are obtained from a variety of sampling media (such as soil, till, regolith, lake sediments, stream sediments, and bedrocks) gathered at spatial scales appropriate for the geological/ geochemical processes under investigation. Finding certain anomalies connected to mineral deposits is often the aim of geochemical surveys; nevertheless, achieving this goal comes with challenges. First, the emplacement of mineral deposits results in unique multi-and uni-element geochemical signatures. Therefore, identifying geochemical signatures that characterise mineral deposits and ore-forming processes is crucial (Parsa et al., 2023). The goal of geochemical surveys is to give a spatial geochemical description of the principal geochemical processes or general geology as they appear in the medium being sampled. Most sampling media use a distinction between a fine-grained (<63 μm) and coarse-grained (usually >63 μm and < 2mm) fraction. In numerous

instances, course-grained material can be regarded as locally generated particles or minerals that have not been subjected to chemical dissolution, comminution, or weathering (Grunsky et al., 2017, 2020).

In mineral mapping and tectonic research, lithogeochemical data which include major oxide and trace elements, are widely utilized to classify different types of rocks, detect chemical differences brought on by trends in fractionation, and describe tectonic settings (Harris et al., 2000). It is difficult to identify the type of buried deposits using surface methods due to poor recognition, which makes mineral exploration difficult. However, in order to detect both buried mineral deposits and sub-outcropping, geochemical data of bedrock and till may offer useful targeting criteria (Puchhammer et al., 2024).

By showing compositional patterns along drill cores and differentiating minerals with similar spectral signatures, geochemical data offer quantitative elemental information that enhances HS imaging (Nikonow et al., 2019). Geochemical measurements enhance the precision of drill-core mineral mapping, dependability, and interpretability when combined with HS data, facilitating improved resource assessment and exploration choices.

Target area reduction and the likelihood of discovering mineral deposits are greatly enhanced by the analysis of geochemical data. As mineral exploration continues, exposed and/or shallowly buried deposits might no longer be sufficient to sustain the fast development of the social economy (Wang et al., 2023).

1.5. Integration of Deep Learning Models

Deep learning (DL) algorithms have become remarkably popular for identifying minerals in images of rocks. DL algorithms are now widely used for automatic classification and image analysis in several Earth science fields, including palaeontology, well-log analysis, geophysical imaging, and more. The efficacy of DL algorithms in image analysis is primarily due to their rapid acquisition of complicated visual representation and patterns (Dell'Aversana, 2023). In order to identify complex patterns and correlations that are challenging to manually specify, DL models automatically learn and extract features from the data. This differs with the conventional image-processing methods that depend on features that are manually generated. Numerous researchers studied at the use of DL in mineral mapping, micro-scale imaging of quartz and resin, and geochemical mapping (Liu et al., 2019). It demonstrates how excellent DL models are in mapping minerals and processing and analysing complex image data. By integrating DL models, geologists may develop mineral maps, design targeted and successful exploration programs, and make well-informed decisions about mineral resources by gaining thorough and precise information about promising zones (Hoseinzade et al., 2025). Compared to an individual DL model, a hybrid DL model can detect regions with significant uncertainty at boundaries of geochemical anomalies and can identify smaller areas as high-potential, resulting in more reliable and precise mineral potential forecasts, particularly when handling complex and heterogenous geochemical data.

1.6. Problem Statement

Resource exploration depends on accurate and efficient mineral mapping in drill-cores, yet conventional HS scanning techniques are costly, time-consuming, and error-prone. In addition, expert-dependent sample selection frequently places limitations on geochemical analysis, resulting in reduced constrained spatial resolution and potential biases. Recent researches have

shown that integrating HS imaging with geochemical data can improve mineral mapping, especially when integrated with DL models like convolutional neural networks (CNNs) and recurrent neural networks (RNNs). However, there are still issues with estimating uncertainty, enhancing precision, and maximizing computational effectiveness in drill-core mineral mapping. By using DL-based hybrid system that combines HS and geochemical data, this study aimed to overcome those obstacles and enhance prediction in mineral mapping. In particular, it assessed how well the algorithms operate together to improve the accuracy and provide a framework to quantify uncertainty in mineral predictions along drill-cores.

This study directly addresses the drawbacks of traditional drill-core analysis, such as low spatial resolution, possible biases, and labour-intensive procedures, by utilizing a deep learning–based hybrid approach that integrates HS and geochemical data. By doing this, it assesses the combined prediction performance of CNN and RNN models and offers a framework for measuring uncertainty, making drill-core mineral mapping more precise, effective, and impartial.

1.7. Research Aim and Objectives

This research aimed to integrate drill-core hyperspectral and geochemical data in order to improve the predictive performance of drill-core mineral mapping. The aim of the research was supported by the following objectives:

1. To quantify uncertainty in drill-core mineral mapping using DL approaches applied to integrate HS and geochemical data.
2. To evaluate and compare the effectiveness of CNNs and RNNs in accurate identification of minerals, using the integration of HS and geochemical data in drill-core analysis.

3. To develop and evaluate a hybrid system that combines CNNs and RNNs for enhanced mineral identification using drill-core HS and geochemical data.

1.8. Research Questions

This study aimed to address the following questions:

1. How can uncertainty be quantified after the application of CNNs and RNNs to integrated HS and geochemical data for mineral mapping?
2. Which DL models, CNNs or RNNs, provide the most accurate mineral identification when applied to HS and geochemical data integration in drill-core analysis?
3. How can CNN and RNN models be combined to create a better system for mineral mapping prediction?

1.9. Methodology

Figure 1.1 shows the steps taken for the DL method followed in this research to integrate HS and geochemical data for predictive drill core mineral mapping.

In the training phase, the dimensionality of HS data was reduced using PCA, and geochemical data were transformed using centred log-ratio (CLR) and after transformation, PCA was applied to the CLR-transformed geochemical data. During this phase, HS and geochemical data were used as predictor variables, and mineral abundance was used as the target variable, and a DL model was trained to predict mineral compositions using both data types.

The prediction phase involved the use of DL trained into a new drill-core data to predict mineral abundance by assigning labels to an unknown spectral data. The last step was to validate the model using the metrics mean absolute error (MAE) and coefficient of determination (R^2) for measuring model performance.

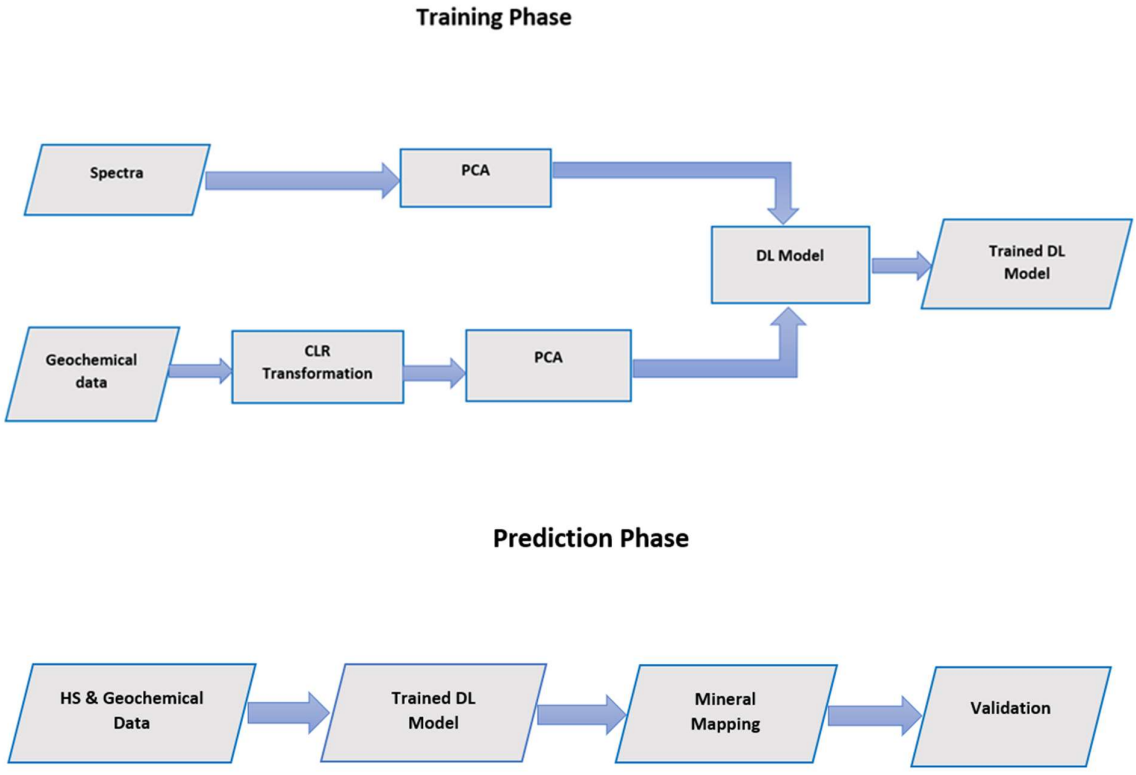


Figure 1.1. Flow diagram for the DL method to integrate HS and geochemical data for drill-core mineral mapping.

1.10. Thesis Structure

- The **introduction** chapter establishes the fundamental framework for the whole study.
- The **literature review** chapter is a comprehensive overview and analysis of the body of the research on the chosen topic. It highlights gaps or limitations in the existing knowledge, highlight previous research, and assist in positioning the study within the academic conversation.

- The **methodology** chapter explains and justifies the methods used to conduct this research study, detailing the analysis techniques to answer the research questions.
- The **hybrid CNN-LSTM implementation and model testing** chapter shows the results of all the models used in this study during the testing stage.
- The **discussion** chapter explains the meaning, significance, and implications of the research result, while connecting them to the research questions and existing literature, and acknowledging the limitations.
- The **conclusion and recommendation** chapter summarizes the whole research, states the key findings, provides answers to the research questions, and suggests actionable steps for future research and improvements based on the study.

CHAPTER 2: LITERATURE REVIEW

One essential step in the mining exploration industry and discovery of mineral resources is the analysis of drill-core samples (De la Rosa et al., 2021). Direct and physical observation of rock characteristics are possible using conventional drill-core analysis methods such as visual logging, manual lithological and mineralogical descriptions, geotechnical evaluation, and core sampling. In the past, these techniques have been essential for determining the distribution of minerals and comprehending the underlying geological formations (De la Rosa et al., 2021). Although these methods were essential, they have lots of drawbacks, especially when it comes to mineral mapping processes. Because visual evaluations are subjective by nature, standardization is challenging and it leads to observer variability. Additionally, manual logging is labour-intensive, time-consuming, and prone to human error, particularly when dealing with significant amounts of drill-core (Harraden et al., 2019).

The above-mentioned traditional techniques frequently fail to provide the necessary spatial resolution and compositional accuracy as the mining industry shifts toward more data-driven and automated approaches to resource modelling and mine planning (Zuo et al., 2021). In order to quantify mineral assemblages, high-resolution analytical techniques like scanning electron microscopy combined with software for liberation analysis are employed (Fandrich et al., 2007). These sophisticated techniques provide in-depth mineralogical insights, but they are expensive, time-consuming, destructive, and usually based on a small number of samples, which reduces the spatial resolution of data (De la Rosa et al., 2021).

In this regard, geochemical assays are essential to contemporary mineral mapping. Major, minor, and trace elements in rock samples are quantitatively measured by these assays and are usually expressed as weight percent or parts per million. Because they are inherently compositional data, they are crucial for directing mine planning, assisting with environmental evaluations, and defining ore bodies as well as estimating resources (Grunsky and Caritat, 2020). However, geochemical assays have drawbacks as well: they are time-consuming, frequently represent average compositions across relatively length drill-core intervals, which might obscure fine-scale lithological or mineralogical variations (Zekri et al., 2025). Notwithstanding these difficulties, new research shows that it is possible to extrapolate results of geochemical assays to whole drill cores, allowing for more continuous, high-resolution analysis of subsurface geology (Acosta et al., 2020). Therefore, the resolution mineral mapping models can be greatly improved by carefully combing geochemical data with mineralogical tools.

2.1. Hyperspectral Imaging/Data in Mapping Minerals

HS imaging has been used in many different geological applications since it was first introduced in the 1970s. Geologists have developed a number of methods over the last few decades to analyse HS image data and statistically extract geological information from high-spectral-resolution remote sensing images (Peyghambari and Zhang 2021). By recording the spectral signature of every pixel in an image, HS imaging in geological research aims to identify certain materials. The spectral information is then used in mineral mapping as it is an important method for locating, categorizing, and spatially distributing materials within a drill-core (Sudharsan et al., 2019).

Nonetheless, atmospheric interference, such as the scattering and absorption of solar radiation by gases and aerosols, affects HS data. To achieve precise quantitative remote sensing, these atmospheric influences must be corrected during the pre-processing stage. Additionally, dimensionality reduction methods like principal component analysis (PCA), minimum noise fraction, and independent component analysis are frequently used to highlight the most important spectral information for mineral mapping, because of high dimensionality of HS data ((Peyghambari and Zhang 2021).

The use of HS imaging in mineral mapping is a basic application. The advantage of speed and wide coverage over large and frequently inaccessible areas are provided by airborne HS survey (Guha, 2020). By producing precise mineralogical maps quickly, these surveys can drastically cut down on the time and expense involved in traditional integrated studies. Geologists use these maps to evaluate the mineral potential of specific areas and find promising exploration prospects at an early-stage pf prospecting (Bedini, 2017). According to Krupnik and Khan (2019), HS images also support improved planning analysis of possible mining operations, including determining the best extraction techniques and logistics for transporting resources from the site to processing facilities.

Despise the advantages of HS images, it is difficult to precisely identify mineralized surface features from HS datasets. In order to increase the precision of surface mineral mapping, it is frequently necessary to integrate other relevant datasets (Kruse, 2021). Although there are commercial software tools available for processing, validating, and calibrating HS-based mineral maps, additional customization is often required. This is especially true when using novel algorithms that mathematically model the spectral behaviour of complex backgrounds in

order to improve target detection within pixel-level spatial extents. Enhancing the spectral contrast between mineral target and surrounding materials is the goal of these sophisticated algorithms. In order to extract valuable geological information for mineral mapping, a number of pre-processing procedures are carried out after HS data have been correctly calibrated and spectral integrity from ground to an image has been confirmed (Guho, 2020).

2.2. Overview of Deep Learning Algorithms

The class of ML algorithms known as DL has demonstrated remarkable performance when used on complex and unstructured datasets (Sarker, 2021). DL has potential capacity to automatically extract high-level representation from complex data, which made it a significant and "hot" topic in fields like mineral mapping in recent years. These techniques are ideal for geological applications because they allow computational models to learn characteristics gradually at various levels of abstraction (Xiong et al., 2018). DL has the potential to make a substantial contribution to mineral exploration and geoscientific analysis as the volume of geoscience data is growing at a rapid pace.

In essence, DL is a type of representation learning where models are built to automatically find patterns required for tasks like segmentation, detection, and classification without the need for human feature engineering (Zuo et al., 2019). The number of layers through which data passes is referred to as "deep" and each layer converts the input into progressively more abstract representation (Mathew et al., 2021; Sarker, 2021)

In general, supervised and unsupervised learning algorithms can be used to classify DL models (Xu et al., 2021). CNNs, deep autoencoders, long short-term memory (LSTM), generative

adversarial networks, gated recurrent unit models, and variational autoencoders are a few of the DL models that are frequently utilized in geoscience (Zuo and Xu, 2023). Because these algorithms can handle complex and nonlinear interactions in huge datasets, they have proven very useful in the field of geoscience. This has improved the detection of geochemical anomalies and helped uncover hidden or subtle pattern (Mathew et al., 2021). For instance, even in difficult settings like areas covered in forests, DL techniques have been effectively used for mineral prospectivity mapping. However, the type and characteristics of the input data affect how well various DL models perform (Xu et al., 2021).

The computational cost is a crucial factor in DL applications. Due to large number of parameters involved, training a DL model can take a long time, particularly when processing large datasets (Sarker, 2021). DL models have the advantage of significantly faster interference times than classic ML algorithms once trained, despite the fact that they typically demand significant computational resources and enormous volumes of labelled data to develop successful data-driven models (Liu et al., 2022). However, DL models may perform poorly with small datasets because there is not enough data to identify generalized patterns (Chen et al., 2020). In spite of these drawbacks, DL has demonstrated remarkable efficacy in the classification of HS images. Several studies have shown how DL models can achieve mapping and classification accuracies of over 90% and, in certain cases, over 95% (Zhang et al., 2022). These findings show how DL has the ability to completely transform mineral mapping by offering precise, automated interpretation of high-dimensional geoscience data.

2.2.1. Artificial Neural Networks: Convolutional Neural Networks and Recurrent Neural Networks

CNNs are a popular class of DL algorithms that are particularly well-known for their effectiveness in computer vision and image classification problems (Wang et al., 2024). The ability of CNNs to automatically learn hierarchical feature representations straight from input data, without the need for manual feature engineering, has made them popular in geoscience applications like mineral mapping. With convolution operations at each layer, they are therefore very good at obtaining profound and significant spatial patterns (Li et al., 2021 and Sarker, 2021).

From a structural standpoint, a CNN is a feedforward neural network in which information moves in a single path, without circles, from one input to output. Convolutional layers, activation functions (non-linear mappings), and pooling layers are commonly included; these layers work together to minimize model complexity while maintain important features (Agrawal and Govil, 2023). CNNs take advantage of local connection, shift invariance, and spatial compositionality by treating input images as sets of regular pixel patches in Euclidean space. These features allow CNNs to effectively capture local features and identify spatial anomalies related to mineralisation (Zuo and Xu, 2023).

CNNs do not rely on manually created features like traditional mineral mapping methods do. Rather, they use spatial patterns to directly generate and train convolutional features. They are greatly supported by this when it comes to identifying significant representation in complex and multivariate geospatial datasets. CNNs can better model associations between image pixels and mineral classes by optimizing feature extraction during training (Agrawal and Govil,

2023). Numerous CNN architectures, including LeNet, have found successful applications in the field of mineral mapping. Convolution layers for feature extraction, pooling layers for dimensionality reduction, and fully connected layers for final classification are the three main parts of a typical CNN (Li et al., 2023; Zuo, 2024). CNNs' inherent spatial structure may cause them to ignore topological correlations of data over time.

When it comes to learning from input data, RNNs provide a compelling alternative to overcome these limitations. RNNs are specifically useful for tasks involving temporal dependencies, such as predicting time series or modelling sequential natural evens. They are designed for sequential, or time-series data and process input sequence using internal memory, feeding the output from previous step as input to the current step, allowing the model to retain and utilize contextual information from earlier data points (Sarker, 2021; Qaderi et al., 2025).

Simple RNNs, gated recurrent units, and LSTM networks are examples of RNN variations (Qaderi et al., 2025). Due to their capacity to analyse contextual relationship within sequential datasets, these frameworks are becoming more and more popular in geoscientific applications (Lipton et al., 2015). In mineral mapping, RNNs provide the following advantages: (1) activation functions for non-linear model fitting; (2) recurrent connections for learning variable dependencies; and (3) deep representative learning capabilities that can yield new information to direct mineral exploration (Yin et al., 2022).

LSTM networks tend to be popular among RNNs because they can solve the vanishing and exploding gradient issues that arise during training. LSTM units are made up of memory cells with long-term information storage capabilities that are managed by input, output, and forget

gates (Sarker, 2021). This design makes LSTM models appropriate for deriving deeper insights from geochemical time-series data since it retains contextual information and models long-term dependencies within sequences. By maintaining connections between hidden layers across time steps, LSTMs maintain memory states and correlations between sequential inputs, in contrast to CNNs, which process data in a simple feedforward manner (Tian et al., 2018). LSTMs are very good at interpreting geochemically significant sequences because of their chain-like structure, which allows previous outputs to impact present and future predictions. They are especially useful for evaluating genesis-relevant mineral indicators where context over time or depth is critical, and they complement CNNs in mineral mapping by modelling the temporal evolution of geochemical anomalies (Wang et al., 2024).

2.3. Ensemble Learning

ML encompasses both ensemble and single-learner learning models. In single-learner models, tasks like regression and classification are carried out by a single, robust algorithm. The powerful ML technique known as ensemble learning, on the other hand, integrates the outputs of several models using specific combination strategies and combines them during training stage. This method frequently outperforms individual models in terms of performance (Guo et al., 2020). In ML, ensembled learning has become one of the most sophisticated and successful techniques. It successfully reduces bias and variation by training several weak learners and combining their predictions, as a result, the accuracy, robustness, and reliability of the model are enhanced (Zhang et al., 2024). Ensemble approaches are major focus of current research since they also aid in addressing common issues in traditional ML, such as high sensitivity to training data, overfitting, and computing complexity.

Bagging, boosting, voting, and stacking are a few popular ensemble learning methods (Yin and Li, 2022). These can be categorized into two categories: heterogenous and homogeneous ensembles. Heterogeneous ensembles, like stacking and boosting, use different kinds of base learners, whereas homogenous ensembles, like bagging and voting, use identical base learners (Zhao et al., 2024). It has been demonstrated that these ensemble approaches improve generalization ability, successfully reduce model variance, and produce excellent results in real-world applications. Numerous fields, such as geoscience, biology, engineering, medicine, computer vision, and image processing, have made an extensive use of ensemble learning (Mienye and Sun, 2022). Nonetheless, its use in the exploration of mineral resources is still restricted. Even though data-driven approaches are quite accurate at prediction and classification tasks, they frequently lack ability in capturing physical mechanisms and integrating domain-specific information. In practical situations, this limitation may result in restricted generalization capabilities (Li et al., 2024).

The application of stacking ensemble model for mineralization prediction is an example of ensemble learning in this field of study. This model uses logistic regression (LR) as the meta-learner, and random forest (RF), extreme gradient boosting (XGBoost), and adaptive boosting (AdaBoost) as the base learners. When the performance of an ensemble model was compared with that of an individual models (RF, XGBoost, AdaBoost, and LR), the former performed noticeably better than the latter in terms of accuracy and stability (Yu et al., 2024). In particular, ensemble learning approaches can use a variety of data sources and featured in mineral mapping to generate predictions that are more accurate and reliable (Zhang et al., 2024).

2.4. Prior Research on Integration of Geochemical and Hyperspectral Data

HS scans of drill-cores are generally expensive, time-consuming, and prone to error when used for mineral mapping. In addition, the location chosen for geochemical investigation is constrained and subject to the expertise and experience of geologists (Kutzke et al., 2022). Nevertheless, Kutzke et al. (2022) showed that, even with small geochemical datasets, by integrating sparse geochemical observation with HS drill-core-scan-derived mineral abundances in mineral mapping, it is possible to make good geochemical predictions of elemental geochemical concentrations along hundreds of kilometres of drill-cores.

Acosta et al. (2020) integrated geochemical and HS data for drill-core analysis using RF, support vector machine (SVM), and composite kernel support vector machine (ckSVM). The accuracy of ckSVM was the greatest, followed by RF and SVM; RF outperformed SVM because of its flexibility when it comes to processing various data sources. For HS data classification, Chen et al. (2014) developed a spectral–spatial classification framework with a stacked autoencoder (SAE). Compared to more conventional techniques like PCA and nonnegative matrix factorisation, these features help classification models like logistic LR and SVM to become more accurate, resulting in SAE-extracted features with increased classification accuracy. By including spatial context, the SAE–LR combination can improve the accuracy of mineral mapping even further. Compared to SVM, SAE–LR tested more quickly, but it took longer to train.

In order to identify minerals, Okada et al. (2024) used HS imaging and ML, concentrating on the visually identical minerals anorthite and albite. These minerals are spectrally similar and difficult to distinguish using conventional imaging alone, making them an ideal test case for

data integration approaches. They integrated HS and geochemical data to facilitate mineral identification using 34 ML approaches, such as decision tree (DT) and SVM, using a MATLAB-based HS analysis tool. The use of multiple algorithms allowed the authors to systematically compare classification performance and assess the robustness of different modelling strategies. Mineralogical data with high resolution obtained from thin, polished sections of the same drill-core samples, which were around 30 μm thick, were combined by Acosta et al. (2019) with HS data. These thin-section analyses provided ground-truth mineralogical labels for training and validating predictive models. The final method of integrating HS data with high-resolution mineralogical data from mineral liberation analysis and scanning electron microscopy (SEM) produces comprehensive maps and provides quantitative data regarding the relative abundance of every mineral found in the samples, thereby enhancing the geological interpretability of HS predictions and enabling deposit-scale mineralogical characterization.

