Study on the Development of an AI-Based Mining Methods Recommendation System

(AI による採掘手法レコメンデーションシステム開発に関する研究)

By

Elsa Pansilvania Andre MANJATE

A dissertation submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy in Engineering

Division of Sustainable Resources Engineering
Graduate School of Engineering
Hokkaido University
JAPAN

March 2023

DECLARATION

I, Elsa Pansilvania Andre Manjate (Elsa Pansilvânia André Manjate), declare that this dissertation is an original report of my study and has been written by me under the supervision of Professor Youhei Kawamura of Hokkaido University. I carried out the entire work, and due references have been provided on all supporting literature and resources. I ensure that this dissertation has not been submitted, in whole or part, in any previous application for any degree.

______________________________ ______________________________

Candidate´s signature Date

Certified by

Professor Youhei Kawamura (Supervisor)

______________________________ ______________________________

Supervisor´s signature Date

ACKNOWLEDGEMENT

Pursuing this dream was only possible thanks to the Japan International Cooperation Agency (JICA) for the KIZUNA program scholarship from my Master's degree to PhD (from September 2017 to March 2023). I am grateful to JICA for sponsoring my studies and my professional growth throughout the program by providing special training programs, assisting with internships and field trips, and supporting my life in Japan.

I am forever grateful to Professor Youhei KAWAMURA for his endless support from the beginning, his supervision, mentorship, and guidance throughout my academic life and beyond my academic life.

I extend my gratitude to Professors Hisatoshi TORIYA and Tsuyoshi ADACHI from Akita University and Professor Yoko OHTOMO from Hokkaido University for their support and guidance throughout my academic life. I would not forget to thank Akita University and Hokkaido University in general for their active support during my academic life in Japan.

To Professor Bernardo Miguel BENE, the General Director of the Instituto Superior Politécnico de Tete in Mozambique, who has motivated me and given me the opportunity to grow professionally and pursue further academic degrees. I am deeply grateful for his continuous support in my professional and academic careers. I would not forget to thank my colleagues and everyone at Instituto Superior Politécnico de Tete, for their support and active collaboration, especially during the field trips in Mozambique.

I thank Mitsui & Co., Ltd (during my Masters in September 2018) and Mitsubishi Materials Corporation at Higashitani Mine (during my PhD in December 2021) in Japan for providing the internship programs which were essential to fulfilling my academic degree in Japan.

I also thank Mitsui & Co. Europe Plc (