In the Goldfields-Esperance region of Western Australia, Choros et al. (2022) investigated whether ore and waste discrimination might be possible at the excavation site by integrating HS imaging with geochemical analysis of samples from a gold mine. To analyse the data, Choros et al. (2022) used spectral angle mapper (SAM) and CNN. This integration enhances the accuracy and efficiency of mineral mapping. CNN model fared better than the SAM in discriminating between waste and ore at the point of excavation because CNN demonstrated a very strong resistance to change in orientation and shadowing. However, the CNN's accuracy decreased in comparison to gold concentration reference values from the samples at the scanned mine face when finer resolution analysis was performed to differentiate between lower and higher ore grades. Overall, the study concluded that integrating geochemical data and HS

imaging makes it possible to discriminate between ore and waste during excavation with the help from ML algorithms (Choros et al., 2022).

Galdames et al. (2022) showed that integrating geochemical data with HS images may be used to classify rock lithologies based on the geochemical characteristics of rocks. The integration of geochemical data from 13 different lithological classes with HS images effectively carries out instance segmentation by using CNN to minimize dimensionality of HS image data, precisely delineating edges and detecting multiple instances of objects belonging to the same class (Galdames et al., 2022). In this context, dimensionality reduction refers to compressing the hyperspectral feature space while retaining the most discriminative spectral information, thereby improving computational efficiency and reducing noise in the classification process. This integration offers a more robust, automated way of mineral mapping by integrating spectral and chemical characteristics to improve classification precision, edge delineation, and segmentation, demonstrating how multimodal data fusion can overcome the limitations of single-data-source approaches in complex geological settings and enhance the reliability of lithological interpretation.

To ascertain ore grade and 2D geometry of mineralisation at the mine face, Lobo et al. (2021) investigated the integration of HS imaging with traditional ground-based sample data (geochemical assays) at the San Finx tin-tungsten mine. Effective mapping of ore grade and mineralisation geometry was made possible by integration of HS data and geochemical assays using ML methods, such as RF, SVM, and linear discriminant analysis. Among the techniques, RF achieved the best accuracy (Lobo et al., 2021).

In order to create HS models for the extraction of mineralization-indicative mineral information, Lin et al. (2022) integrated the sparrow search algorithm (SSA) with random RF and gradient boosting decision tree (GBDT). Because of its high ability to optimize globally and its low sensitivity to initial settings, SSA was selected. For mineral extraction, limonite and chlorite geochemical data were integrated with HS data. Both SSA-RF and SSA-GBDT identified known deposits and fault structures with comparable, precise results. Compared to SVM, the combined models demonstrated improved generalization, increased stability, and more successful changed mineral extraction (Lin et al., 2022).

Zhao et al. (2020) used hierarchical spatial–spectral and LSTM (HSS-LSTM) to integrate geochemical and HS data from the Nevanda mine. This method outperformed LSTM, 3D-CNN, and SVM in mineral identification. In order to find minerals that are invisible by conventional HS interpretation, Rotem et al. (2023) integrated HS data with geochemical and SEM-based automated mineralogical data using the SisuChema imaging system and found that the accuracy of mineral mapping is increased by integrating spectral, geochemical, and mineralogical data with ML models such as CNN. HS images and geochemical data were effectively integrated by Okada et al. (2020) to automatically identify minerals prior to mineral processing. They found that CNNs were more accurate than red, green, and blue imaging when applied to five minerals; CNNs provide flexibility in determining crystal size, and this method improves mineral mapping.

Tusa et al. (2020) automated the assessment of mineral abundance in diamond drill core samples by integrating HS data from unpolished core halves with quantitative geochemical data from thin sections using RF, SVM and neural network regression model. All

tested ML methods produced good results, but RF was more resilient to sparse and imbalanced training data. Rodger et al. (2021) identified lithologies, alteration patterns, and weathering impacts by integrating geochemical and HS data from drill cores in the Gawler Craton, South Australia. They found that, because hierarchical density-based clustering (HDBSCAN) can accommodate different density regions, include noise, and function without specified cluster numbers, it was more effective than DBSCAN.

Previous studies, such as by Kutzke et al. (2022), have shown the efficacy of integrating geochemical data with HS data for geochemical predictions of elemental geochemical concentrations across hundreds of kilometres of drill core using CNN. Data integration enhances mineral mapping by providing a cost-effective, high-resolution approach to map weathering, alteration, and mineralogical gradients. However, as noted by Laukamp et al. (2023), this technique works best only when the mineral assemblages in drill cores are relatively similar to one another. Hamedianfar et al. (2023) investigated ensemble methods, including bagging, AdaBoost, RF, LightGBM, and GBDT, as well as the DL model ENVINet5, for semi-automated mineral mapping by integrating geochemical data and HS imaging from the drill cores at the Hirvilavanmaa study site. GBDT and LightGBM achieved the highest accuracy. Agrawal (2023) applied ML algorithms, including SVM, K-NN, ELM, DT, XGBoost, RF, and artificial neural network (ANN), to integrate geochemical and HS data for mineral mapping in Jahazpur, Rajasthan, India; they found that ANN achieved the highest accuracy as compared to other ML methods.

In the field of geoscience, uncertainty quantification has grown in significance as a part of machine learning and deep learning applications, especially when projections guide high-risk

exploration choices. Aleatoric uncertainty, which results from noise or variability in the data, and epistemic uncertainty, which reflects limited knowledge of the model and can be decreased through better training or larger datasets, are the two primary types of uncertainty that are typically distinguished ((Hullermeier and Waegeman, 2021). Previous research has included uncertainty estimation using methods that enable the quantification and spatial mapping of prediction variability, such as Monte Carlo dropout, ensemble modelling, Bayesian neural networks, and repeated cross-validation (Davis et al., 2020; Puzyrev and Duuring, 2025).

Uncertainty metrics have been employed in mineral exploration settings to enable risk-aware targeting strategies, direct follow-up sampling, and identify low-confidence prediction zones (Zuo et al., 2021; Yousefi et al., 2024). In order to improve the interpretability and practical usefulness of DL-based mineral mapping results, more recent work has focused on explicitly incorporating uncertainty into prospectivity models using confidence indices, Bayesian frameworks, or hybrid ensemble approaches. When taken as a whole, these studies show that uncertainty analysis is not only a diagnostic tool but also an essential part of assessing model reliability and assisting with decision-making in resource assessment and drill-core mineral mapping.

2.5. Concluding Remarks

The accuracy of mineral prediction, resolution, and the effectiveness of mineral identification and classification have all been shown to increase with the integration of HS and geochemical data in mineral mapping. It is evident from the literature that a more reliable interpretation of the mineralogical and geochemical variability in drill-cores can be achieved by integrating this complimentary data with ML and DL algorithms. By using both chemical and spatial data,

techniques like CNN, RF, SAE, and sophisticated ensemble methods (GBDT and LightGBM) outperformed previous (traditional) models and made it possible to map minerals, estimate ore grades, differentiate between ore and waste, and discover subtle mineralogical features. CNN-based models are particularly effective at extracting spatial–spectral patterns from hyperspectral imagery, whereas tree-based ensembles such as RF and GBDT offer greater interpretability and robustness to noisy geochemical inputs; SAE excel at dimensionality reduction but may require careful tuning to avoid loss of geological detail, while boosting frameworks such as LightGBM often achieve high accuracy at the cost of increased model complexity. According to the reviewed literature, integrating geochemical and HS data using DL provides a flexible, automated, and affordable method of high-resolution mineral mapping that may be applied to both active mining operations and exploration, although several studies also note challenges related to data availability, transferability between deposits, and uncertainty quantification, highlighting the need for hybrid architectures and probabilistic modelling strategies in future work.

CHAPTER 3: METHODOLOGY

3.1. Case Study Area

The Rocklea Dome channel iron deposit is located in Western Australia, Hamersley province (Figure 3.1), which is the main export source for iron ore from Australia (Laukamp et al., 2021; Haest et al., 2012). Minor mafic and ultramafic dykes have intruded into the Archean-age biotite monzogranite pluton that forms the core of Rocklea Dome. The Pilbara Craton, which forms the foundation of the Hamersley province, includes these igneous rocks (Laukamp et al., 2021).

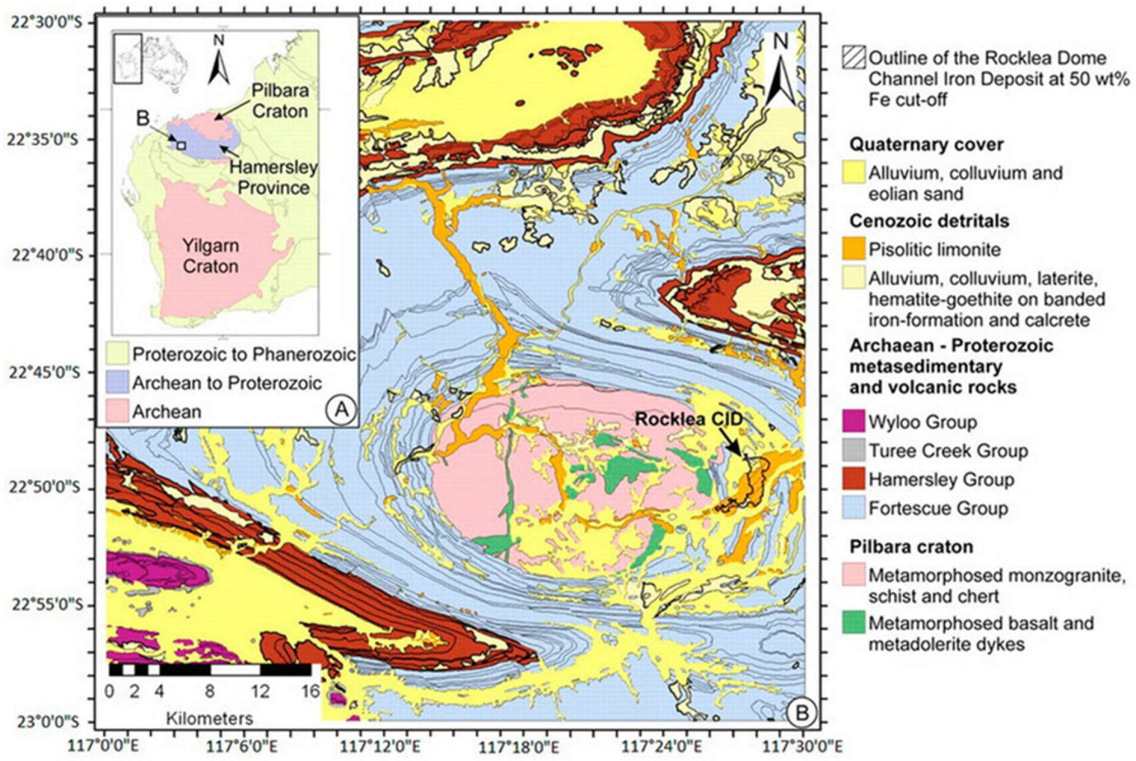


Figure 3.1. (A) Map of Western Australia showing the location of Rocklea Dome. (B) Rocklea Dome geologic map showing the Rocklea Dome channel iron deposit (CID) (from Haest et al. 2012).

Archean greenstone assemblages, such as metabasalts, schist, chert, as well as mafic and pegmatite dykes, encircle the central pluton. These rocks are unconformably covered by the sedimentary and volcanic successions of the Hamersley basin, which date back more than 2765 million years. Units of the Fortescue, Hamersley, Turee Creek, and Wyloo Groups make up the surrounding strata; the Hamersley Group is especially noteworthy because it is the main host of bedded iron ore deposits (Haest et al., 2012; Laukamp et al., 2021).

A Tertiary-aged paleochannel, which cuts into the weathered Hamersley surface, crosses the Rocklea Dome. Pisolitic limonite (15-45m thick), calcrete, and detrital sediments are among the iron-rich materials found in this paleochannel, which was filled between the middle Eocene and middle Miocene (Haest et al., 2012). Channel iron deposits, which are found in these sediments, are mostly made up of pelletoids that are rich in iron (oxyhydr-) oxide and pieces of ferruginized wood that range in size from 1 to 10 mm. According to Haest et al. (2012), the pelletoids usually include non-structured iron oxyhydroxide granules and hematite or goethite nuclei surrounded by goethitic cortices.

3.2. Drill-Core Geochemical Data

The geochemical data used in this study were obtained from the Rocklea Dome 3D Mineral Mapping Test Dataset (Laukamp, 2021; https://doi.org/10.25919/5ed83bf55be6a). They comprise analyses of 486 RKC drill-holes and 14 RKD drill-hole with the longest drill-hole having the length of 109 meters deep down, and from these 486 RKC drill-cores, 30 drill-cores (1311 samples) were used.

Figure 3.2 below represents a plan-view of the drill-hole layout (blue and red dots) of Rocklea Dome whereas Figure 3.3 shows a plan view of the drill-holes (coloured in red) that were used in this study because they have less of censored values as compared to the drill-holes in blue colour from Figure 3.3, and these plan-views were generated using Micromine.

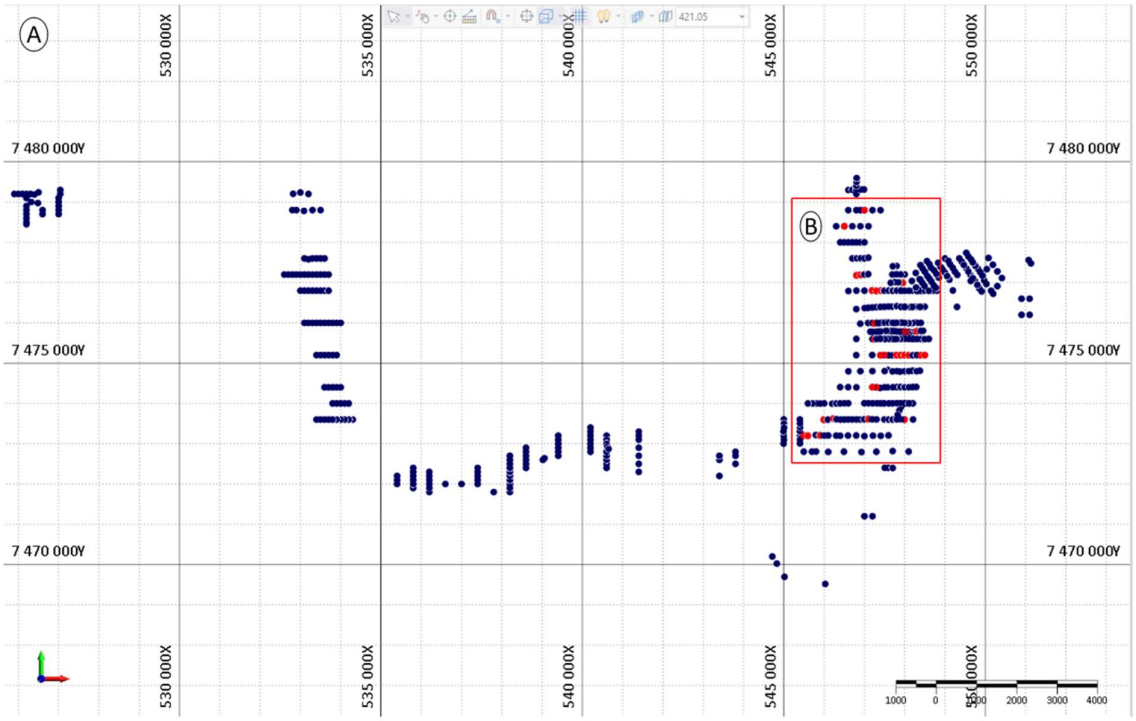


Figure 3.2. (A) Rocklea Dome drill-holes (blue and red dots) and (B) Figure 3.3 showing drill-hole used in this study highlighted in red.

The dataset included concentrations of the following elements: Fe, Fe_2O_3, P, S, SiO_2, Al_2O_3, MnO, Mn, CaO, K_2O, MgO, Na_2O, and TiO_2. However, Fe_2O_3, MnO and Na_2O data were excluded from the data analysis because they were mostly below their corresponding analytical detection limits (their whole columns in the data spreadsheet were filled with zeros). The decision to exclude them was taken because Reimann et al. (2022) recommended the exclusion of variables if censored (i.e., below detection limit) values exceed 25% of a dataset. Mineral abundance data were also present in the same file, written as *Fe_Ox_ai* representing hematite,

goethite, jarosite, and limonite; *kaolin* abundance having kaolin group minerals, namely kaolinite halloysite, dickite and nacrite; *wmAlsmai* representing abundance of white micas (e.g. illite, muscovite, paragonite, brammalite, phengite, lepidolite, margarite) and Al-smectites (montmorillonite, beidellite); and *carbai3pfit* representing carbonates (Figure 3.4).

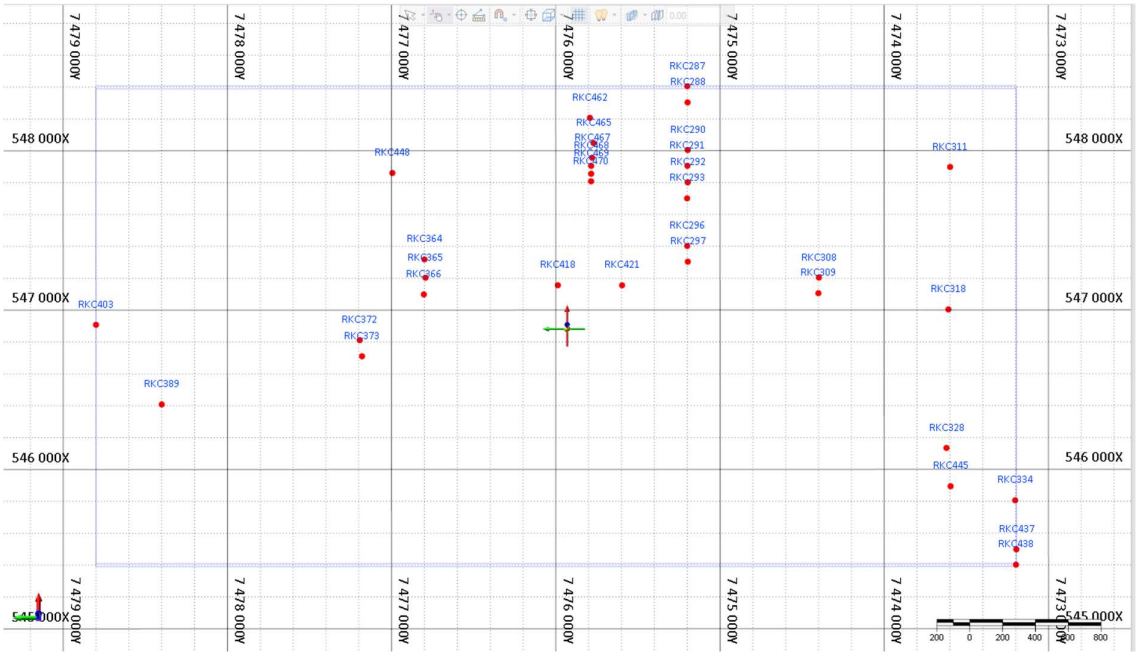


Figure 3.3. Data from drill-holes (in red colour) from Rocklea Dome were the ones used in this study.

For variables having censored values of less than 1%, Grunsky and Smee (1999) suggested substituting a constant value between 0.33 and 0.5 times the detection limit. However, this study chose to use the imputation strategy described by Abedini et al. (2023), as it was used recently and it statistically robust method when it comes to handling censored data compared to Grunsky and Smee (1999) method.

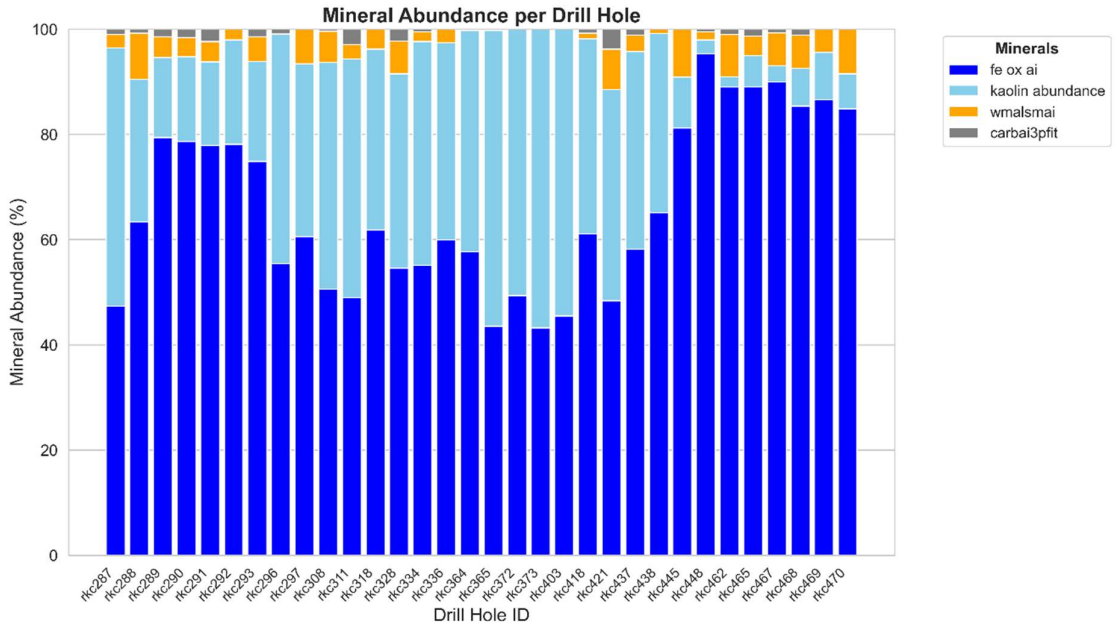


Figure 3.4. Mineral abundances per drill-hole

3.3. Drill-Core Hyperspectral Data

In this study: three primary data sources – RC_data.bip, RC_data.tsg, and Rocklea_ Assay_MMX.xlsx – were used to extract HS data from Rocklea Dome (Laukamp, 2020; https://doi.org/10.25919/5ed83bf55be6a) using Python software. Figure 3.5A shows the extracted hyperspectral (HS) dataset with 9 layers, 7520 samples, 531 channels, 91 params and 4 sets. In total, the HS data were composed of 7520 reflectance spectra for rock chip samples, and 66 853 reflectance spectra for diamond drill (RKD) core samples (Laukamp et al., 2021). From 7520 spectral bands, 1276 spectral bands that correlate with the geochemical data were extracted and were used in this study.

3.3.1. Hyperspectral Data Pre-processing

The ability to extract morphological characteristics (external features) and spectral properties (internal compositional attributes) of a sample makes HS imaging an effective tool for target classification. The practical use of HS images may be hindered by the presence of redundant

information, dark noise, and spatial-spectral dimension noise, which are caused by strong correlation between adjacent spectral bands (Li et al., 2021). Therefore, in order to minimize noise and enhance the quality of data, pre-processing is crucial before regression and mineral mapping.

Because HS bands are dense and continuous, processing these data is extremely difficult due to their high dimensionality. This redundancy may obscure important patterns in addition to increasing computational load. Dimensionality reductions methods are used to solve this problem, increasing processing time while maintaining important spectral data (Hajaj et al., 2024).

In general, dimensionality reduction techniques can be divided into two categories: supervised and unsupervised, as well as linear and non-linear. The most popular unsupervised methods are minimum noise fraction, independent component analysis, and principal PCA (Tripathi and Garg 2021). These techniques are frequently used to extract meaningful spectral bands or as a first step in classification workflows incorporating ML and DL algorithms. They are particularly efficient at boosting lithological properties from HS data (Hajaj et al., 2024).

In this study, the high dimensionality of the HS data was reduced using PCA implemented in Python. This approach transformed the original correlated variables into a smaller set of uncorrelated principal components that retained most of the variability in the dataset, ensuring that the most informative features were preserved for subsequent analysis (Figure 3.5B).

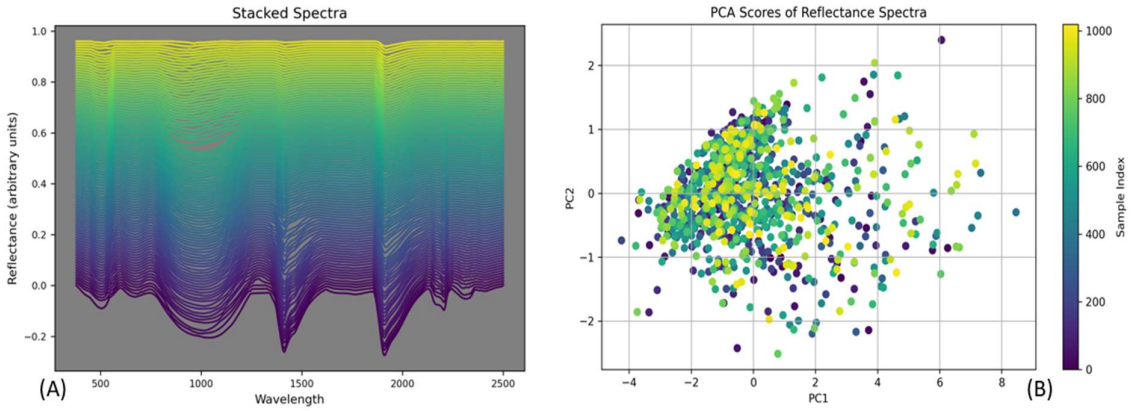


Figure 3.5. (A) Rocklea dome extracted stacked spectra for RKC holes, with wavelength and reflectance arbitrary units and (B) PCA of the extracted reflectance spectra showing transformation of original high-dimensional HS data into a reduced set of uncorrelated components.

3.4. Geochemical Data Pre-processing

3.4.1. Centred Log-Ratio Transformation

Log-ratio transformations are one of the most popular transformation techniques that have been developed for the analysis of compositional data. The main aim of log-ratio transformation is to produce a reasonably symmetric distribution that will provide a good basis for further statistical techniques and to convert data into an unconstrained space (West Robert, 2022). Standard statistical methods can be applied to compositional data, which are limited by a constant sum (typically 100%), according to these transformations. According to Mahdiyandar and Salimi (2022), there are three primary types of log-ratio transformations: isometric log-ratio (ILR), CLR, and addictive log-ratio (ALR). Real-valued variables with values ranging from negative to positive infinity are produced by converting the original components into log-ratio coordinates. These transformations make data to be suitable for multivariate statistical analysis.

In ALR transformation, one element of composition is chosen to act as the common denominator. After dividing each of the other components by the chosen component, the

logarithms of the resulting ratios are calculated. Although this approach is simple, it creates asymmetry by eliminating the reference element from further investigations. In addition, there is no defined theoretical foundation for the denominator selection, which is subjective and frequently based on domain expertise or individual judgement (Carranza, 2011).

In contrast, CLR transformation computes each the logarithm of each component in relation to geometric mean of all components. CLR requires all components to be in the same unit and, in contrary to ALR, keeps all variables in the modified dataset. The primary drawback of CLR transformation is that it produces unique (i.e., non-invertible) covariance matrix, which limits the application of some multivariate techniques. Because it retains all compositional information without dropping any variable, CLR is frequently chosen over ALR in spite of this drawback (Mahdiyanfar and Salimi, 2022; Khammar et al., 2021).

ILR transformation, where D is the number of original components, creates orthonormal coordinates in a (D-1) dimensional Euclidean space, so addressing the drawbacks of both ALR and CLR. This makes it possible to conduct a greater variety of statistical analysis and to compute an invertible covariance matrix. On the other hand, ILR transformations are more complex and require creation of orthonormal bases, which may complicate the interpretation of the outcomes. ILR is not suitable for univariant analysis because of its reduction in the number of dimensions (Graffelman et al., 2018).

In light of these considerations, CLR transformation was used in this study because it preserves all components in the dataset and works well with the multivariate analysis techniques used in this study.

Geometric Mean:

$$GM = \left(\prod_{i=1}^{D} xi \right)^{1/D} \qquad \text{.....equation 3.4.1}$$

$$CLR\ (x_i) = \ln \left(\frac{x_i}{GM} \right)$$

The geochemical data were processed using the CLR transformation prior to the application of DL techniques. The data was effectively transformed from a confined compositional space (simplex) into an unconstrained Euclidean space by removing the constant-sum constraint that is intrinsic to compositional datasets. By accomplishing this, the CLR transformation ensures that relationships between elements are meaningfully represented and permits effective multivariate statistical analysis by maintaining the data's true covariance structure.

Following transformation, inter-element correlations were evaluated (Figure 3.7) and the frequency distributions of elemental concentrations in the drill-core samples were analysed (Figure 3.6). PCA was used to further prepare the data for DL tasks and lower computational complexity. In order to reduce the dimensionality of the dataset and emphasize the most informative characteristics for further modelling, PCA processed the altered data into a new set of uncorrelated principal components while keeping the majority of the variance (Figure 3.8).

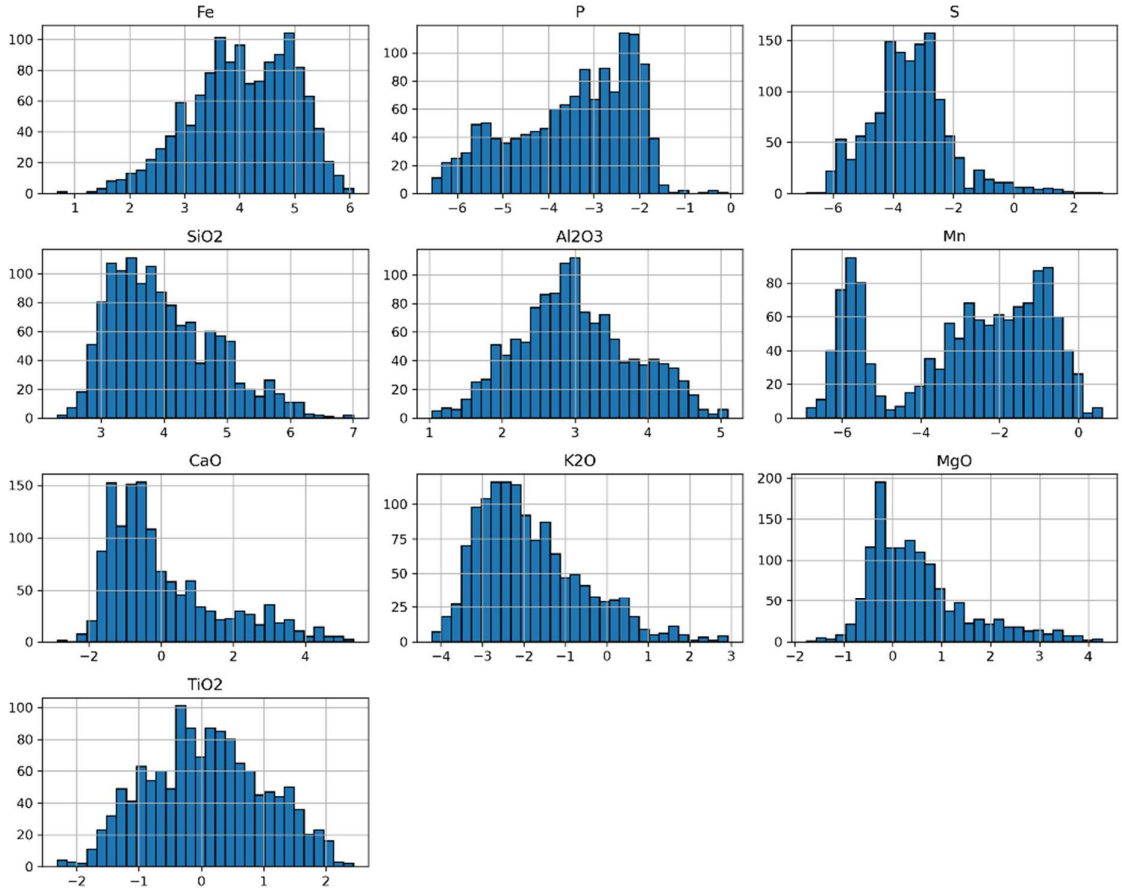


Figure 3.6. Frequency distributions of CLR-transformed element data from drill-core samples, highlighting differences in skewness and modality among the elements.

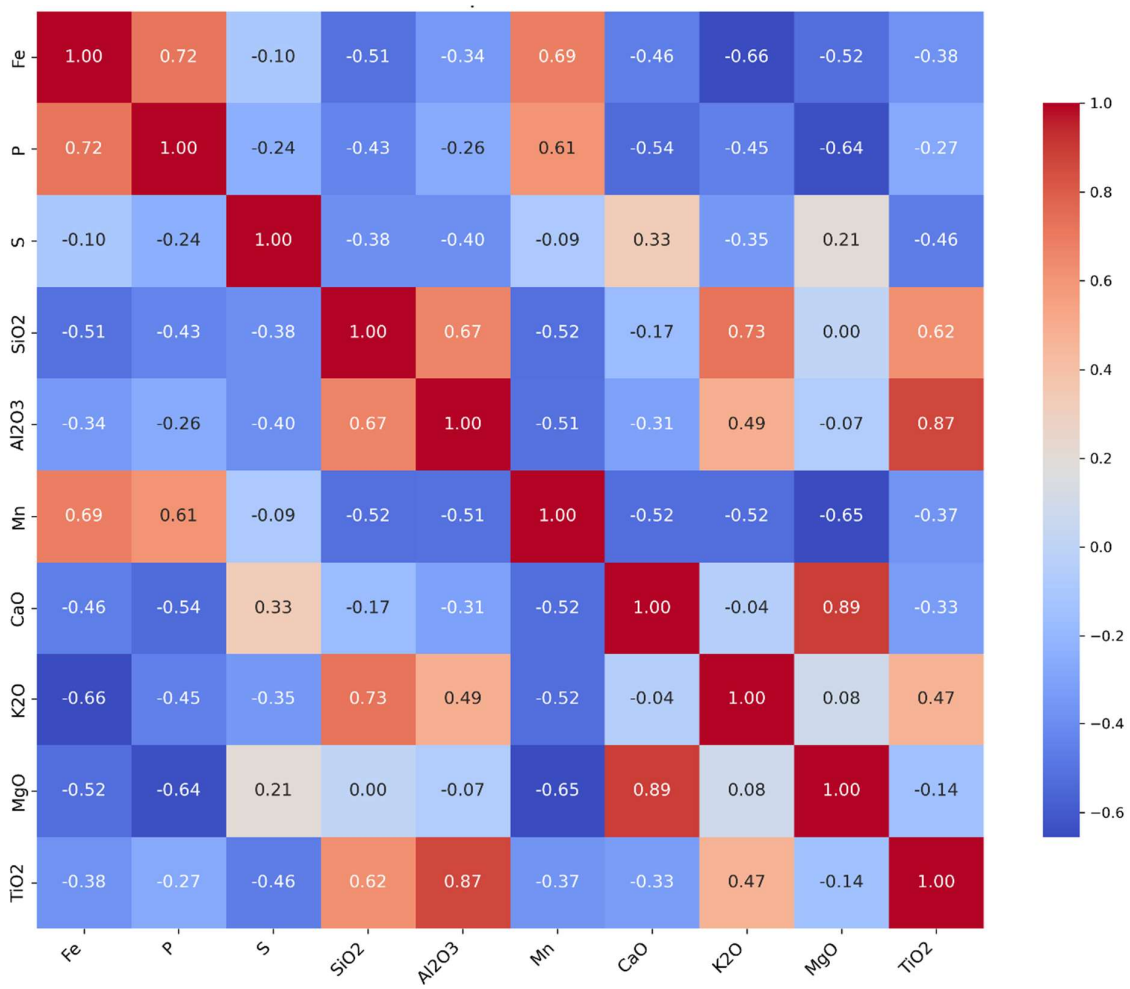


Figure 3.7. Correlation matrix of CLR-transformed geochemical data illustrating the pairwise correlation coefficient among the compositional variables, revealing potential interdependences

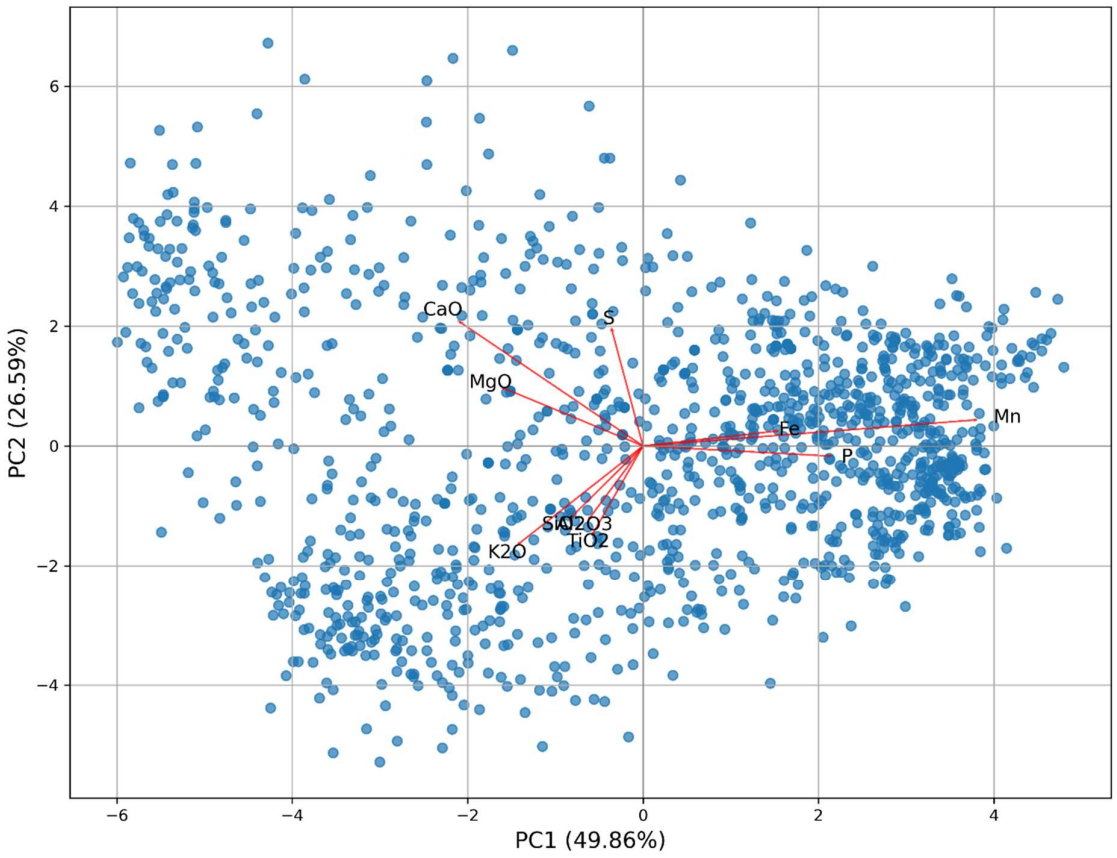


Figure 3.8. PCA of CLR-transformed data

3.5. Model Design and Selection

This section describes how DL models were created and chosen to fit the unique characteristics of the Rocklea Dome datasets. Two DL models were used in this study due to the spatial and temporal components of the dataset: CNNs and LSTM networks.

CNNs are very good at finding hierarchical features and spatial patterns in structured data, like image or grid-like geospatial inputs (Younesi, et al 2024). For the analysis of remotely sensed or geologically gridded datasets, they are an excellent choice due to their capacity to maintain spatial relationships. On the other hand, when working with time-series or temporarily ordered data, LSTMs are essential for modelling long-term temporal patterns and sequential

dependences (Krause et al., 2016). Additionally, both CNNs and LSTMs have shown encouraging outcomes in tasks involving geological interpretation and mineral mapping (Wang et al., 2024; Yang and Zuo, 2024), which supports their selection even more.

Each model is described thoroughly in the following subsections, together with information on its architectural layout, training methods, and specific functions within this study.

3.5.1. Convolutional Neural Networks (CNNs)

CNNs are a type of feedforward neural network that is frequently employed for supervised learning tasks, especially for the interpretation of image and spatial data. CNNs do not create feedback loops; instead, information moves unidirectionally from input layer to output layer. For tasks involving classification or regression, the architecture usually consists of convolutional layers, pooling layers, and activation (non-linear mapping) functions, followed by fully connected layers (Figure 3.9).

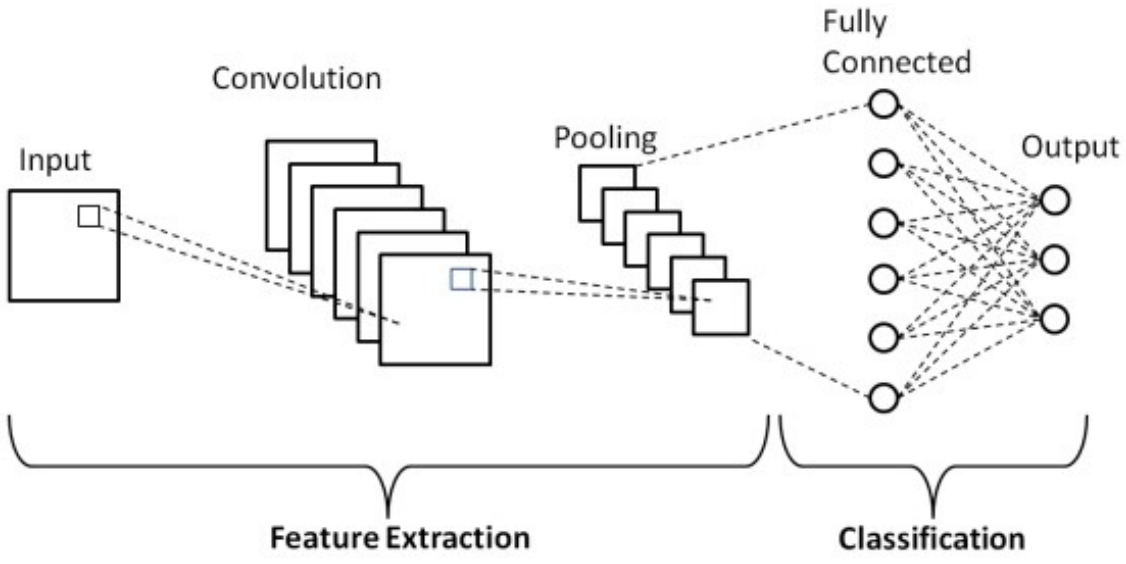


Figure 3.9. CNN model consisting of input, convolution, pooling and fully connected layers.

Hierarchical features are extracted from input data by applying filters or kernels in convolutional layers. A network can identify spatial patterns like edges, textures, or spectral features due to this operation, which is essentially a dot product between a kernel and local receptive field of the input (Li et al., 2023; Agrawal and Govil, 2023). In order to extract the most pertinent features for the given task, the weights of these filters are distributed throughout the input space and improved during training via backpropagation.

A non-linear function is applied to the output of each convolution process, allowing the model to learn non-linear correlation and introducing complexity. The sigmoid, hyperbolic tangent, and rectified linear unit (ReLU) are examples of common activation functions. The ReLU activation function was used in this study, and it is defined as:

$$f(y) = \max(0, y) \qquad \text{.....equation 3.5.1}$$

where y represents the input to ReLU function and $f(y)$ is the output of the ReLU function, defined as:

$$f(y)\begin{cases} y & \text{if } y > 0 \\ 0 & \text{if } y \leq 0 \end{cases}$$

This function outputs the input value only if it is positive, and if not, then it will be zero. The ReLU is frequently used to improve computational efficiency and model convergence by boosting sparse activations and addressing the vanishing gradient issue (Zhang et al., 2022; Li et al., 2023).

The computational process in a CNN layer can be defined as:

$$x_i^l = \sigma\left(\sum_{j \in M_i} x_j^{l-1} . W_{ij}^l + b_i^l\right) \qquad \text{.....equation 3.5.2}$$

where x_i^l represents the result after the convolution operation, σ represents the activation function, x_j^{l-1} represents the region to be convolved, W_{ij}^l represents the convolution kernel, and b_i^l represents the bias factor.

Pooling layers are used to summarize local information and decrease spatial dimensions of feature maps after convolution and activation. Calculating statistical measurements, like maximum or average value within a neighbourhood, is typically part of pooling. This leads to a coarser yet informative representation that lowers computational cost and improves the capacity of network generalization (Chen et al., 2025).

The data are flattened and sent to fully connected layers after going through a number of convolutional and pooling layers. A loss function is used to compare the actual labels with expected outputs during training. The weight of the networks is iteratively updated by gradient descent propagation, which minimizes loss (Alzubaidi et al., 2021; Wang et al., 2024).

Compared to conventional multilayer perceptron, CNNs have a number of advantages, especially when dealing with high-dimensional input data like images or hyperspectral data. A CNN drastically lowers memory needs and computational complexity because of weight sharing and local connectivity. Additionally, CNNs are very good at modelling complex, non-linear relationships, like those between image pixel and mineral classes in geospatial datasets,

because of the architecture, which enables the model to automatically learn optimal feature representations straight from raw data (Agrawal and Govil, 2023).

In this study, the dataset was split into training and testing subsets to develop and evaluate the deep learning models. Specifically, 80% of the data (1,020 corresponding samples combining HS data, geochemical data, and mineral abundance measurements) was used for training, while the remaining 20% (256 corresponding samples) was reserved for testing. HS and geochemical measurements served as predictor variables, and mineral abundances were treated as the target variables.

Five-fold cross-validation was used to enhance model generalizability and thoroughly evaluate performance. Five identically sized folds were created from the dataset, and the model was trained on four of these folds and validated on the fifth. This process was performed five times, with each fold serving as the validation set once. According to Acosta et al. (2020), this method improves the dependability of model performance indicators and lessens the impact of overfitting.

Figure 3.10 displays the CNN model training results, and Table 3.1 lists the training and validation losses for each fold of the five-fold cross-validation along with the related MAE values. The goals of the study to measure predictive performance and uncertainty were directly addressed by using these assessment measures to gauge how well the models predicted mineral abundances from HS and geochemical data. The measurements assist guarantee that the DL models can accurately map minerals in unknown drill-core data by tracking both training and validation mistakes. This thorough setup improves repeatability and enables future researchers

to duplicate the modelling process while assessing predicted accuracy in accordance with the objectives of the study.

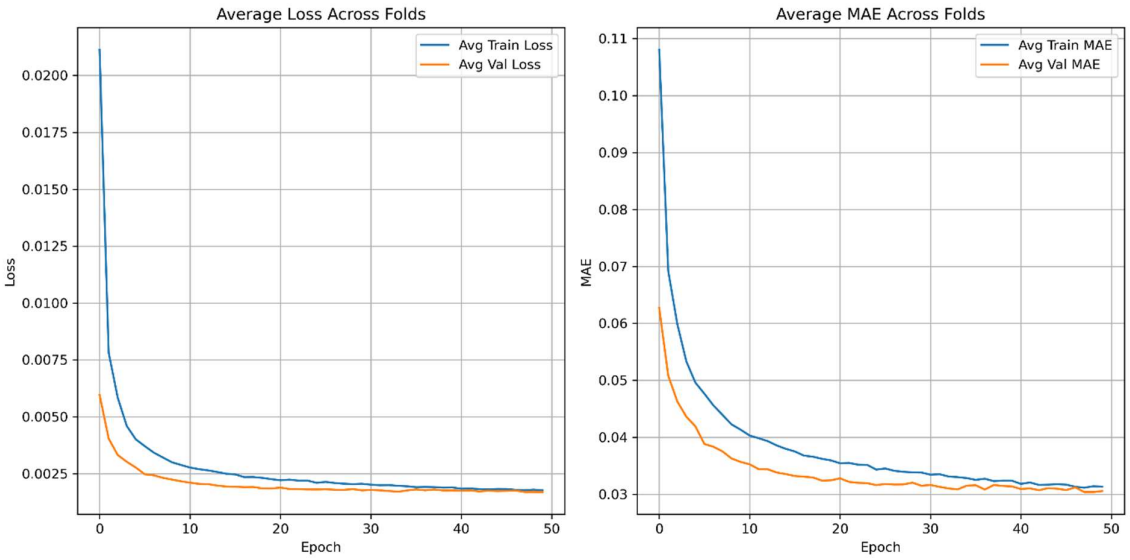


Figure 3.10. Average training loss across folds and average MAE across folds for CNN model over 50 epochs.

Table 3.1 Training loss with training MAE and validation loss with validation MAE during CNN model training on five-fold cross validation (epoch 50/50)

Five-Fold Cross Validation	Training Loss	Training MAE	Validation Loss	Validation MAE
Fold 1	0.0017	0.0208	0.0017	0.0315
Fold 2	0.0019	0.0328	0.0015	0,0296
Fold 3	0.0017	0.0308	0.0016	0.0282
Fold 4	0.0018	0.0320	0,0019	0,0328
Fold 5	0.0019	0.0327	0.0016	0.0297

During model training, dropout layers were incorporated to enhance robustness and prevent overfitting. By randomly disabling a specified fraction of neurons in each training iteration, a dropout layer forces the network to learn redundant and distributed feature representations, ensuring that the model does not rely too heavily on any single neuron. This mechanism improves generalization to unseen data, making CNN and LSTM models more reliable for predicting mineral abundances in new drill-core samples.

3.5.2. Long-Short Term Memory (LSTM)

A type of neural network called recurrent neural network (RNN) is suitable to processes sequential data. An input layer, an output layer, and a recurrent hidden layer make up a standard RRN model. In contrast to feedforward neural networks, RNNs are able to retain an internal memory of past inputs because of a feedback loop in the hidden layer. The model can learn temporal dependencies by integrating data from previous states with the current input by updating this internal state at each time step (Graves, 2013; Wang et al., 2024).

The capacity of RNNs to represent contextual and sequential relationships between variables is one of the advantages of RNNs. However, when working with prolonged sequences, conventional RNNs frequently have problems like vanishing and exploding gradients. These restrictions make it more difficult for RNNs to accurately identify long-term dependency. LSTM network is a specialized RNN variation for overcoming these issues. The limitations of conventional RNNs are addressed by LSTM networks, which are specifically made to preserve information across long sequences (Wang and Zuo, 2022). Memory cells and gating mechanisms, which control the information flow, are the main innovations in LSTMs. Figure 3.11 below shows the structure of a LSTM model.

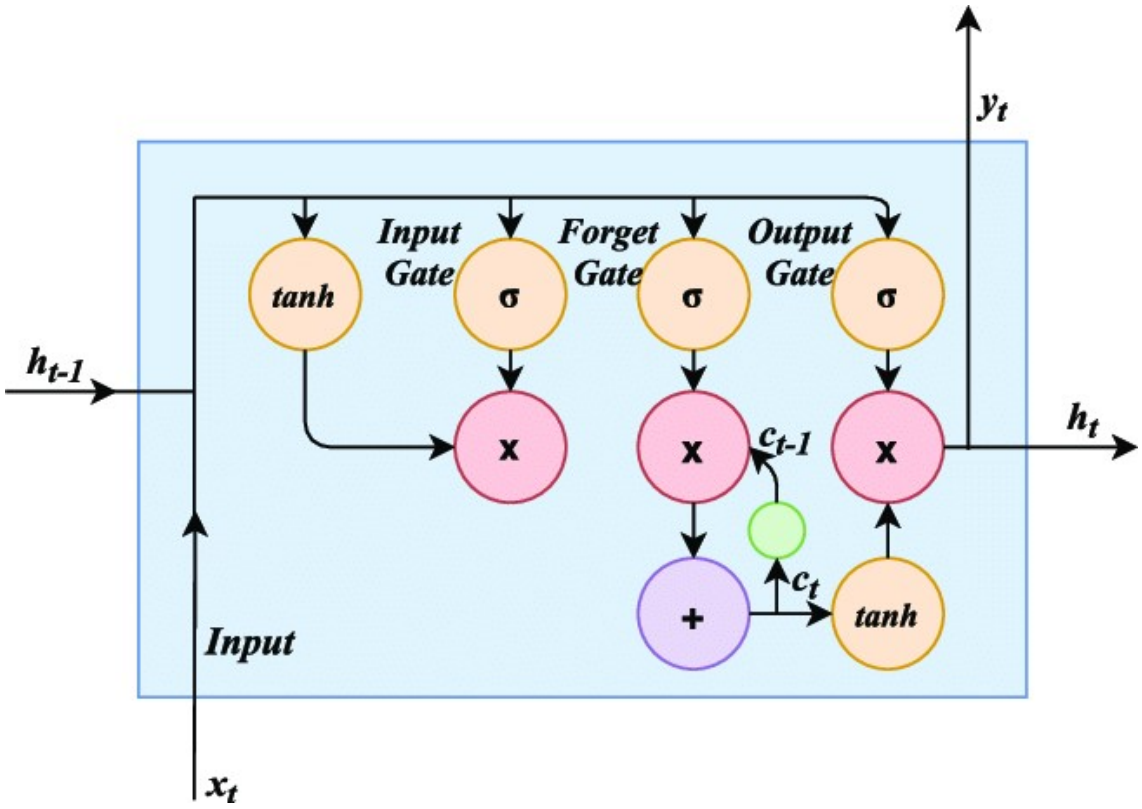


Figure 3.11. LSTM structure showing the input, forget and output gates.

Three different kinds of gates are included in each LSTM unit:

- The input gate regulates the amount of fresh data that enters the cell state. It is made up of two parts, a tanh layer that generates candidate values to be added to the state and sigmoid layer that decides which values to update.
- The forget gate scales the impact of prior memories by using a sigmoid function to determine which parts of the past should be kept or deleted, yielding values between 0 and 1.
- The output gate is responsible for determining which part of the cell state should be output to the next step.

The gates provide LSTMs the ability to selectively recall and forget data according to the significance of both current and previous inputs. LSTMs are especially well-suited for tasks

like speech recognition, natural language processing, and time series forecasting that call for modelling long-term dependencies because of LSTMs architecture (Wang and Zuo, 2022; Chen et al., 2025).

LSTM networks are becoming more and more acknowledged for their ability to represent intricate temporal or sequential patterns in HS and geochemical data, despite not being historically used in spectral data analysis. The limitations of CNNs, which mostly concentrate on spatial hierarchies, are offset by their ability in capturing sequence-based data. Additionally, LSTMs are especially useful for interpreting the genesis-relevant and contextual properties of geochemical elements (Chen et al., 2025; Wang et al., 2024).

In this study, a LSTM model was trained using the same data preparation approach as that used for CNN model above. The model was trained with 80% of the samples and the 20% was reserved for testing, with HS and geochemical data used as predictor variables and mineral abundances as target variable during training. Likewise, five-fold cross validation was implemented, consistent with the methodology applied in CNN training (Acosta, 2020). Additionally, dropout layers were integrated into the LSTM model to prevent overfitting and improve the ability of the model to generalize to unseen data. Figure 3.12 shows average training loss across folds and average MAE across folds for LSTM model over 50 epochs and Table 3.2 shows training loss with training MAE and validation loss with validation MAE during LSTM model training on five-fold cross validation. The results in Table 3.2 shows that the model trained well as the MAE was very low during training.

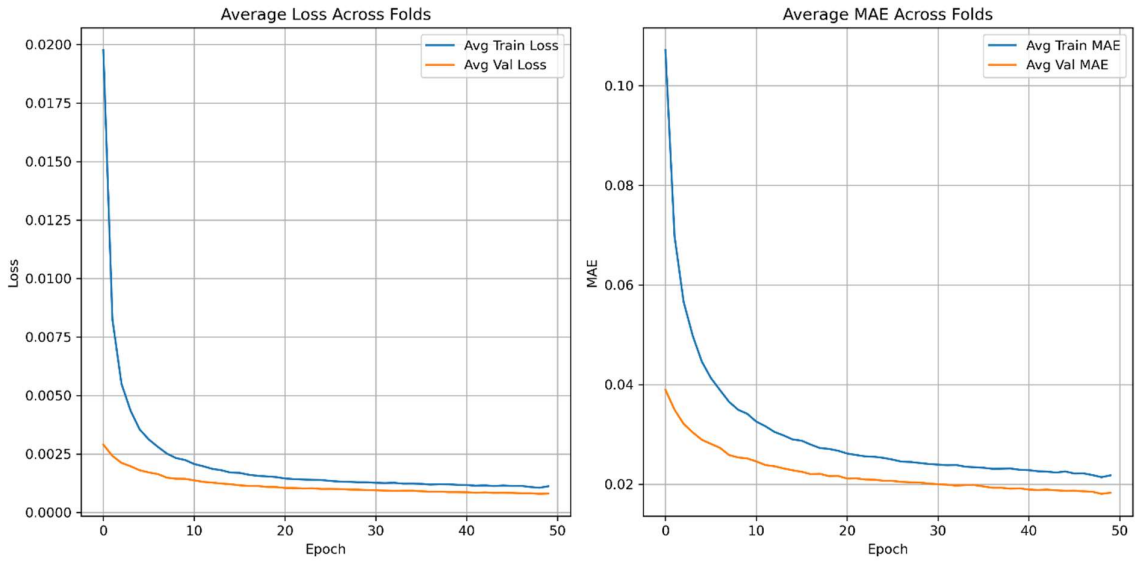


Figure 3.12. Average training loss across folds and average MAE across folds for LSTM model over 50 epochs.

Table 3.2. Training loss with training MAE and validation loss with validation MAE during LSTM model training on five-fold cross validation (epoch 50/50)

Five-Fold Cross Validation	Training Loss	Training MAE	Validation Loss	Validation MAE
Fold 1	0.0018	0.0306	0.0012	0.0258
Fold 2	0.0016	0.0302	0.0011	0.0256
Fold 3	0.0015	0.0294	0.0010	0.0231
Fold 4	0.0017	0.0308	0.0012	0.0270
Fold 5	0.0017	0.0313	0.0011	0.0249

3.6. Concluding Remarks

This chapter describes the methodological framework used in this study to analyse the Rocklea Dome dataset. Pre-processing of HS and geochemical data, mineral abundances per drill-core, as well as creation, training and validation of DL models for mineral mapping, are the main areas of emphasis.

PCA was used as the only pre-processing step for HS data in order to minimize dimensionality and emphasize the most informative spectral features. In contrast, a more intricate methodology

was required for the pre-processing of geochemical data. This involved CLR transformation to handle compositional data limitations, and PCA for dimensionality reduction.

After pre-processing, HS and geochemical data were integrated for mineral mapping using DL models (CNNs and LSTMs) as predictor variables and mineral abundance was used as the target variable. To guarantee robustness, five-fold cross validation was used for both training and validation of these models. The training loss with training MAE and validation loss with validation MAE during CNN model training on five-fold cross validation the models are shown in this chapter.

The techniques presented in this chapter provide a strong basis for the next chapter, which presents a CNN-LSTM hybrid method. This hybrid model underwent training and validation as the single models. In addition, all the three models (CNN, LSTM, and CNN-LSTM) went through testing for mineral mapping and performance evaluation, which included evaluating the MAE of all the models.

CHAPTER 4: HYBRID CNN-LSTM IMPLEMENTATION AND MODEL TESTING

4.1. Hybrid CNN-LSTM Implementation

On the one hand, CNNs have proven to be highly effective in processing time-series, audio, and image data because of their capacity to extract important features and minimize noise from complex and high dimensional dataset (e.g., Kiranyaz et al., 2021; Ragab et al., 2021; Raj and Kos, 2025). Convolutional layers are used to automatically learn spatial feature hierarchies by backpropagation. This makes CNNs ideal for applications for the recognition of spatial patterns, such as geological mapping and remote sensing (Yang et al., 2020).

On the other hand, temporal sequences can be effectively modelled by LSTM networks, a particular type of RNNs. Both short-term and long-term dependencies in sequential data can be captured by LSTM networks through the use of memory cells and gating mechanisms. Because of this, they are especially useful for time-dependent applications where predictions rely heavily on past observations. According to Yang et al. (2020), LSTM networks are ideal for sequential modelling application because they integrate historical data with present input to improve prediction accuracy.

Several researches have shown how useful hybrid DL architectures are for mineral prospecting and risk assessment. For example, Zhao et al. (2020) introduced a hierarchical spatial-spectral (HSS) feature extraction method combined with LSTM networks to investigate the interaction between spatial and spectral information. Their suggested HSS-LSTM model greatly outperformed conventional ML techniques in mineral identification tasks, achieving an impressive overall classification accuracy of 94.7%. Likewise, Du et al. (2021) used a hybrid

model to map prospective gold deposit zones in Karamay, northwest China, by combining a genetic algorithm (GA) and SVM. The outcomes demonstrated the potential of integrating optimization algorithms with ML classifiers and validated the resilience of GA-SVM model for mineral prospectivity mapping. Another noteworthy contribution was made by Yang et al. (2020), who created a CNN-LSTM hybrid model to assess the risks associated with tailings ponds. When compared to standalone CNN or LSTM models, the hybrid model performed better in terms of MAE, room mean square error, and R^2. Together, these studies demonstrated the benefits of combining hybrid ML models, particularly when working with spatiotemporal dataset that exhibits both temporal dependency and spatial variability.

The current study built on pre-existing hybrid frameworks by putting forth a novel hybrid CNN-LSTM designed especially for mineral mapping using geochemical and hyperspectral (HS) data. High-level spatial features were extracted from input imagery using CNN components in this architecture, while sequential or temporal patterns present in the dataset were captured by LSTM layers (Zhao et al., 2021). By utilizing the advantages of both CNNs and LSTM networks, this hybrid model approach makes it possible to express spatiotemporal features more thoroughly, which is essential for precise mineral mapping.

The dataset was divided into training and testing subsets in 80:20 ratio in order to assess the effectiveness of the proposed CNN-LSTM model. The training set was used to optimize the parameters of the model, while testing set was used to assess the capacity of the model for generalization. A five-fold cross validation method was used to guarantee robustness and reduce the possibilities of overfitting. By dividing the training data into five equal subsets, the model was relatively trained on four training subsets and validated on the fifth. The stability of

the model was evaluated using the average performance across the folds. For both the training and validation datasets, important performance indicators including MAE was measured during the training process. Figure 4.1 illustrates the average training loss across folds and average MAE across folds for hybrid CNN-LSTM model over 50 epochs. Table 4.1 shows training loss with training MAE and validation loss with validation MAE during hybrid CNN-LSTM model training on five-fold cross validation, and the training results show good performance of the model as the MAE kept on decreasing. The efficacy of hybrid CNN-LSTM, as thoroughly assessed in this empirical study, lays groundwork for further studies in data-driven geoscientific modelling.

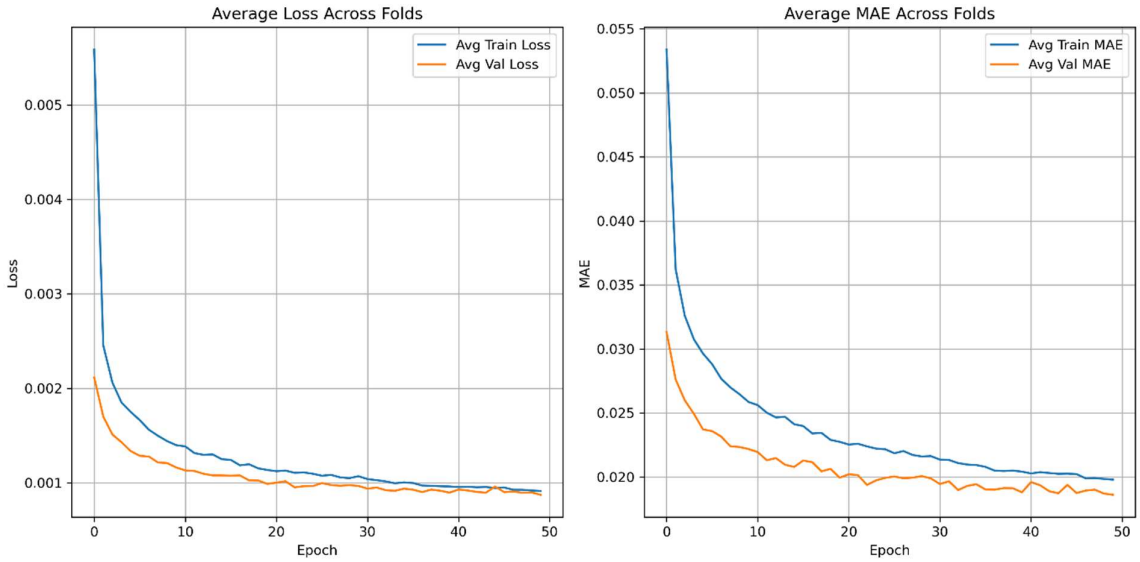


Figure 4.1. Curves of average training and validation loss across folds, and average training and validation MAE across folds for hybrid CNN-LSTM model over 50 epochs.

Table 4.1. Training loss with training MAE and validation loss with validation MAE during CNN model training on five-fold cross validation.

Five-Fold Cross Validation	Training Loss	Training MAE	Validation Loss	Validation MAE
Fold 1	0.0013	0.0269	0.0011	0.0247
Fold 2	0.0011	0.0253	0.0012	0.0270
Fold 3	0.0010	0.0241	0.0010	0.0226
Fold 4	0.0011	0.0252	0.0009	0.0233
Fold 5	0.0011	0.0254	0.0010	0.0222

4.2. CNN, LSTM, and Hybrid CNN-LSTM Model Testing

According to Lwakatare et al. (2020), ML models work by identifying patterns from input data and by using these representations to carry out tasks like regression or classification. Assigning input data to one of several predefined groups is the goal of classification tasks. The objective of regression is to predict a continuous output variable (e.g., mineral abundance in this research). It is crucial to make sure that ML models produce accurate and reliable predictions, particularly in high-stake fields like healthcare, remote sensing, or geoscientific exploration.

To ascertain the efficacy and reliability of newly created and trained ML or DL models, thorough testing and comparative analyses are crucial (Rainio et al., 2024). This guarantees that models not only function as anticipated in test experimental settings but also produce reliable outcomes in practical applications. Thorough testing entails a number of evaluations intended to confirm the errors, applicability, and robustness of a model in various conditions. In addition to verifying proper operation, it helps in locating problems such as algorithms bias, overfitting, data imbalance, and designed flaws. In this way, testing helps create more reliable and understandable models that satisfy certain application objectives and user expectations.

The MAE is a statistical metric that calculates the average absolute difference between the actual and the predicted values. Mathematically, it is calculated as the mean of the absolute

value of errors, with each error representing the deviation of a predicted value from its corresponding actual value. MAE is commonly employed as a loss function in predictive modelling, with the goal of lowering its value to improve model performance (Babu and Venkatram, 2020). The MAE is also known as mean absolute deviation. The range of MAE is (0,+ ∞), with smaller numbers suggesting higher prediction accuracy. One of the MAE benefits is that it uses the same unit as the original data, making it simple to interpret. Additionally, the MAE is a symmetrical loss function, which means that it penalizes both overestimation and underestimation equally (Jierula et al., 2021). In order to calculate the MAE of each model, the following equation was applied:

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \qquad \text{.....equation 4.2.1}$$

where y_i is the i^{th} sample point output and $\hat{y}_i$ is the estimated value obtained from the model.

The R^2 is a statistical metric that measures the fit of a regression model. It measures the amount of variance in the target variable that can be explained by the predictor variables of a model (Anand et al., 2024). The values of R^2 varies from 0 to 1. A value of 0 implies that the model does not explain any of the variability in the response variable, while a value of 1 suggests that the model accounts for all such variability. In multiple regression, higher R^2 values indicate a better fit between a model and the data (Allison, 2013). Values of R^2 are calculated as follows:

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

.....equation 4.2.2

$$R^2 = 1\frac{\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}{\sum_{i}^{n}(y_i - \bar{y}_i)^2}$$

where $SS_{res} = \sum_{i=1}^{n}(\hat{y}_i - y_i)^2$ is the residual sum of squares, $SS_{tot} = \sum_{i}^{n}(y_i - \bar{y}_i)^2$ is the total sum of squares, n is the number of observations, y_i is the actual or true value of the dependent

variable for observation i, $\hat{y}_i$ is the predicted value for observation i, and $\bar{y}$ is the mean of all actual value.

Following the training on 80% of the dataset and using five-fold cross validation, each DL model (CNN, LSTM, and hybrid CNN-LSTM) was assessed on the 20% testing set. This 20% refers to HS and geochemical data, which are the predictor variables. So, these models were tested on the predictor variables in order to predict the mineral abundances. During model testing, the overall MAE of the three DL models were recorded. The overall MAE of the CNN model was 0.0312; the overall MAE of the LSTM model was 0.0278, and that the hybrid CNN-LSTM was 0.0258. These results indicate good performance of the models, as the MAEs of all the models are very low. Table 4.2 shows the MAEs of the three models in predicting mineral abundance, *Fe_ox_ai* representing hematite, goethite, jarosite, and limonite; *kaolin* representing kaolin group minerals, namely kaolinite halloysite, dickite and nacrite; *wmAlsmai* representing abundance of white micas (e.g. illite, muscovite, paragonite, brammalite, phengite, lepidolite, margarite) and Al-smectites (montmorillonite, beidellite); and *carbai3pfit* representing carbonates, and the last column represents the overall abundance per model.

Table 4.2. Mineral abundance MAEs and overall MAE during model testing of CNN, LSTM and a hybrid CNN-LSTM models.

Model	Fe_ox_ai	Kaolin	Wmalsmai	Carbai3pfit	Overall, MAE
CNN	0.0321	0.0399	0.0186	0.0101	0.0312
LSTM	0.0259	0.0372	0.0199	0.0124	0.0278
CNN-LSTM	0.0244	0.0348	0.0178	0.0131	0.0258

The CNN model explained 76% of the variance in the target variable during testing, with an R^2 value of 0.76 (Table 4.3). The LSTM model outperformed the CNN model with R^2 value of

0.83, accounting for approximately 83% of the variation. The hybrid CNN-LSTM model outperformed both stand-alone models (CNN and LSTM), with R^2 value of 0.84. These results compliment the MAEs of all the models, because the errors are very low. This is shown by plotting the predicted mineral abundances against the true mineral abundances (Figure 4.2).

The performance variations that have been noted are a reflection of each model design unique advantages. CNN can identify localized mineralogical features in the HS data and explain a significant amount of variance thanks to its capacity to extract spatial-spectral patterns. Sequential dependencies along the drill core, capturing the continuity of mineralogical variations, are crucial for accurate forecasts, according to the increased performance of LSTM model. Hybrid CNN-LSTM model higher prediction performance can be attributed to its ability to extract spatial-spectral data while concurrently modelling sequential patterns. This demonstrates how drill-core mineral mapping can better capture intricate, non-linear interactions by incorporating complimentary model designs.

Table 4.3: Performances of CNN, LSTM, and CNN-LSTM Models across mineral abundances based on R^2 and overall R^2

Model	Fe_ox_ai	Kaolin Abundance	Wmalsmai	Carbai3pfit	Overall, R^2
CNN	0.6737	0.5737	0.1840	- 0.6967	0.7590
LSTM	0.7975	0.6927	0.1988	- 2.0502	0.8341
CNN-LSTM	0.8250	0.6782	0.2591	- 1.5609	0.8437

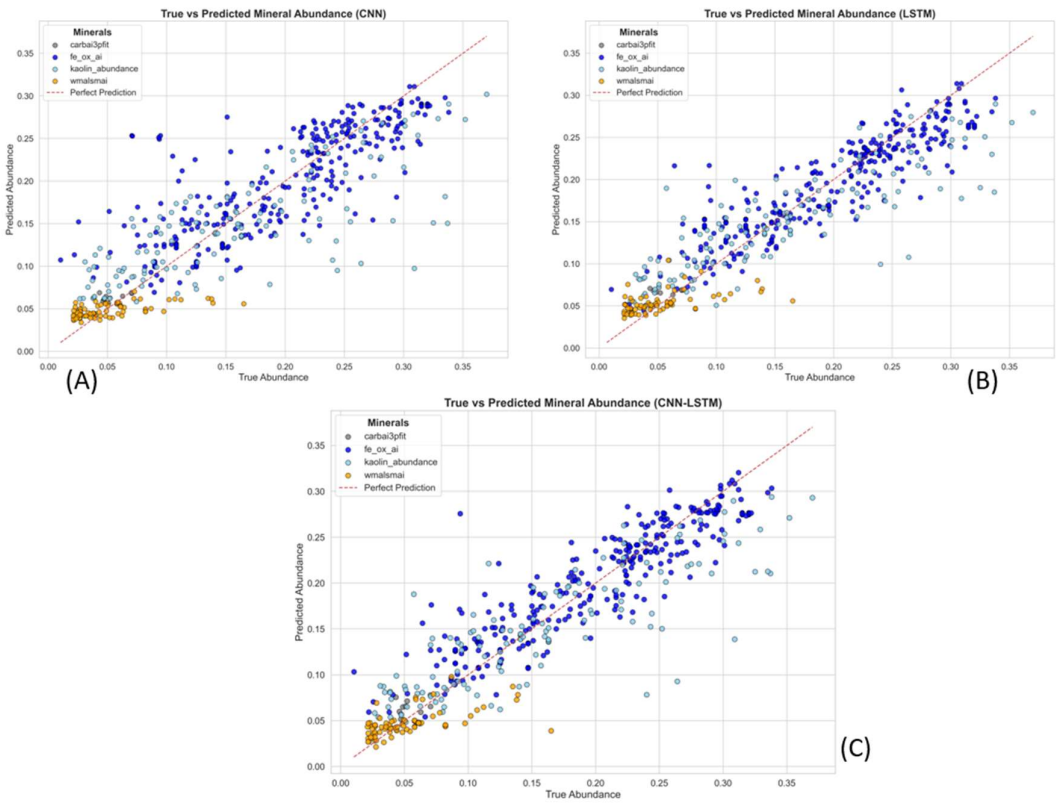


Figure 4.2. Scatter-plots of the true vs predicted mineral abundances of (A) CNN model, (B) LSTM model, and (C) hybrid CNN-LSTM model.

The true mineral abundances and those predicted by the LSTM and hybrid CNN-LSTM models showed overall a good correlation with few outliers related to kaolin minerals. Based on the results of the hybrid CNN-LSTM model, combining spatial feature extraction from CNNs with temporal dependency modelling from LSTMs resulted in a modest improvement. These findings show that, while LSTM model already had good predictive capabilities, the hybrid architecture can achieve increased performance, possibly due to its ability to exploit the complementary qualities of both network types.

4.3. Uncertainty Quantification

Uncertainty quantification has evolved as an important aspect of predictive modelling, especially in domains where model outputs influence high-impact decision (McClarren et al.,

2018). Uncertainty quantification serves multiple purposes by explicitly estimating the uncertainty associated with predictions:

- It enables robust assessment of model performance, with high uncertainty often signalling the need for model refinement.
- It informs the estimation of the necessary training dataset size, guiding whether additional data collection is wanted to improve predictive accuracy.
- It supports more reliable prediction-making while mitigating the risk of biases in downstream analysis.

According to Quigley et al. (2019), uncertainty is a fundamental component of modelling and prediction in the natural sciences and can be generally divided into aleatory uncertainty and epistemic uncertainty. Aleatory uncertainty results from natural variation of inputs and parameters (e.g., small imprecisions of geochemical assays or GPS coordinates) and it is also known as stochastic or statistical uncertainty. This type of uncertainty cannot be reduced, even if more data are gathered, the variability will still exist. On the other hand, a lack of knowledge or insufficient information about the system under study is the cause of epistemic uncertainty, also referred to as systemic uncertainty (Yousefi et al., 2024 and Davis et al., 2020).

Theoretically, epistemic uncertainty is reducible, unlike aleatory uncertainty, and it can be reduced by adding information to the modelling process, improving the model architecture, or collecting more data (Hullermeier and Waegeman, 2021).

Recent advances in uncertainty quantification within ML have resulted in various notable approaches, such as Bayesian neural networks (BNNs), Monte Carlo (MC) Dropout, and deep ensembles. The latter two make several predictions, requesting statistical analysis of their predictive distributions. While BNNs provide a sound Bayesian foundation for uncertainty estimation, high parameters requirements frequently provide them computationally prohibitive (Puzyrev and Duuring, 2025).

Among these techniques, the MC Dropout has become particularly popular. Dropout was first proposed by Hinton et al. (2012) as a regularization strategy for reducing overfitting in neural networks. It works by randomly disabling a subset of neurons during each training iteration. The proportion of neurons disabled, known as the dropout rate, normally ranges between 0 (no dropout) and 0.3, with higher values up to 0.5 used in certain circumstances. The appropriate dropout rate is determined by several architectural and data-related criteria, such as network depth, the number of neurons per layer, and the degree of overfitting observed (Puzyrev and Duuring, 2025). When used in probabilistic inference context, the MC Dropout can approximate Bayesian uncertainty estimates of fully Bayesian methods.

In this study, uncertainty estimation was carried out as part of the model evaluation process for DL models, namely CNN, LSTM networks, and hybrid CNN-LSTM models. The epistemic uncertainty was estimated using MC Dropout. The following equation was used to calculate the epistemic uncertainty of the three models:

$$\text{Uncertainty} = \frac{1}{T}\sum_{t=1}^{T}\hat{y}_t^2 - \left(\frac{1}{T}\sum_{t=1}^{T}\hat{y}_t\right)^2 \qquad \text{.....equation 4.3.1}$$

where T is the number of Monte Carlo samples (forward passes with dropout active), $\hat{y}_t$ prediction from the model on the t-th forward pass with dropout mask applied.

The outcomes displayed in Table 4.5 demonstrate that the epistemic uncertainty of all three models is very low, as calculated using Python software. This low uncertainty reflects the reliability of the models' predictions, since each model was trained multiple times (50 epochs per fold) with carefully tuned hyperparameters and regularization strategies, which reduced variability in model outputs. Notably, hybrid CNN-LSTM model exhibited the lowest uncertainty among the three, indicating that its superior predictive performance is not only a result of combining spatial feature extraction (CNN) with temporal sequence modelling (LSTM), but also of achieving high confidence in predictions across different mineral variables. These findings suggest that models with integrated architectures can simultaneously improve accuracy and reduce uncertainty, providing more dependable drill-core mineral mapping outcomes. By explicitly linking low uncertainty to model robustness, this interpretation highlights how uncertainty estimation directly informs the trustworthiness and practical applicability of the predictive results.

Table 4.5. Evaluation of model confidence and uncertainty metrics (CNN, LSTM, and hybrid CNN-LSTM)

Model	Fe_ox_ai	Kaolin	Wmalsmai	Carbaipfit	Overall Uncertainty
CNN	0.000484	0.000542	0.000138	0.000296	0.000437
LSTM	0.000630	0.000846	0.000116	0.000342	0.000594
CNN-LSTM	0.000278	0.000253	0.000061	0.000080	0.000231

4.4. Mineral Abundance and Mineral Association Maps

Following the prediction stage, the outputs of three models (CNN, LSTM, and hybrid CNN-LSTM) were used to create mineral abundances maps for the study area. These maps were

generated in Python using the predicted mineral abundance values from the test dataset. Each model predicted its own maps because the predicted mineral abundances varied between models due to the differences in their underlying structures and feature extraction capabilities.

Figures 4.3, 4.4, and 4.5 show the mineral abundance maps for each mineral group, as predicted by the CNN, LSTM, and hybrid CNN-LSTM models, respectively. A visual comparison of these maps shows that Fe-oxide minerals are the most common in the area, followed by kaolin or clay minerals. These results are congruent with the geological characteristics of the Rocklea Dome, a channel iron ore deposit known for its Fe-oxide mineralisation.

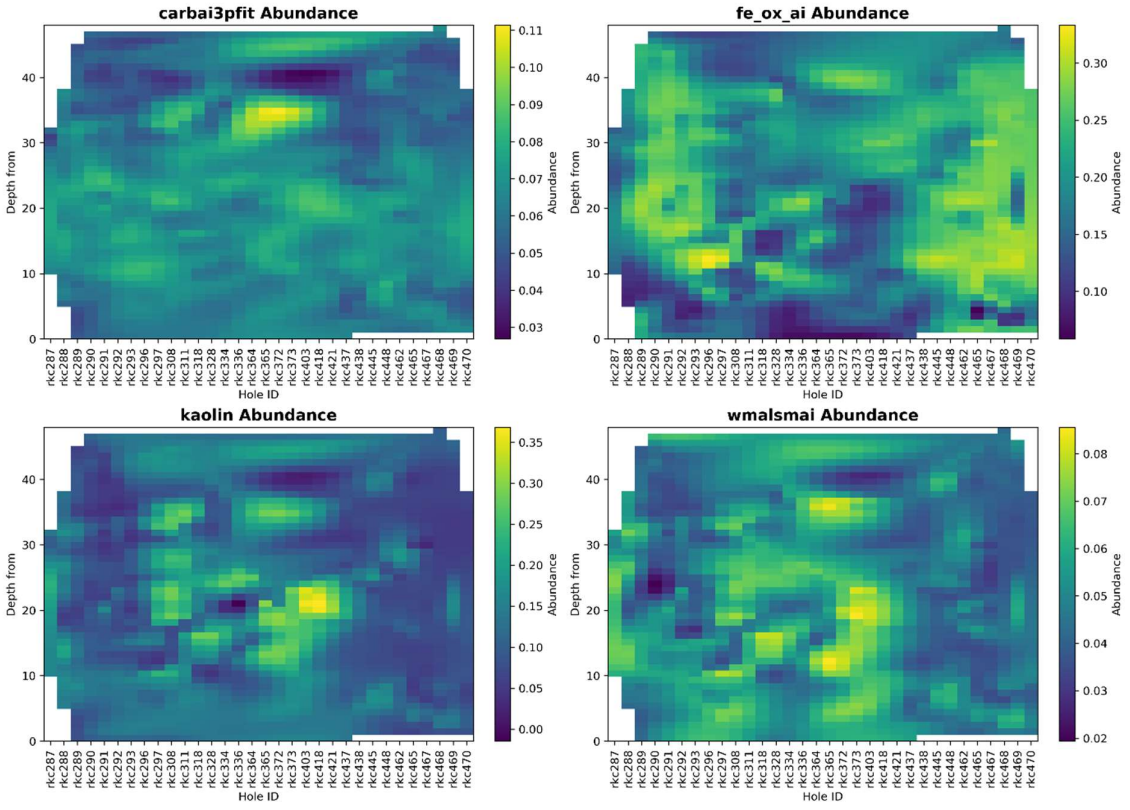


Figure 4.3. Mineral abundance map created from the mineral abundance predictions which were predicted by the CNN model.

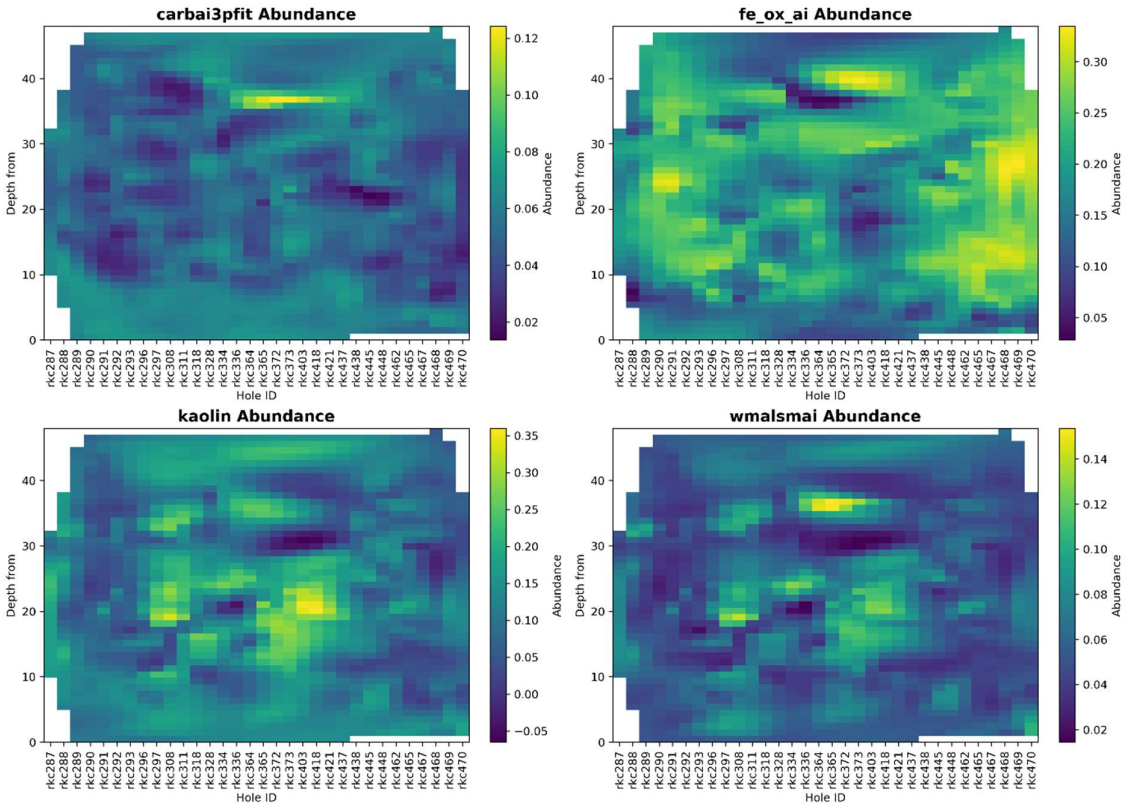


Figure 4.4. Mineral abundance map created from the mineral abundance predictions which were predicted by the LSTM model.

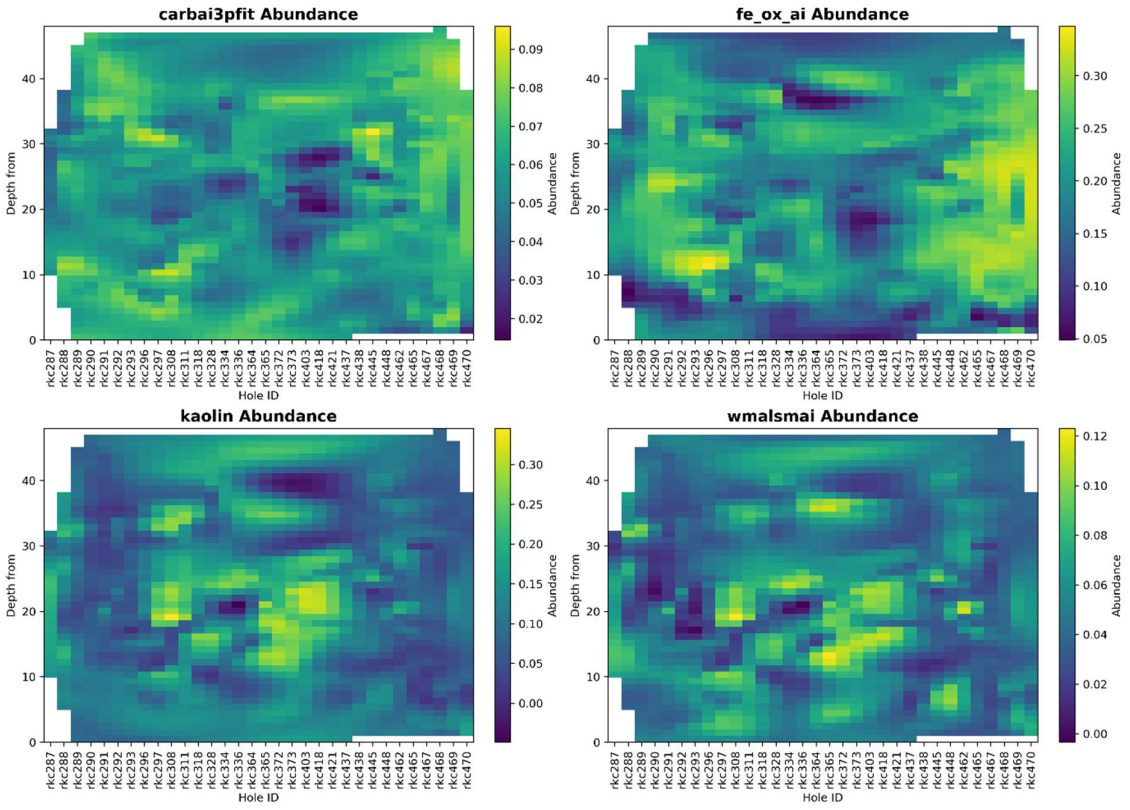


Figure 4.6. Mineral abundance map created from the mineral abundance predictions which were predicted by the hybrid CNN-LSTM model.

In contrast, white micas and carbonates minerals appear in far lower proportions in all the three model predictions. These patterns maybe linked to the geological and geochemical conditions of the deposit, which favour the accumulation of iron-rich minerals over alteration minerals such as micas and carbonates.

Mineral association maps were then created using the mineral abundances predicted by three models (CNN, LSTM, and hybrid CNN-LSTM) to analyse spatial mineralogical relationships. The goal of these maps was to identify zones with similar mineralogical compositions, which would provide insight into potential geochemical and geological patterns in the studied area.

The predicted mineral abundances from each model were initially clustered using K-nearest neighbours (KNN) clustering technique. For consistency, four clusters were chosen for each model, characterised by the average predicted mineral abundances (Tables 4.6, 4.7, and 4.8).

Table 4.6. Average mineral abundance per cluster from the CNN model

Cluster	Fe_Ox_ai	Kaolin	Wmalsmai	Carbai3pfit
1	0.2633	0.0814	0.0432	0.0690
2	0.1398	0.1398	0.0566	0.0637
3	0.1619	0.2041	0.0662	0.0746
4	0.2361	0.0723	0.0415	0.0531

Table 4.7. Average mineral abundance per cluster from the LSTM model

Cluster	Fe_Ox_ai	Kaolin	Wmalsmai	Carbai3pfit
1	0.2170	0.0914	0.0499	0.0581
2	0.1413	0.2181	0.0922	0.0548
3	0.2564	0.0654	0.0450	0.0408
4	0.1331	0.1473	0.0596	0.0639

Table 4.8. Average mineral abundance per cluster from the hybrid CNN-LSTM model

Cluster	Fe_Ox_ai	Kaolin	Wmalsmai	Carbai3pfit
1	0.2609	0.0615	0.0376	0.0728
2	0.1600	0.1409	0.0588	0.0631
3	0.2410	0.0672	0.0332	0.0548
4	0.1463	0.2298	0.0847	0.0408

The four clusters were chosen in order to strike a balance between the resolution of the mineralogical variability and interpretability for geological applications. Figure 4.6 depicts the mineral association maps for all the three models, with each map directly correlating to the mineral abundance maps displayed above (Figure 4.3 - 4.5).

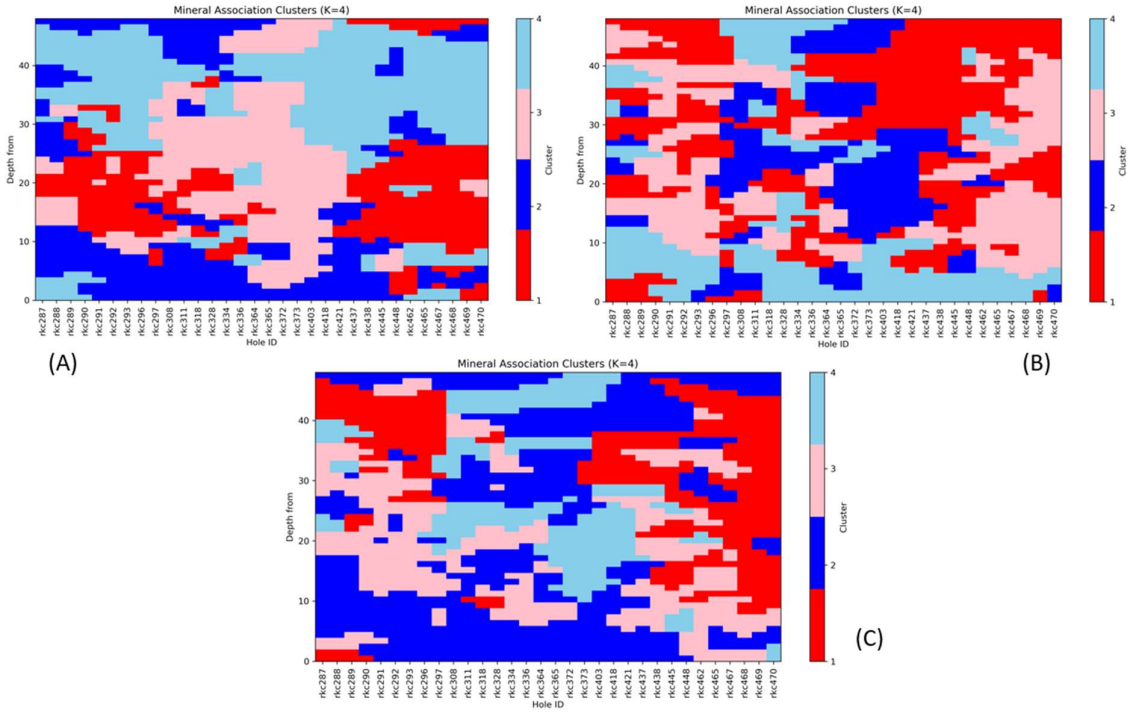


Figure 4.7. Mineral association maps based on mineral abundances predicted by (A) CNN model, (B) LSTM model, and (C) hybrid CNN-LSTM model.

Clustering of CNN model predictions revealed that:

- Cluster 1 had the highest Fe-oxide abundance (0.263).
- Cluster 2 showed moderate kaolin (0.140) and Fe-oxide (0.140) abundances.
- Cluster 3 had high kaolin abundance (0.204) and moderate Fe-oxide (0.162) abundance.
- Cluster 4 had high Fe-oxide (0.236) but low kaolin (0.072) abundance.

For LSTM mode predictions:

- Cluster 1 had high Fe-oxide (0.217) and low kaolin (0.091) abundances.
- Cluster 2 had the highest kaolin abundance (0.218).
- Cluster 3 had the highest Fe-oxide abundance ().256).
- Cluster 4 exhibited moderate Fe-oxide (0.133) and kaolin (0.147).

For hybrid CNN-LSTM model predictions:

- Cluster 1 had the highest Fe-oxide abundances (0.261).
- Cluster 2 had moderate Fe-oxide and kaolin (0.140) abundances.
- Cluster 3 had high Fe-oxide (0.241) and low kaolin (0.067) abundances.
- Cluster 4 had highest kaolin (0.230) and moderate Fe-oxide (0.146) abundances.

It is important to note that the cluster labels do not have the same meanings across the three models. This difference resulted from the differences in the mineral abundances predicted by CNN, LSTM, and hybrid CNN-LSTM models, as each clustering was done individually for each set of predictions. As a result, the number of clusters remained constant, but their mineralogical compositions are fundamentally model specific.

4.5. Results Discussion

This section interprets and discusses the results of the three DL models used in this study (CNN, LSTM networks, and hybrid CNN-LSTM). This discussion only relates to the outcomes of model testing on the mineral regression task, which was carried out in this study.

The hybrid CNN-LSTM model outperformed the other two models in terms of MAE of 0.0258, LSTM model 0.0278 and CNN model with 0.0312. These outcomes showed that compared to the stand separate standalone models, the hybrid CNN-LSTM model, which combines spatial and sequential feature extraction capabilities, provides a performance advantage. Additionally,

based on R^2, hybrid CNN-LSTM model outperformed the stand-alone models with 0.84, LSTM model with 0.83 and lastly CNN with 0.76.

The ability of the hybrid CNN-LSTM model to integrate the temporal sequence learning capability of LSTM networks with the local spatial feature extraction ability of CNNs is responsible for its superior performance. This is especially helpful for mineral classification tasks where the input data may contain both sequential dependencies and spectra-spatial correlations.

Interestingly, LSTM model outperformed CNN model. This result is consistent with the nature of the mineral abundance data and inherent properties of the models. HS data used in mineral mapping and remote sensing frequently show local dependencies and sequential patterns across the spectral dimension. Such structured, ordered patterns are ideal for LSTM networks, which are designed to capture temporal or sequential dependencies efficiently.

CNNs on the other hand, are specifically designed to extract local spatial patterns from input using convolutional filters. Their strength is in detecting and learning spatially correlated features, which makes them especially useful when the dataset has high spatial structures, such as mineralogical changes throughout an area. However, their performance may be less obvious when temporal or sequential dependences are more important than spatial relationships. Additionally, CNNs can be sensitive to filter size and network depth, and may require extensive testing with kernel parameters, pooling techniques, and architectural design to achieve optimal performance.

The results of this study align with, and in several instances surpass, those published in relevant literature. For example, Boiger et al. (2024) developed a CNN-based regression model to predict mineral content from drill-core images, obtaining R^2 values of 0.609 for carbonate, 0.707 for silicate, and 0.811 for total clay content. Additionally, Don et al., 2024 used LSTM and hybrid CNN-LSTM model under prediction of soil organic carbon content in complex vegetation areas and the results showed that compared with common algorithmic models (RF, CNN, LSTM), the predictions of CNN-LSTM algorithm achieved high accuracy ($R^2 = 0.69$).

Overall, the DL models developed in this study performed comparably, if not better, than those in previous studies. The marginally superior performance of LSTM model over CNN model shows that sequential features are of greater significance in the dataset studied. Additionally, the findings show that hybridizing models might improve performance through complementary, especially when dealing with complex geospatial data.

4.6. Concluding Remarks

This chapter included the development and training of a hybrid CNN-LSTM model. In keeping with the methodology employed for training the separate models (CNN and LSTM), 80% of the available data were used to train hybrid CNN-LSTM model, and the remaining 20% set aside for testing. After training the hybrid model, all the three models were tested on the 20% of the data that were set aside for testing. The hybrid CNN-LSTM model had the lowest MAE of 0.0258, followed by the hybrid LSTM model with 0.0278 and CNN model with 0.0312. Additionally, uncertainty matrices of the three DL models were acquired, and hybrid CNN-LSTM had the lowest uncertainty compared to the stand-alone models.

CNN model explained 76% of the variance in the target variable during testing, with R^2 value of 0.76. LSTM model outperformed CNN model with R^2 value of 0.83, accounting for approximately 83% of the variation. However, CNN-LSTM model outperformed both the stand-alone models and acquired R^2 value of 0.84 (84%). To visualize the results, a scatter-plot of true versus predicted mineral abundance was acquired and LSTM and hybrid CNN-LSTM models showed overall good correlation with few outliers related to kaolin minerals.

Mineral abundance and mineral association maps were created to evaluate and contrast the performance of each model in greater depth. The mineral abundance maps in all the three models showed Fe-oxide minerals to be in more abundance, then followed by kaolin minerals. The abundance of white micas and carbonates were in a very lower proportion as compared to the Fe-oxide and kaolin minerals. Mineral association maps were acquired through KNN clustering, and 4 clusters were chosen. During clustering, predicted mineral abundance results for each model went through clustering and each sample had a cluster label based on the abundance of minerals a sample had, and this made the cluster meaning to be different as clustering was done on each of the predicted mineral abundance. The acquired mineral association maps were corresponding to each model mineral abundance map.

CHAPTER 5: DISCUSSION

The aim of this study was to improve mineral mapping by integrating drill-core HS and geochemical data using DL models. The objectives were (i) to evaluate and compare the performances of CNNs and RNNs in mineral identification, (ii) to quantify uncertainty in drill-core mineral mapping using integrated HS and geochemical datasets, and (iii) to develop and test a hybrid model that combines CNN and RNN models for improved mineral identification. To achieve these objectives, three DL models were used: a CNN, a LSTM network (a form of RNN), and hybrid CNN-LSTM. In all these models, geochemical and HS data were used as predictor variable and mineral abundance as the target variable. In this study, 80% of the data were used as a training set and the remaining 20% of the data used as a testing set. This is because the larger the training set, there more a model is able to learn patterns across the data, which is crucial because DL models require large amounts of data to generalize well (Sun et al., 2017).

After model testing, the findings showed that all three models had high prediction accuracy, as reflected by consistently low MAE values and uncertainty (Table 5.1). Among the models, hybrid CNN-LSTM outperformed the stand-alone models, followed by the LSTM and finally CNN. These findings demonstrate the ability of DL models to capture complex, non-linear correlations of a target variable with predictor variables, as well as the importance of hybrid DL models in enhancing mineral mapping performance. More broadly, the results demonstrated the ability of DL models to extract significant mineralogical information from high-dimensional datasets, improving the accuracy and reliability of drill-core mineral identification.

Table 5.1. Mineral abundance MAEs and overall MAE during model testing of CNN, LSTM and a hybrid CNN-LSTM models, and model uncertainty.

Model	Fe_ox_ai	Kaolin	Wmalsmai	Carbai3pfit	Overall MAE	Uncertainty
CNN	0.0321	0.0399	0.0186	0.0101	0.0312	0.000437
LSTM	0.0259	0.0372	0.0199	0.0124	0.0278	0.000594
CNN-LSTM	0.0244	0.0348	0.0178	0.0131	0.0258	0.000231

In this table, Fe_ox-ai represents iron oxide abundance, kaolin represents kaolin mineral abundance, Wmalsmai represents white mica abundance, and Carbai3pfit represents carbonate mineral abundance.

The superior performance of the hybrid architecture can be attributed to its ability to jointly exploit spatial–spectral feature extraction through convolutional layers and sequential dependencies along the drill core through recurrent units, enabling a more complete representation of geological continuity. The lower MAE and uncertainty of the hybrid model compared to the standalone models indicate the advantages of using multi-architecture systems when working with complex geological datasets such as HS data.

From an operational standpoint, intervals linked to higher uncertainty can be prioritized for further data collection or laboratory verification, making the quantified prediction uncertainty a useful tool for directing follow-up drilling and sampling tactics. Therefore, by separating zones that need more research from high-confidence mineral projections, using uncertainty measures into exploratory workflows enables risk-aware decision-making.

That the LSTM model outperformed CNN, suggest that RNN-based models are better suited to capture temporal and sequential structures in data. In comparison, while CNN model performed poorly, it added crucial pattern recognition capability. The stronger performance of LSTM relative to CNN further suggests that mineral distributions along drill cores exhibit

significant depth-dependent patterns, highlighting the importance of modelling stratigraphic and alteration-related trends in mineral systems. This becomes evident in the hybrid CNN-LSTM model, whereby combining CNN-derived spatial features with LSTM sequential learning led to improved overall outcomes, demonstrating the complementarity of the two models.

The results emphasize the importance of data integration in mineral mapping. The study showed that integrating HS and geochemical data in DL models can result in more reliable and accurate mineral mapping. This not only enhances mineral mapping resolution but also gives more powerful tools for mineral exploration and resource evaluation, as well as a methodological framework that can be applied to different types of ore deposits and geological settings.

Although most of the relevant studies have traditionally used classification methodologies, the findings of this study are closely aligned with the recent advances in the literature. For example, Li et al. (2023) demonstrated that an ensemble attention-based CNN outperformed CNNs in prediction accuracy; Agrawal and Govil (2023) reported that a mineral-CNN-LSTM framework achieved the highest prediction accuracy; Zhao et al. (2020) showed that hierarchical spatial-spectral extraction combined with LSTM achieved greater overall accuracy compared to CNN and LSTM; and Chen et al. (2025) found that the CNN-LSTM composite acquired lower MAE values relative to CNN, LSTM, DT, RF, and XGBoost models. Collectively, these studies support the results of this study that hybrid models consistently outperform stand-alone models, especially when applied to complex datasets.

Apart from model performance, the mineral abundance and mineral association maps produced in this study offer significant geological understanding. Based on the mineral abundance maps, Fe-oxide minerals are substantially more abundant than kaolin minerals, white micas, and carbonate minerals. This is consistent with the Rocklea Dome, a channel iron ore deposit in terms of geology (Haest et al., 2012; Laukamp et al., 2021). Therefore, the abundance of Fe-oxide supports the predictive accuracy of the studied DL models in reflecting the known mineralogical composition of the deposit. Crucially, high-resolution mapping and visualization of mineral distributions improves resource characterization and helps guide exploration targeting and ore quality evaluation. Practically, this shows how predictive modelling can provide a continuous and spatially coherent representation of mineralogical variability across drill-cores complementing and sometimes reducing the need for expensive and time-consuming geochemical analysis.

The quantity and quality of the drill-hole data used for analysis are the main sources of limitation in this study. The representativeness of the mineralogical predictions might have been limited by the quantity of samples used and the presence of censored values. Despite the fact that k-nearest neighbours' imputation was used to impute values that were below detection limit, this method might not adequately represent the underlying variability of the dataset. Additionally, only data from Rocklea Dome were used to train and test the predictive models (CNN, LSTM, and hybrid CNN-LSTM) created in this study. This might minimize the applicability of the findings to other channel iron deposits or geological settings, although the models can capture site-specific mineralogical features. Consequently, the results should be interpreted as demonstrating the feasibility of data-driven mineral mapping primarily within well-sampled channel iron systems rather than as universally transferable models. Future

applications would require retraining and validation using multi-deposit datasets and alternative imputation strategies to assess robustness across different geological environments.

In summary, this study showed that mineral mapping can be greatly enhanced by integrating HS and geochemical data using DL models (e.g., CNN, LSTM, and hybrid CNN-LSTM). The predictive strengths of DL models and dependability in identifying patterns of mineral abundance are demonstrated by the consistently low MAE values that were achieved. The results highlight the potential of integrating HS and geochemical data using DL as a reliable method for mineral mapping, despite the limitations of the analysis, which included the small quantity of the drill-hole data, presence of censored values in the data, and site-specific focus on Rocklea Dome. These imply that the techniques used here might be scaled and modified to improve mineral mapping across a wider range of ore deposit types with larger and more complex datasets. More broadly, the results indicate that integrating high-dimensional HS data and geochemical data with DL models allow the models to disentangle subtle mineralogical variations that may be difficult to resolve using single-data-source approaches.

CHAPTER 6: CONCLUSION, RECCOMENDATION AND FUTURE RESEARCH WORK

This study integrated drill-core hyperspectral (HS) and geochemical data using DL models to enhance drill-core mineral mapping. Three DL models were investigated: CNNs, LSTM, and hybrid CNN-LSTM. The objectives were to assess the predictive performance of CNN and LSTM models in predicting minerals, quantifying uncertainty of the models, and lastly, developing and testing a hybrid model that combines CNN and LSTM models for improved mineral identification.

The results showed that all the three models had high predictive performance, with low MAE and uncertainty. Among the models, hybrid CNN-LSTM outperformed the stand-alone models with MAE of 0.0258, followed by LSTM model with MAE of 0.0278, and lastly CNN model with MAE of 0.0312. For hybrid CNN-LSTM to outperform the stand-alone models, it demonstrated the benefits of combining spatial-spectral feature extraction with sequential dependency modelling, and for LSTM model to outperform CNN, it revealed that sequential features in the data are crucial for accurate mineral mapping.

The mineral abundance and association maps created using the prediction results of all the three models indicated that Fe-oxide minerals are more abundant compared to kaolin, white mica, and carbonate minerals. This is consistent with the geological classification of the Rocklea Dome as a channel iron ore deposit.

From a practical aspect, this study demonstrates how DL approaches can complement traditional mineral exploration methodologies, and by integrating HS and geochemical data using DL models, it can provide a rapid and cost-effective mineralogical insight, minimizing the need for labour-intensive geochemical assays and allowing geologists to visualize mineral distribution on a smaller scale, which enhances mineral mapping.

Following these results, a number of recommendations can be made. First, the integration of HS and geochemical data with DL models could be improved as complementary technique in mineral exploration and resource assessment. Expanding datasets to include more drill-holes, deposit with various geological settings, and more samples would contribute to improved model robustness and generalizability. Second, further research into sophisticated DL architectures, such as attention-based models, graph neural networks or transformer networks, could improve predictive accuracy and reduce uncertainty.

To further verify model transferability and resilience, future research should concentrate on increasing the geological and spatial diversity of datasets by adding more drill holes, deposits from various mineral systems, and multi-scale sampling techniques. Predictive performance and geological interpretability may also be improved by including additional supplementary data sources, such as whole-rock geochemistry, petrographic observations, and downhole geophysics.

In conclusion, this study has shown that integrating HS and geochemical data using DL models, particularly hybrid CNN-LSTM model, offers a promising pathway for enhancing mineral mapping. These models not only enhance predictive accuracy but also produce geologically

significant results that can help exploration decisions and future advances in computational mineralogy.

REFERENCES

Abedini, M., Ziaii, M., Timkin, T. and Pour, A.B., 2023. Machine learning (ML)-based copper mineralization prospectivity mapping (MPM) using mining geochemistry method and remote sensing satellite data. *Remote Sensing*, *15*(15), p.3708.

Acosta, I.C.C., Khodadadzadeh, M., Tolosana-Delgado, R. and Gloaguen, R., 2020. Drill-core hyperspectral and geochemical data integration in a superpixel-based machine learning framework. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 13, pp.4214–4228.

Acosta, I.C.C., Khodadadzadeh, M., Tusa, L., Ghamisi, P. and Gloaguen, R., 2019. A machine learning framework for drill-core mineral mapping using hyperspectral and high-resolution mineralogical data fusion. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 12(12), pp.4829–4842.

Agrawal, N. and Govil, H., 2023. A deep residual convolutional neural network for mineral classification. Advances in Space Research, 71(8), pp.3186–3202.

Agrawal, N., Govil, H., Mishra, G., Gupta, M. and Srivastava, P.K., 2023. Evaluating the performance of PRISMA shortwave infrared imaging sensor for mapping hydrothermally altered and weathered minerals using the machine learning paradigm. Remote Sensing, 15(12), p.3133.

Allison, P., 2013. What's the best R-squared for logistic regression. Statistical Horizons, 13, pp.126–129.

Alzubaidi, L., Zhang, J., Humaidi, A.J., Al-Dujaili, A., Duan, Y., Al-Shamma, O. and Farhan, L., 2021. Review of deep learning: concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8, pp.1–74.

Anand, R., Andrew, J. and Makki, I., 2024. Optimizing land mine detection across diverse mining environments: A hyperspectral data approach with regression models. International Journal of Intelligent Networks, 5, pp.351–363.

Babu, G.H. and Venkatram, N., 2020. A survey on analysis and implementation of state-of-the-art haze removal techniques. Journal of Visual Communication and Image Representation, 72, p.102912.

Barker, R.D., Barker, S.L., Wilson, S. and Stock, E.D., 2021. Quantitative mineral mapping of drill core surfaces I: A method for μXRF mineral calculation and mapping of hydrothermally altered, fine-grained sedimentary rocks from a Carlin-type gold deposit. Economic Geology, 116(4), pp.803–819.

Bedini, E., 2017. The use of hyperspectral remote sensing for mineral exploration: A review. Journal of Hyperspectral Remote Sensing, 7(4), pp.189–211.

Bioucas-Dias, J.M., Plaza, A., Camps-Valls, G., Scheunders, P., Nasrabadi, N. and Chanussot, J., 2013. Hyperspectral remote sensing data analysis and future challenges. *IEEE Geoscience and remote sensing magazine*, *1*(2), pp.6-36.

Boiger, R., Churakov, S.V., Ballester Llagaria, I., Kosakowski, G., Wüst, R. and Prasianakis, N.I., 2024. Direct mineral content prediction from drill core images via transfer learning. Swiss Journal of Geosciences, 117(1), p.8.

Bigdeli, A., Maghsoudi, A. and Ghezelbash, R., 2022. Application of self-organizing map (SOM) and K-means clustering algorithms for portraying geochemical anomaly patterns in Moalleman district, NE Iran. Journal of Geochemical Exploration, 233, p.106923.

Chen, C., Zhang, P., Zhang, H., Dai, J., Yi, Y. and Zhang, Y., 2020. Deep learning on computational-resource-limited platforms: A survey. Mobile Information Systems, 2020(1), p.8454327.

Chen, H., Xia, M., Zhang, Y., Zhao, R., Song, B. and Bai, Y., 2025. Iron ore information extraction based on CNN-LSTM composite deep learning model. IEEE Access.

Chen, Y., Lin, Z., Zhao, X., Wang, G. and Gu, Y., 2014. Deep learning-based classification of hyperspectral data. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 7(6), pp.2094–2107.

Choros, K.A., Job, A.T., Edgar, M.L., Austin, K.J. and McAree, P.R., 2022. Can hyperspectral imaging and neural network classification be used for ore grade discrimination at the point of excavation?. Sensors, 22(7), p.2687.

Contreras Acosta, I.C., Khodadadzadeh, M. and Gloaguen, R., 2021. Resolution enhancement for drill-core hyperspectral mineral mapping. Remote Sensing, 13(12), p.2296.

Cornuéjols, A., Wemmert, C., Gançarski, P. and Bennani, Y., 2018. Collaborative clustering: Why, when, what and how. Information Fusion, 39, pp.81–95.

Davis, J., Zhu, J., Oldfather, J., MacDonald, S., Trzaskowski, M. and Kelsen, M., 2020. Quantifying uncertainty in deep learning systems. Aws Perspective Guidance.

De la Rosa, R., Khodadadzadeh, M., Tusa, L., Kirsch, M., Gisbert, G., Tornos, F. and Gloaguen, R., 2021. Mineral quantification at deposit scale using drill-core hyperspectral data: A case study in the Iberian Pyrite Belt. Ore Geology Reviews, 139, p.104514.

Dell'Aversana, P., 2023. An integrated deep learning framework for classification of mineral thin sections and other geo-data, a tutorial. Minerals, 13(5), p.584.

Dong, Z., Yao, L., Bao, Y., Zhang, J., Yao, F., Bai, L. and Zheng, P., 2024. Prediction of soil organic carbon content in complex vegetation areas based on CNN-LSTM model. Land, 13(7), p.915.

Du, X., Zhou, K., Cui, Y., Wang, J. and Zhou, S., 2021. Mapping mineral prospectivity using a hybrid genetic algorithm–support vector machine (GA–SVM) model. ISPRS International Journal of Geo-Information, 10(11), p.766.

Duan, P., Lai, J., Ghamisi, P., Kang, X., Jackisch, R., Kang, J. and Gloaguen, R., 2020. Component decomposition-based hyperspectral resolution enhancement for mineral mapping. Remote Sensing, 12(18), p.2903.

Fabre, C., Devismes, D., Moncayo, S., Pelascini, F., Trichard, F., Lecomte, A. and Motto-Ros, V., 2018. Elemental imaging by laser-induced breakdown spectroscopy for the geological characterization of minerals. Journal of Analytical Atomic Spectrometry, 33(8), pp.1345–1353.

Fandrich, R., Gu, Y., Burrows, D. and Moeller, K., 2007. Modern SEM-based mineral liberation analysis. International Journal of Mineral Processing, 84(1–4), pp.310–320.

Farahbakhsh, E., Goel, D., Pimparkar, D., Müller, R.D. and Chandra, R., 2025. Convolutional neural networks for mineral prospecting through alteration mapping with remote sensing data. PFG–Journal of Photogrammetry, Remote Sensing and Geoinformation Science, pp.1–22.

Ferreira, A.J. and Figueiredo, M.A., 2012. Boosting algorithms: A review of methods, theory, and applications. In: Ensemble machine learning: Methods and applications, pp.35–85.

Filzmoser, P., Hron, K. and Reimann, C., 2009. Univariate statistical analysis of environmental (compositional) data: problems and possibilities. Science of the Total Environment, 407(23), pp.6100–6108.

Galdames, F.J., Perez, C.A., Estevez, P.A. and Adams, M., 2022. Rock lithological instance classification by hyperspectral images using dimensionality reduction and deep learning. Chemometrics and Intelligent Laboratory Systems, 224, p.104538.

Ghezelbash, R., Maghsoudi, A., Shamekhi, M., Pradhan, B. and Daviran, M., 2023. Genetic algorithm to optimize the SVM and K-means algorithms for mapping of mineral prospectivity. Neural Computing and Applications, 35(1), pp.719–733.

Graves, A., 2013. Generating sequences with recurrent neural networks. arXiv preprint arXiv:1308.0850.

Grunsky, E.C., 2010. The interpretation of geochemical survey data.

Grunsky, E.C. and Caritat, P.D., 2020. State-of-the-art analysis of geochemical data for mineral exploration. Geochemistry: Exploration, Environment, Analysis, 20(2), pp.217–232.

Grunsky, E.C. and de Caritat, P., 2017. Advances in the use of geochemical data for mineral exploration. In: Proceedings of exploration, vol. 17, pp.441–456.

Grunsky, E.C. and Smee, B.W., 1999. The differentiation of soil types and mineralization from multi-element geochemistry using multivariate methods and digital topography. Journal of Geochemical Exploration, 67(1–3), pp.287–299.

Guha, A., 2020. Mineral exploration using hyperspectral data. In: Hyperspectral Remote Sensing, pp.293–318. Elsevier.

Guo, Y., Wang, X., Xiao, P. and Xu, X., 2020. An ensemble learning framework for convolutional neural network based on multiple classifiers. Soft Computing, 24(5), pp.3727–3735.

Haest, M., Cudahy, T., Laukamp, C. and Gregory, S., 2012a. Quantitative mineralogy from infrared spectroscopic data. I. Validation of mineral abundance and composition scripts at the Rocklea channel iron deposit in Western Australia. Economic Geology, 107(2), pp.209–228.

Haest, M., Cudahy, T., Laukamp, C. and Gregory, S., 2012b. Quantitative mineralogy from infrared spectroscopic data. II. Three-dimensional mineralogical characterization of the Rocklea channel iron deposit, Western Australia. Economic Geology, 107(2), pp.229–249.

Hajaj, S., El Harti, A., Pour, A.B., Jellouli, A., Adiri, Z. and Hashim, M., 2024. A review on hyperspectral imagery application for lithological mapping and mineral prospecting: Machine learning techniques and future prospects. Remote Sensing Applications: Society and Environment, p.101218.

Hamedianfar, A., Laakso, K., Middleton, M., Törmänen, T., Köykkä, J. and Torppa, J., 2023. Leveraging high-resolution long-wave infrared hyperspectral laboratory imaging data for mineral identification using machine learning methods. Remote Sensing, 15(19), p.4806.

Harraden, C.L., Cracknell, M.J., Lett, J., Berry, R.F., Carey, R. and Harris, A.C., 2019. Automated core logging technology for geotechnical assessment: A study on core from the Cadia East porphyry deposit. Economic Geology, 114(8), pp.1495–1511.

Harris, J.R., Wilkinson, L. and Grunsky, E.C., 2000. Effective use and interpretation of lithogeochemical data in regional mineral exploration programs: application of geographic information systems (GIS) technology. Ore Geology Reviews, 16(3–4), pp.107–143.

Hoseinzade, Z., Shojaei, M., Khademi, F., Mokhtari, A.R. and Saremi, M., 2025. Integration of deep learning models for mineral prospectivity mapping: a novel Bayesian index approach to reducing uncertainty in exploration. Modeling Earth Systems and Environment, 11(3), p.161.

Hüllermeier, E. and Waegeman, W., 2021. Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods. Machine Learning, 110(3), pp.457–506.

Jierula, A., Wang, S., Oh, T.M. and Wang, P., 2021. Study on accuracy metrics for evaluating the predictions of damage locations in deep piles using artificial neural networks with acoustic emission data. Applied Sciences, 11(5), p.2314.

Kiranyaz, S., Avci, O., Abdeljaber, O., Ince, T., Gabbouj, M. and Inman, D.J., 2021. 1D convolutional neural networks and applications: A survey. Mechanical Systems and Signal Processing, 151, p.107398.

Kirsch, M., Lorenz, S., Zimmermann, R., Tusa, L., Möckel, R., Hödl, P. and Gloaguen, R., 2018. Integration of terrestrial and drone-borne hyperspectral and photogrammetric sensing methods for exploration mapping and mining monitoring. Remote Sensing, 10(9), p.1366.

Khammar, F., Yousefi, S. and Joonaghani, S.A., 2021. Analysis of lithogeochemical data using log-ratio transformations and CA fractal to separate geochemical anomalies in Tak-Talar, Iran. Arabian Journal of Geosciences, 14, pp.1–15.

Kokaly, R.F., Clark, R.N., Swayze, G.A., Livo, K.E., Hoefen, T.M., Pearson, N.C. and Klein, A.J., 2017. USGS spectral library version 7 (No. 1035). US Geological Survey.

Krupnik, D. and Khan, S., 2019. Close-range, ground-based hyperspectral imaging for mining applications at various scales: Review and case studies. Earth-Science Reviews, 198, p.102952.

Kruse, F.A., 2012. Mapping surface mineralogy using imaging spectrometry. Geomorphology, 137(1), pp.41–56.

Kruse, F.A., Bedell, R.L., Taranik, J.V., Peppin, W.A., Weatherbee, O. and Calvin, W.M., 2012. Mapping alteration minerals at prospect, outcrop and drill core scales using imaging spectrometry. International Journal of Remote Sensing, 33(6), pp.1780–1798.

Krause, B., Lu, L., Murray, I. and Renals, S., 2016. Multiplicative LSTM for sequence modelling. arXiv preprint arXiv:1609.07959.

Kutzke, A., Eichstaedt, H. and Kahnt, R., 2022. Potential of hyperspectral-based geochemical predictions with neural networks for strategic and regional exploration improvement. Australian Journal of Earth Sciences, 69(8), pp.1197–1206.

Laukamp, C., Haest, M. and Cudahy, T., 2021. The Rocklea Dome 3D mineral mapping test data set. Earth System Science Data, 13(3), pp.1371–1383.

Li, S., Wang, X. and Barnard, A.S., 2024. Diverse explanations from data-driven and domain-driven perspectives in the physical sciences. Machine Learning: Science and Technology.

Li, Q., Chen, G. and Luo, L., 2023. Mineral prospectivity mapping using attention-based convolutional neural network. Ore Geology Reviews, 156, p.105381.

Li, Y.H., Tan, X., Zhang, W., Jiao, Q.B., Xu, Y.X., Li, H. and Fang, Y.P., 2021. Research and application of several key techniques in hyperspectral image preprocessing. Frontiers in Plant Science, 12, p.627865.

Li, Z., Liu, F., Yang, W., Peng, S. and Zhou, J., 2021. A survey of convolutional neural networks: analysis, applications, and prospects. IEEE Transactions on Neural Networks and Learning Systems, 33(12), pp.6999–7019.

Lin, N., Liu, H., Li, G., Wu, M., Li, D., Jiang, R. and Yang, X., 2022. Extraction of mineralized indicator minerals using ensemble learning model optimized by SSA based on hyperspectral image. Open Geosciences, 14(1), pp.1444–1465.

Liu, C., Li, M., Zhang, Y., Han, S. and Zhu, Y., 2019. An enhanced rock mineral recognition method integrating a deep learning model and clustering algorithm. Minerals, 9(9), p.516.

Liu, P., Wang, L., Ranjan, R., He, G. and Zhao, L., 2022. A survey on active deep learning: From model driven to data driven. ACM Computing Surveys, 54(10s), pp.1–34.

Lipton, Z.C., Berkowitz, J. and Elkan, C., 2015. A critical review of recurrent neural networks for sequence learning. arXiv preprint arXiv:1506.00019.

Lobo, A., Garcia, E., Barroso, G., Martí, D., Fernandez-Turiel, J.L. and Ibáñez-Insa, J., 2021. Machine learning for mineral identification and ore estimation from hyperspectral imagery in tin–tungsten deposits: simulation under indoor conditions. Remote Sensing, 13(16), p.3258.

Lwakatare, L.E., Raj, A., Crnkovic, I., Bosch, J. and Olsson, H.H., 2020. Large-scale machine learning systems in real-world industrial settings: A review of challenges and solutions. Information and Software Technology, 127, p.106368.

Mahdiyanfar, H. and Salimi, A., 2022. Fractal modeling of geochemical mineralization prospectivity index based on centered log-ratio transformed data for geochemical targeting: a case study of Cu porphyry mineralization. Journal of Mining and Environment, 13(3), pp.821–838.

Mathew, A., Amudha, P. and Sivakumari, S., 2021. Deep learning techniques: an overview. In: *Advanced Machine Learning Technologies and Applications: Proceedings of AMLTA 2020*, pp.599–608.

McClarren, R.G., McClarren, P. and Penrose, R., 2018. Uncertainty quantification and predictive computational science.

Mienye, I.D. and Sun, Y., 2022. A survey of ensemble learning: Concepts, algorithms, applications, and prospects. IEEE Access, 10, pp.99129–99149.

Nikonow, W., Rammlmair, D., Meima, J.A. and Schodlok, M.C., 2019. Advanced mineral characterization and petrographic analysis by µ-EDXRF, LIBS, HSI and hyperspectral data merging. *Mineralogy and Petrology*, *113*(3), pp.417-431.

Okada, N., Bino Sinaice, B., Kim, J., Nozaki, H., Takizawa, K., Owada, N. and Kawamura, Y., 2024. Advanced mineralogical classification and concentration estimation in mining with MATLAB-powered hyper-spectral imaging and machine learning. International Journal of Mining, Reclamation and Environment, 38(10), pp.851–871.

Okada, N., Maekawa, Y., Owada, N., Haga, K., Shibayama, A. and Kawamura, Y., 2020. Automated identification of mineral types and grain size using hyperspectral imaging and deep learning for mineral processing. Minerals, 10(9), p.809.

Parsa, M., Shirazy, A., Shirazi, A. and Pour, A.B., 2023. Processing and interpretation of geochemical data for mineral exploration. In: *Geospatial Analysis Applied to Mineral Exploration*, pp.171–188. Elsevier.

Peyghambari, S. and Zhang, Y., 2021. Hyperspectral remote sensing in lithological mapping, mineral exploration, and environmental geology: an updated review. Journal of Applied Remote Sensing, 15(3), pp.031501–031501.

Puchhammer, P., Kalubowila, C., Braus, L., Pospiech, S., Sarala, P. and Filzmoser, P., 2024. A performance study of local outlier detection methods for mineral exploration with geochemical compositional data. Journal of Geochemical Exploration, 258, p.107392.

Puzyrev, V. and Duuring, P., 2025. Uncertainty quantification of geochemical data imputation using Monte Carlo dropout. Journal of Geochemical Exploration, 272, p.107695.

Qaderi, S., Maghsoudi, A., Yousefi, M. and Pour, A.B., 2025. Translation of mineral system components into time step-based ore-forming events and evidence maps for mineral exploration: Intelligent mineral prospectivity mapping through adaptation of recurrent neural networks and random forest algorithm. Ore Geology Reviews, p.106537.

Quigley, M.C., Bennetts, L.G., Durance, P., Kuhnert, P.M., Lindsay, M.D., Pembleton, K.G. and White, C.J., 2019. The provision and utility of earth science to decision-makers: synthesis and key findings. Environment Systems and Decisions, 39(3), pp.349–367.

Ragab, M.G., Abdulkadir, S.J., Aziz, N., Alhussian, H., Bala, A. and Alqushaibi, A., 2021. An ensemble one-dimensional convolutional neural network with Bayesian optimization for environmental sound classification. Applied Sciences, 11(10), p.4660.

Rainio, O., Teuho, J. and Klén, R., 2024. Evaluation metrics and statistical tests for machine learning. Scientific Reports, 14(1), p.6086.

Raj, R. and Kos, A., 2025. An extensive study of convolutional neural networks: applications in computer vision for improved robotics perceptions. Sensors, 25(4), p.1033.

Reimann, C., Filzmoser, P. and Garrett, R.G., 2002. Factor analysis applied to regional geochemical data: problems and possibilities. Applied Geochemistry, 17(3), pp.185–206.

Rodger, A., Fabris, A. and Laukamp, C., 2021. Feature extraction and clustering of hyperspectral drill core measurements to assess potential lithological and alteration boundaries. Minerals, 11(2), p.136.

Rotem, A., Vidal, A., Pfaff, K., Tenorio, L., Chung, M., Tharalson, E. and Monecke, T., 2023. Interpretation of hyperspectral shortwave infrared core scanning data using SEM-based automated mineralogy: a machine learning approach. Geosciences, 13(7), p.192.

Sarker, I.H., 2021. Deep learning: a comprehensive overview on techniques, taxonomy, applications and research directions. SN Computer Science, 2(6), pp.1–20.

Singh, R.K., Ray, D. and Sarkar, B.C., 2018. Recurrent neural network approach to mineral deposit modelling. In: *2018 4th International Conference on Recent Advances in Information Technology (RAIT)*, pp.1–5. IEEE.

Sudharsan, S., Hemalatha, R. and Radha, S., 2019. A survey on hyperspectral imaging for mineral exploration using machine learning algorithms. In: *2019 International Conference on Wireless Communications Signal Processing and Networking (WiSPNET)*, pp.206–212. IEEE.

Sun, C., Shrivastava, A., Singh, S. and Gupta, A., 2017. Revisiting unreasonable effectiveness of data in deep learning era. In: *Proceedings of the IEEE International Conference on Computer Vision*, pp.843–852.

Thity, N.T., Rahman, A., Dulmini, A., Yasmin, M.N. and Rois, R., 2025. An illustration of multi-class ROC analysis for predicting internet addiction among university students. PLoS One, 20(7), p.e0325855.

Tian, C., Ma, J., Zhang, C. and Zhan, P., 2018. A deep neural network model for short-term load forecast based on long short-term memory network and convolutional neural network. Energies, 11(12), p.3493.

Tripathi, P. and Garg, R.D., 2021. Feature extraction of DESIS and PRISMA hyperspectral remote sensing datasets for geological applications. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, 44, pp.169–173.

Tuşa, L., Khodadadzadeh, M., Contreras, C., Rafiezadeh Shahi, K., Fuchs, M., Gloaguen, R. and Gutzmer, J., 2020. Drill-core mineral abundance estimation using hyperspectral and high-resolution mineralogical data. Remote Sensing, 12(7), p.1218.

Valls, R., 2023. Unraveling the Earth's secrets: How to interpret geochemical data to identify alteration zones.

Van der Meer, F.D., Van der Werff, H.M., Van Ruitenbeek, F.J., Hecker, C.A., Bakker, W.H., Noomen, M.F. and Woldai, T., 2012. Multi- and hyperspectral geologic remote sensing: A review. International Journal of Applied Earth Observation and Geoinformation, 14(1), pp.112–128.

Wang, Z., Li, T. and Zuo, R., 2024. Leucogranite mapping via convolutional recurrent neural networks and geochemical survey data in the Himalayan orogen. Geoscience Frontiers, 15(1), p.101715.

Wang, W., Xie, S. and Carranza, E.J.M., 2023. Introduction to the thematic collection: applications of innovations in geochemical data analysis. Geochemistry: Exploration, Environment, Analysis, 23(1), p.geochem2022-058.

Wang, Z. and Zuo, R., 2022. Mineral prospectivity mapping using a joint singularity-based weighting method and long short-term memory network. Computers & Geosciences, 158, p.104974.

West, R.M., 2022. Best practice in statistics: The use of log transformation. Annals of Clinical Biochemistry, 59(3), pp.162–165.

Xiong, Y., Zuo, R. and Carranza, E.J.M., 2018. Mapping mineral prospectivity through big data analytics and a deep learning algorithm. Ore Geology Reviews, 102, pp.811–817.

Xu, Y., Li, Z., Xie, Z., Cai, H., Niu, P. and Liu, H., 2021. Mineral prospectivity mapping by deep learning method in Yawan-Daqiao area, Gansu. Ore Geology Reviews, 138, p.104316.

Yang, F. and Zuo, R., 2024. Geologically constrained convolutional neural network for mineral prospectivity mapping. Mathematical Geosciences, 56(8), pp.1605–1628.

Yang, J., Qu, J., Mi, Q. and Li, Q., 2020. A CNN-LSTM model for tailings dam risk prediction. IEEE Access, 8, pp.206491–206502.

Yin, J. and Li, N., 2022. Ensemble learning models with a Bayesian optimization algorithm for mineral prospectivity mapping. Ore Geology Reviews, 145, p.104916.

Yin, S., Li, N., Xiao, K., Song, X., Yin, J. and Wang, C., 2024. Mineral prospectivity mapping using knowledge embedding and explainable ensemble learning: A case study of the Keeryin ore concentration in Sichuan, China. Ore Geology Reviews, p.106017.

Yin, B., Zuo, R. and Xiong, Y., 2022. Mineral prospectivity mapping via gated recurrent unit model. Natural Resources Research, pp.1–15.

Yokoya, N., Chan, J.C.W. and Segl, K., 2016. Potential of resolution-enhanced hyperspectral data for mineral mapping using simulated EnMAP and Sentinel-2 images. Remote Sensing, 8(3), p.172.

Younesi, A., Ansari, M., Fazli, M., Ejlali, A., Shafique, M. and Henkel, J., 2024. A comprehensive survey of convolutions in deep learning: Applications, challenges, and future trends. IEEE Access, 12, pp.41180–41218.

Yousefi, M., Lindsay, M.D. and Kreuzer, O., 2024. Mitigating uncertainties in mineral exploration targeting: Majority voting and confidence index approaches in the context of an exploration information system (EIS). Ore Geology Reviews, 165, p.105930.

Yu, Z., Li, B. and Wang, X., 2024. Mineral prospectivity mapping susceptibility evaluation based on interpretable ensemble learning. Ore Geology Reviews, p.106248.

Zhang, C., Yi, M., Ye, F., Xu, Q., Li, X. and Gan, Q., 2022. Application and evaluation of deep neural networks for airborne hyperspectral remote sensing mineral mapping: a case study of the Baiyanghe uranium deposit in northwestern Xinjiang, China. Remote Sensing, 14(20), p.5122.

Zhang, H., Xie, M., Dan, S., Li, M., Li, Y., Yang, D. and Wang, Y., 2024. Optimization of feature selection in mineral prospectivity using ensemble learning. Minerals, 14(10), p.970.

Zhao, H., Deng, K., Li, N., Wang, Z. and Wei, W., 2020. Hierarchical spatial-spectral feature extraction with long short-term memory (LSTM) for mineral identification using hyperspectral imagery. Sensors, 20(23), p.6854.

Zhao, C., Zhao, J., Wang, W., Yuan, C. and Tang, J., 2024. A novel hybrid ensemble model for mineral prospectivity prediction: A case study in the Malipo W-Sn mineral district, Yunnan Province, China. Ore Geology Reviews, p.106001.

Zhao, H., Zhang, Y., Xu, Y., Shao, Y., Chen, X., Hao, J. and Wang, X., 2024. Machine learning model for deep exploration: Utilizing short wavelength infrared (SWIR) of hydrothermal alteration minerals in the Qianchen gold deposit, Jiaodong Peninsula, Eastern China. Ore Geology Reviews, 168, p.106060.

Zhao, Y., Xu, H., Yang, T., Wang, S. and Sun, D., 2021. A hybrid recognition model of microseismic signals for underground mining based on CNN and LSTM networks. Geomatics, Natural Hazards and Risk, 12(1), pp.2803–2834.

Zhou, Y., Wang, Y., Su, J., Wen, Z., Zhang, P. and Zhang, W., 2025. EMSNet: Efficient multimodal symmetric network for semantic segmentation of urban scene from remote sensing imagery. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing.

Zhu, W., Zeng, N. and Wang, N., 2010. Sensitivity, specificity, accuracy, associated confidence interval and ROC analysis with practical SAS implementations. NESUG Proceedings: Health Care and Life Sciences, Baltimore, Maryland, 19, p.67.

Zuo, R., Kreuzer, O.P., Wang, J., Xiong, Y., Zhang, Z. and Wang, Z., 2021. Uncertainties in GIS-based mineral prospectivity mapping: Key types, potential impacts and possible solutions. Natural Resources Research, 30, pp.3059–3079.

Zuo, R., Xiong, Y., Wang, J. and Carranza, E.J.M., 2019. Deep learning and its application in geochemical mapping. Earth-Science Reviews, 192, pp.1–14.

Zuo, R. and Xu, Y., 2023. Graph deep learning model for mapping mineral prospectivity. Mathematical Geosciences, 55(1), pp.1–21.

APPENDICES

Appendix A: Hyperspectral Data Extraction

```
import json
import urllib.parse
import urllib.request
from pathlib import Path

# using the swagger api to get the files from here programatically this first call here is to list all the available files
url =  'https://data.csiro.au/dap/ws/v2/collections/44783v1/data'
with urllib.request.urlopen( url ) as response:
    response_text = response.read()
# the variable response_text is returned as a json package we are going to parse
# it with the python json library
c3dmm_files = json.loads(response_text)
base_dir = Path('data/')

# list of the target files that were needed to
# do this experiment
target_files = ['RC_data.bip','RC_data.tsg','Rocklea_Assay_MMX.xlsx']

# loop over each of the files in the json
# if already downloaded them then don’t download the files again.
for i in c3dmm_files['file']:
    # pathlib makes path handling simple I prefer it generally over the os library
    fname = Path(i['filename'])
    url = i['link']['href']
    # if the file names is in the list of target files let's download
    if fname.name in target_files:
        outfile = base_dir.joinpath(fname)
        # check if the folder exists if not make it
        if not outfile.parents[0].exists():
            outfile.parent.mkdir(parents=True,exist_ok=True)
        # check if the file exists if it does then don't download
        if not outfile.exists():
            # download the file
```

```
        with urllib.request.urlopen( url ) as response:
            response = response.read()
            with open(outfile,'wb') as file:
                file.write(response)

import pandas as pd
from pytsg.parse_tsg import read_tsg_bip_pair

assay_path = base_dir.joinpath('drill hole data/drillcore geochem/Rocklea_Assay_MMX.xlsx')
assay_data = pd.read_excel(assay_path)
tsg_file = base_dir.joinpath('drill hole data/RC_hyperspectral_geochem/RC_data.tsg')
bip_file = base_dir.joinpath('drill hole data/RC_hyperspectral_geochem/RC_data.bip')
nir = read_tsg_bip_pair(tsg_file, bip_file,'nir')

import numpy as np
import matplotlib.pyplot as plt
from scipy import spatial

# Get colormap
steps = 100
cmap = plt.get_cmap('viridis')  # Fixed line

# Get spectra and shape info
n_spectra, n_wavelengths = nir.spectra.shape

tmp_spec = []
for i in range(min(steps, n_spectra)):  # Ensure we don't go out of bounds
    spec = nir.spectra[i, :]

    # Combine wavelength and spectrum for convex hull
    spec_xy = np.vstack([nir.wavelength, spec]).T

    # Compute convex hull
    chull = spatial.ConvexHull(spec_xy)

    # Sort vertices and interpolate hull
```

```
    sort_v = np.sort(chull.vertices)
    hull = np.interp(np.arange(n_wavelengths), sort_v, spec[sort_v])

    # Only keep spectra where the original is not above the hull
    if not np.any(spec - hull > 0):
        tmp_spec.append(spec - hull)

# Stack and sort the filtered spectra
if tmp_spec:
    spec = np.vstack(tmp_spec)
    spec = np.sort(spec, axis=0)

    # Plotting
    fig, ax = plt.subplots()
    ax.set_facecolor("grey")
    for i in range(spec.shape[0]):
        ax.plot(nir.wavelength, spec[i, :] + (i / 100), color=cmap(i / steps))  # Fixed coloring line

    plt.title('Stacked Spectra')
    plt.ylabel('Reflectance (arbitrary units)')
    plt.xlabel('Wavelength')
    plt.show()
```

Appendix B: Hyperspectral Data Transformation

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
import os

#Step 1: Loading reflectance data
input_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Keystone_reflectance_spectra_cleaned.xlsx'
df = pd.read_excel(input_path)

# Step 2: Select columns from 380 to 2500
# Converting column names to string just in case
df.columns = df.columns.astype(str)

# Filtering columns that are numeric and in the desired range
reflectance_columns = [col for col in df.columns if col.isdigit() and 380 <= int(col) <= 2500]

# Extracting reflectance data
reflectance_data = df[reflectance_columns]

# Converting to numeric (in case there are strings like 'NULL', etc.)
reflectance_data = reflectance_data.apply(pd.to_numeric, errors='coerce')

# Dropping rows where all reflectance values are NaN
reflectance_data = reflectance_data.dropna(how='all')

# Optional: Fill remaining NaNs with column means
reflectance_data = reflectance_data.fillna(reflectance_data.mean())

# Step 3: Applying PCA
pca = PCA(n_components=2)
pca_scores = pca.fit_transform(reflectance_data)

# Step 4: Plotting PCA Scores
```

```
plt.figure(figsize=(8, 5))
scatter = plt.scatter(pca_scores[:, 0], pca_scores[:, 1], c=np.arange(len(pca_scores)), cmap='viridis')
plt.xlabel('PC1')
plt.ylabel('PC2')
plt.title('PCA Scores of Reflectance Spectra')
plt.colorbar(scatter, label='Sample Index')
plt.grid(True)
plt.tight_layout()
plt.show()

#Step 5: Plotting PCA Loadings
wavelengths = [int(w) for w in reflectance_columns]
plt.figure(figsize=(10, 4))
plt.plot(wavelengths, pca.components_[0], label='PC1', color='darkorange')
plt.plot(wavelengths, pca.components_[1], label='PC2', color='steelblue')
plt.title('PCA Loadings for Reflectance Spectra')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Loading Value')
plt.legend()
plt.grid(True)
plt.tight_layout()
plt.show()

# Step 6: Saving PCA Scores and Loadings
output_dir = os.path.dirname(input_path)

print(f" PCA scores saved to: {scores_path}")
```

Appendix C: Geochemical Data Transformation

```
import pandas as pd
import numpy as np
from sklearn.impute import KNNImputer

# Step 1: Loading the Excel file
file_path = "C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/RKC448.xlsx"
df = pd.read_excel(file_path, sheet_name="Sheet1")

# Step 2: Storing non-numeric columns (like sample names)
non_numeric = df.select_dtypes(exclude=[np.number])

# Step 3: Selecting only numeric columns
numeric_df = df.select_dtypes(include=[np.number])

# Step 4: Replacing 0 and 'LNR' with NaN
numeric_df = numeric_df.replace(0, np.nan)
numeric_df = numeric_df.replace("LNR", np.nan)

# Step 5: Selecting columns from Fe to TiO2 (inclusive)
fe_to_tio2_cols = numeric_df.loc[:, 'Fe':'TiO2']

# Step 6: Applying KNN only to those columns
imputer = KNNImputer(n_neighbors=5)
imputed_part = pd.DataFrame(imputer.fit_transform(fe_to_tio2_cols), columns=fe_to_tio2_cols.columns)

# Step 7: Replacing original Fe–TiO2 columns with imputed values
numeric_df.update(imputed_part)

# Step 8: Combining numeric and non-numeric
final_df = pd.concat([non_numeric.reset_index(drop=True), numeric_df.reset_index(drop=True)], axis=1)

# Step 9: Saving to Excel
output_path = "C:/Users/20191/Documents/imputed_Fe_to_TiO2_data.xlsx"
final_df.to_excel(output_path, index=False)
```

```
print(" Partial KNN Imputation complete. File saved to:", output_path)

import pandas as pd
import numpy as np
from skbio.stats.composition import closure, clr

# Step 1: Loading the KNN-imputed Excel file
file_path = "C:/Users/20191/Documents/imputed_Fe_to_TiO2_data.xlsx"
df = pd.read_excel(file_path)

# Step 2: Extracting the numeric part and store non-numeric separately
non_numeric = df.select_dtypes(exclude=[np.number])
numeric_df = df.select_dtypes(include=[np.number])

# Step 3: Selecting the compositional subset (Fe to TiO2)
comp_data = numeric_df.loc[:, 'Fe':'TiO2']

# Step 4: Ensuring all values are positive
if (comp_data <= 0).any().any():
    raise ValueError("All values in compositional data must be positive for CLR transformation.")

# Step 5: Applying CLR transformation
clr_values = clr(closure(comp_data))

# Step 6: Replacing original columns with CLR-transformed values
clr_df = pd.DataFrame(clr_values, columns=comp_data.columns, index=comp_data.index)
numeric_df.update(clr_df)

# Step 7: Combining with non-numeric columns
final_df = pd.concat([non_numeric.reset_index(drop=True), numeric_df.reset_index(drop=True)], axis=1)

# Step 8: Saving the CLR-transformed result
output_path = "C:/Users/20191/Documents/clr_transformed_Fe_to_TiO2.xlsx"
final_df.to_excel(output_path, index=False)
```

```
print(f"CLR transformation complete. File saved to:\n{output_path}")

import matplotlib.pyplot as plt
import seaborn as sns

# Step 1: Plotting histograms for CLR-transformed data
clr_df.hist(figsize=(12, 10), bins=30, edgecolor='black')
plt.suptitle('Histogram of CLR-Transformed Data', fontsize=16)
plt.tight_layout()

# Step 2: Saving the figure
plt.savefig("C:/Users/20191/Documents/clr_histograms.png", dpi=300)

# Step 3: Showing the plot
plt.show()

import matplotlib.pyplot as plt
import seaborn as sns

plt.figure(figsize=(10, 8))
sns.heatmap(clr_df.corr(), annot=True, cmap='coolwarm', fmt=".2f")
plt.title("Correlation Heatmap of CLR-Transformed Data")

# Step 1: Saving the heatmap
plt.savefig("C:/Users/20191/Documents/clr_heatmap.png", dpi=300)

# Step 2: Showing the heatmap
plt.show()

import matplotlib.pyplot as plt
from sklearn.decomposition import PCA
import numpy as np

# Step 1: Selecting only the elements from Fe to TiO2
selected_elements = clr_df.loc[:, 'Fe':'TiO2']
```

```
# Step 2: Running PCA on selected CLR-transformed data
pca = PCA(n_components=2)
pca_result = pca.fit_transform(selected_elements)

# Step 3: Getting loadings (arrows)
loadings = pca.components_.T * np.sqrt(pca.explained_variance_)

#Step 4: Scaling factor to make arrows more visible
scaling_factor = 2

# Step 5: Creating the PCA biplot
plt.figure(figsize=(10, 8))
plt.scatter(pca_result[:, 0], pca_result[:, 1], alpha=0.7)

# Step 6: Plotting the loadings as arrows with labels
for i, feature in enumerate(selected_elements.columns):
    plt.arrow(0, 0,
              loadings[i, 0] * scaling_factor,
              loadings[i, 1] * scaling_factor,
              color='red', alpha=0.7, head_width=0.02)
    plt.text(loadings[i, 0] * scaling_factor * 1.1,
             loadings[i, 1] * scaling_factor * 1.1,
             feature,
             color='black',
             ha='center',
             va='center',
             fontsize=12)  # Text for feature names

# Step 7: Setting labels and title with increased font sizes
plt.xlabel(f"PC1 ({pca.explained_variance_ratio_[0]*100:.2f}%)", fontsize=14)
plt.ylabel(f"PC2 ({pca.explained_variance_ratio_[1]*100:.2f}%)", fontsize=14)
plt.title("PCA of CLR-Transformed Data", fontsize=16)

# Step 8: Drawing reference lines and grid
plt.axhline(0, color='gray', linewidth=0.5)
plt.axvline(0, color='gray', linewidth=0.5)
```

```
plt.grid(True)
plt.tight_layout()

# Step 9: Saving the figure as an image
plt.savefig("C:/Users/20191/Documents/pca_clr_plot.png", dpi=300)

# Show the figure
plt.show()
```

Appendix D: Model Training (CNN-LSTM)

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import joblib

from sklearn.model_selection import KFold
from sklearn.preprocessing import StandardScaler
import tensorflow as tf
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Conv1D, LSTM, Dense, Dropout, Input, MaxPooling1D
from tensorflow.keras.callbacks import EarlyStopping
import tensorflow.keras.backend as K

# ============================
# Custom Loss: Masked MSE
# ============================
def masked_mse(y_true, y_pred):
    mask = K.cast(K.not_equal(y_true, 0.0), tf.float32)
    squared_error = K.square(y_true - y_pred) * mask
    return K.sum(squared_error) / (K.sum(mask) + K.epsilon())

# ============================
# Custom Metric: Masked MAE
# ============================
def masked_mae(y_true, y_pred):
    mask = K.cast(K.not_equal(y_true, 0.0), tf.float32)
    abs_error = K.abs(y_true - y_pred) * mask
    return K.sum(abs_error) / (K.sum(mask) + K.epsilon())

# ============================
# Loading & Preprocess Data
# ============================
geo_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_training/Geochemical_data_training.xlsx'
```

```
pca_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_training/PCA_scores_reflectance.xlsx'

mineral_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_training/Mineral_abundance_training.xlsx'


geo_df = pd.read_excel(geo_path)

pca_df = pd.read_excel(pca_path)

mineral_df = pd.read_excel(mineral_path)


# Standardizing column names and types

for df in [geo_df, pca_df, mineral_df]:

    df.columns = df.columns.str.strip().str.lower().str.replace(' ', '_').str.replace('-', '_')

    df['hole_id'] = df['hole_id'].astype(str).str.strip().str.lower()

    df['depth_from'] = pd.to_numeric(df['depth_from'], errors='coerce')


# Merging dataframes

merged = geo_df.merge(pca_df, on=['hole_id', 'depth_from'], how='inner')

merged = merged.merge(mineral_df, on=['hole_id', 'depth_from'], how='inner')


# ============================

# Featuring + Target Preparation

# ============================

target_cols = mineral_df.columns.difference(['hole_id', 'depth_from'])

X = merged.drop(columns=['hole_id', 'depth_from'] + list(target_cols))

X = X.select_dtypes(include=[np.number])

X = X.fillna(X.mean(numeric_only=True))


# Standardizing features

scaler = StandardScaler()

X_scaled = scaler.fit_transform(X).astype(np.float32)


# Saving scaler

out_dir = os.path.dirname(geo_path)

scaler_path = os.path.join(out_dir, 'mineral_mapping_scaler.save')

joblib.dump(scaler, scaler_path)

print(f"Scaler saved: {scaler_path}")
```

```
# Target
Y_raw = merged[target_cols]
Y_filled = Y_raw.fillna(0.0).astype(np.float32).to_numpy()

# ============================
# 5-Fold Cross-Validation
# ============================
kf = KFold(n_splits=5, shuffle=True, random_state=42)

fold_no = 1
all_val_losses = []
all_val_maes = []
all_histories = []

for train_idx, val_idx in kf.split(X_scaled):
    print(f"\n=== Fold {fold_no} ===")
    X_train, X_val = X_scaled[train_idx], X_scaled[val_idx]
    y_train, y_val = Y_filled[train_idx], Y_filled[val_idx]

    # Reshape for CNN-LSTM: (samples, sequence_length, features=1)
    X_train_hybrid = X_train.reshape(X_train.shape[0], X_train.shape[1], 1)
    X_val_hybrid = X_val.reshape(X_val.shape[0], X_val.shape[1], 1)

    # Model
    model = Sequential([
        Input(shape=(X_train.shape[1], 1)),
        Conv1D(filters=64, kernel_size=3, activation='relu'),
        MaxPooling1D(pool_size=2),
        Dropout(0.1),
        LSTM(64, return_sequences=False),
        Dropout(0.1),
        Dense(64, activation='relu'),
        Dropout(0.1),
        Dense(Y_filled.shape[1], activation='linear')
    ])
```

```
model.compile(optimizer='adam', loss=masked_mse, metrics=[masked_mae])
early_stop = EarlyStopping(monitor='val_loss', patience=15, restore_best_weights=True)

history = model.fit(
    X_train_hybrid, y_train,
    validation_data=(X_val_hybrid, y_val),
    epochs=50,
    batch_size=32,
    callbacks=[early_stop],
    verbose=1
)

val_loss, val_mae = model.evaluate(X_val_hybrid, y_val, verbose=0)
print(f"Fold {fold_no} - Val Loss: {val_loss:.4f} - Val MAE: {val_mae:.4f}")
all_val_losses.append(val_loss)
all_val_maes.append(val_mae)
all_histories.append(history.history)

# Save model
model_path = os.path.join(out_dir, f'mineral_mapping_hybrid_fold{fold_no}.keras')
model.save(model_path)
print(f"Model saved to: {model_path}")

# Save Fold Predictions
val_ids = merged.iloc[val_idx][['hole_id', 'depth_from']].reset_index(drop=True)
val_predictions = model.predict(X_val_hybrid)

pred_df = val_ids.copy()
for i, col in enumerate(target_cols):
    pred_df[col] = val_predictions[:, i]

prediction_path = os.path.join(out_dir, f'mineral_predictions_hybrid_fold{fold_no}.xlsx')
pred_df.to_excel(prediction_path, index=False)
print(f"Fold {fold_no} predictions saved to: {prediction_path}")

fold_no += 1
```

```
# ============================
# Plotting Average Loss & MAE
# ============================
def pad_history(histories, key):
    max_len = max(len(h[key]) for h in histories)
    padded = []
    for h in histories:
        values = h[key]
        padded.append(np.pad(values, (0, max_len - len(values)), constant_values=np.nan))
    return np.array(padded)

avg_train_loss = np.nanmean(pad_history(all_histories, 'loss'), axis=0)
avg_val_loss = np.nanmean(pad_history(all_histories, 'val_loss'), axis=0)
avg_train_mae = np.nanmean(pad_history(all_histories, 'masked_mae'), axis=0)
avg_val_mae = np.nanmean(pad_history(all_histories, 'val_masked_mae'), axis=0)

plt.figure(figsize=(12, 6))

plt.subplot(1, 2, 1)
plt.plot(avg_train_loss, label='Avg Train Loss')
plt.plot(avg_val_loss, label='Avg Val Loss')
plt.title('Average Loss Across Folds')
plt.xlabel('Epoch')
plt.ylabel('Loss')
plt.legend()
plt.grid(True)

plt.subplot(1, 2, 2)
plt.plot(avg_train_mae, label='Avg Train MAE')
plt.plot(avg_val_mae, label='Avg Val MAE')
plt.title('Average MAE Across Folds')
plt.xlabel('Epoch')
plt.ylabel('MAE')
plt.legend()
plt.grid(True)
```

```
summary_plot_path = os.path.join(out_dir, 'mineral_mapping_hybrid_summary_plot.png')
plt.tight_layout()
plt.savefig(summary_plot_path, dpi=300)
plt.close()
print(f"Summary plot saved to: {summary_plot_path}")

# ============================
# Summary
# ============================
print("\n=== Cross-Validation Summary ===")
print(f"Average Val Loss: {np.mean(all_val_losses):.4f}")
print(f"Average Val MAE:  {np.mean(all_val_maes):.4f}")
```

Appendix E: Model Testing (CNN-LSTM)

```
import os

import pandas as pd

import numpy as np

import joblib

import tensorflow as tf

from tensorflow.keras.models import load_model

from sklearn.metrics import mean_absolute_error, mean_squared_error, r2_score

import math

import matplotlib.pyplot as plt

import seaborn as sns


# ============================

# Paths

# ============================

geo_test_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/Geochemical_data_testing.xlsx'

pca_test_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/PCA_scores_reflectance_testing.xlsx'

scaler_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_training/CNN-LSTM/mineral_mapping_scaler.save'

model_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_training/CNN-LSTM/mineral_mapping_hybrid_fold5.keras'

true_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/Mineral_abundance_testing.xlsx'


predictions_output_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/Mineral_abundance_predictions_CNN_LSTM_uncertainty.xlsx'

metrics_output_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/mineral_mae_rmse_uncertainty_results_CNN_LSTM.xlsx'

scatter_plot_output_path = r'C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/true_vs_predicted_scatter_plot.png'


# ============================

# Loading & preprocessing test data

# ============================
```

```
geo_df = pd.read_excel(geo_test_path)
pca_df = pd.read_excel(pca_test_path)

for df in [geo_df, pca_df]:
    df.columns = df.columns.str.strip().str.lower().str.replace(' ', '_').str.replace('-', '_')
    df['hole_id'] = df['hole_id'].astype(str).str.strip().str.lower()
    df['depth_from'] = pd.to_numeric(df['depth_from'], errors='coerce')

merged = geo_df.merge(pca_df, on=['hole_id', 'depth_from'], how='inner')
id_df = merged[['hole_id', 'depth_from']].copy()

X_test = merged.drop(columns=['hole_id', 'depth_from'])
X_test = X_test.select_dtypes(include=[np.number])
X_test = X_test.fillna(X_test.mean(numeric_only=True))

scaler = joblib.load(scaler_path)
X_scaled = scaler.transform(X_test).astype(np.float32)
X_cnn_lstm = X_scaled.reshape(X_scaled.shape[0], X_scaled.shape[1], 1)

# ============================
# Custom loss and metric
# ============================
@tf.keras.utils.register_keras_serializable()
def masked_mse(y_true, y_pred):
    mask = tf.cast(tf.not_equal(y_true, 0.0), tf.float32)
    squared_error = tf.square(y_true - y_pred) * mask
    return tf.reduce_sum(squared_error) / (tf.reduce_sum(mask) + tf.keras.backend.epsilon())

@tf.keras.utils.register_keras_serializable()
def masked_mae(y_true, y_pred):
    mask = tf.cast(tf.not_equal(y_true, 0.0), tf.float32)
    abs_error = tf.abs(y_true - y_pred) * mask
    return tf.reduce_sum(abs_error) / (tf.reduce_sum(mask) + tf.keras.backend.epsilon())

# ============================
# Loading CNN-LSTM model
```

```
# ============================
model = load_model(model_path, custom_objects={'masked_mse': masked_mse, 'masked_mae': masked_mae})

# ============================
# MC Dropout Prediction helper
# ============================
@tf.function
def predict_with_mc_dropout(x):
    return model(x, training=True)  # keep dropout active during inference

def mc_dropout_predict(model, X, n_iter=200):
    preds = []
    for _ in range(n_iter):
        preds.append(predict_with_mc_dropout(X).numpy())
    preds = np.array(preds)  # shape: (n_iter, n_samples, n_outputs)
    mean_pred = preds.mean(axis=0)
    epistemic_uncertainty = preds.var(axis=0)
    return mean_pred, epistemic_uncertainty

# ============================
# Running MC dropout prediction
# ============================
mean_preds, epistemic_uncertainty = mc_dropout_predict(model, X_cnn_lstm, n_iter=200)

# ============================
# Loading true mineral targets for column names
# ============================
true_df = pd.read_excel(true_path)
true_df.columns = true_df.columns.str.strip().str.lower().str.replace(' ', '_').str.replace('-', '_')
target_cols = true_df.columns.difference(['hole_id', 'depth_from']).tolist()

# ============================
# Saving predictions + uncertainty
# ============================
pred_df = pd.DataFrame(mean_preds, columns=target_cols)
uncertainty_df = pd.DataFrame(epistemic_uncertainty, columns=[f"{col}_uncertainty" for col in target_cols])
```

```
pred_df = pd.concat([id_df.reset_index(drop=True), pred_df, uncertainty_df], axis=1)
pred_df.to_excel(predictions_output_path, index=False)
print(f" Predictions + uncertainty saved to: {predictions_output_path}")

# ============================
# Cleaning true df for merge
# ============================
def clean_df(df):
    df.columns = df.columns.str.strip().str.lower().str.replace(' ', '_').str.replace('-', '_')
    df['hole_id'] = df['hole_id'].astype(str).str.strip().str.lower()
    df['depth_from'] = pd.to_numeric(df['depth_from'], errors='coerce').round(3)
    return df

true_df = clean_df(true_df)

# ============================
# Merging predictions with true values
# ============================
merged_df = pd.merge(pred_df, true_df, on=['hole_id', 'depth_from'], how='inner', suffixes=('_pred', '_true'))
if merged_df.empty:
    raise ValueError(" ✕ Merged dataframe is empty. Check hole_id/depth_from alignment.")

# ============================
# Calculating metrics per mineral and overall, including R2
# ============================
results = []
overall_true = []
overall_pred = []
overall_uncertainty_vals = []

for mineral in target_cols:
    y_true = merged_df[f"{mineral}_true"].values
    y_pred = merged_df[f"{mineral}_pred"].values
    uncertainty_vals = merged_df[f"{mineral}_uncertainty"].values

    mask = ~np.isnan(y_true) & ~np.isnan(y_pred)
```

```
    y_true_clean = y_true[mask]
    y_pred_clean = y_pred[mask]
    uncertainty_clean = uncertainty_vals[mask]

    mae = mean_absolute_error(y_true_clean, y_pred_clean)
    rmse = math.sqrt(mean_squared_error(y_true_clean, y_pred_clean))
    r2 = r2_score(y_true_clean, y_pred_clean)
    mean_uncertainty_val = np.mean(uncertainty_clean)

    results.append({
        'Mineral': mineral,
        'MAE': mae,
        'RMSE': rmse,
        'R2': r2,
        'Mean_Epistemic_Uncertainty': mean_uncertainty_val
    })

    overall_true.extend(y_true_clean)
    overall_pred.extend(y_pred_clean)
    overall_uncertainty_vals.extend(uncertainty_clean)

overall_mae = mean_absolute_error(overall_true, overall_pred)
overall_rmse = math.sqrt(mean_squared_error(overall_true, overall_pred))
overall_r2 = r2_score(overall_true, overall_pred)
overall_mean_uncertainty = np.mean(overall_uncertainty_vals)

results.append({
    'Mineral': 'Overall',
    'MAE': overall_mae,
    'RMSE': overall_rmse,
    'R2': overall_r2,
    'Mean_Epistemic_Uncertainty': overall_mean_uncertainty
})

results_df = pd.DataFrame(results)
results_df.to_excel(metrics_output_path, index=False)
```

```
print(f" MAE, RMSE, R2, and epistemic uncertainty results saved to: {metrics_output_path}")
print(results_df)

# ============================
# Scatter plot: True vs Predicted mineral abundances (All minerals on one figure)
# ============================

sns.set(style="whitegrid")
plt.figure(figsize=(12, 9))

# Define your custom colors here
custom_colors = {
    'fe_ox_ai': 'blue',
    'kaolin_abundance': 'skyblue',
    'wmalsmai': 'orange',
    'carbai3pfit': 'grey'
}

for mineral in target_cols:
    color = custom_colors.get(mineral, 'black')  # fallback black if mineral not in dict
    plt.scatter(
        merged_df[f"{mineral}_true"],
        merged_df[f"{mineral}_pred"],
        alpha=0.8,
        s=60,
        color=color,
        label=mineral,
        edgecolor='black',
        linewidth=0.7,
    )

# Plotting perfect prediction line
min_val = min(merged_df[[f"{mineral}_true" for mineral in target_cols]].min().min(),
              merged_df[[f"{mineral}_pred" for mineral in target_cols]].min().min())
max_val = max(merged_df[[f"{mineral}_true" for mineral in target_cols]].max().max(),
              merged_df[[f"{mineral}_pred" for mineral in target_cols]].max().max())
```

```
plt.plot([min_val, max_val], [min_val, max_val], 'r--', label='Perfect Prediction')

# Making all text bigger and readable
plt.title("True vs Predicted Mineral Abundance (CNN-LSTM)", fontsize=18, fontweight='bold')
plt.xlabel("True Abundance", fontsize=16)
plt.ylabel("Predicted Abundance", fontsize=16)
plt.xticks(fontsize=14)
plt.yticks(fontsize=14)
legend = plt.legend(title="Minerals", fontsize=14, title_fontsize=16)
plt.setp(legend.get_title(), fontweight='bold')

plt.tight_layout()

plt.savefig(scatter_plot_output_path, dpi=300)
print(f" Scatter plot saved to: {scatter_plot_output_path}")
plt.show()
```

Appendix F: Mineral Abundance and Mineral Association Maps (CNN-LSTM)

```
import os
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import griddata
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from matplotlib.colors import ListedColormap

# ======== File path ========
file_path = r"C:/Users/20191/Downloads/Rocklea_Dome_C3DMM-D6mjqY_--/data/drill hole data/RKD/Keystone_D/Methodology_Matimbi_DK/Model_testing/CNN-LSTM/CNN_LSTm_predictions.xlsx"

# ======== Reading data ========
df = pd.read_excel(file_path)

# ======== Identifying mineral columns ========
mineral_cols = [col for col in df.columns if col not in ['hole_id', 'depth_from', 'easting', 'northing']]

# ======== Mapping holes to numbers ========
holes = df['hole_id'].unique()
n_holes = len(holes)
hole_map = dict(zip(holes, np.arange(n_holes)))

# ======== Preparing data for clustering ========
samples = df[mineral_cols].values
samples = np.nan_to_num(samples)  # replace NaNs with 0 for clustering
scaler = StandardScaler()
samples_scaled = scaler.fit_transform(samples)

# ======== KMeans clustering ========
n_clusters = 4
kmeans = KMeans(n_clusters=n_clusters, random_state=42, n_init=10)
clusters = kmeans.fit_predict(samples_scaled)
```

```
df['cluster'] = clusters + 1  # start clusters at 1

# ======== Creating grid for interpolation ========
depths = np.sort(df['depth_from'].unique())
grid_depths, grid_holes = np.meshgrid(depths, np.arange(n_holes))

# ======== Plotting continuous mineral maps ========
n_cols = 2
n_rows = int(np.ceil(len(mineral_cols) / n_cols))
fig, axes = plt.subplots(n_rows, n_cols, figsize=(14, 5 * n_rows), constrained_layout=True)
axes = axes.flatten()

for i, mineral in enumerate(mineral_cols):
    points = np.array([[row['depth_from'], hole_map[row['hole_id']]] for _, row in df.iterrows()])
    values = np.nan_to_num(df[mineral].values)
    grid_values = griddata(points, values, (grid_depths, grid_holes), method='cubic')

    # Flipping vertically so the "upper side" faces downward
    flipped_values = np.flipud(grid_values.T)

    im = axes[i].imshow(
        flipped_values,
        aspect='auto',
        origin='upper',  # keep depth increasing downwards
        cmap='viridis',
        extent=[-0.5, n_holes - 0.5, depths.min(), depths.max()]
    )

    axes[i].set_title(f"{mineral} Abundance", fontsize=14, fontweight='bold')

    # Hole labels at top
    axes[i].xaxis.tick_top()
    axes[i].xaxis.set_label_position('top')
    axes[i].set_xlabel("Hole ID", labelpad=10)

    axes[i].set_ylabel("Depth from (m)")
```

```
    axes[i].set_xticks(np.arange(n_holes))
    axes[i].set_xticklabels(holes, rotation=90)

    # Depth increases downward
    axes[i].invert_yaxis()

    fig.colorbar(im, ax=axes[i], label="Abundance")

# Hide unused subplots
for j in range(i + 1, len(axes)):
    fig.delaxes(axes[j])

# ======== Save mineral abundance map ========
mineral_output_path = os.path.join(os.path.dirname(file_path),
        "all_minerals_continuous_map_cnn_lstm_flipped.png")
plt.savefig(mineral_output_path, dpi=300, bbox_inches='tight')
plt.show()
print(f" Continuous mineral abundance map saved to: {mineral_output_path}")

# ======== Plotting continuous cluster (mineral association) map ========
cluster_colors = ['red', 'blue', 'pink', 'skyblue']
cluster_cmap = ListedColormap(cluster_colors)

points = np.array([[row['depth_from'], hole_map[row['hole_id']]] for _, row in df.iterrows()])
values = df['cluster'].values
grid_clusters = griddata(points, values, (grid_depths, grid_holes), method='nearest')

flipped_clusters = np.flipud(grid_clusters.T)

plt.figure(figsize=(12, 6))
im = plt.imshow(
    flipped_clusters,
    aspect='auto',
    origin='upper',
    cmap=cluster_cmap,
    extent=[-0.5, n_holes - 0.5, depths.min(), depths.max()]
```

```
)

# Hole labels at top
ax = plt.gca()
ax.xaxis.tick_top()
ax.xaxis.set_label_position('top')
plt.xticks(np.arange(n_holes), holes, rotation=90)

plt.xlabel('Hole ID', labelpad=10)
plt.ylabel('Depth from (m)')
plt.title(f'Mineral Association Clusters (K={n_clusters})')

# Depth increases downward
plt.gca().invert_yaxis()

plt.colorbar(im, ticks=np.arange(1, n_clusters+1), label='Cluster')

# ======== Saving cluster map ========
cluster_output_path = os.path.join(os.path.dirname(file_path),
    "mineral_association_clusters_cnn_lstm_flipped.png")
plt.savefig(cluster_output_path, dpi=300, bbox_inches='tight')
plt.show()
print(f" Mineral association (cluster) map saved to: {cluster_output_path}")
```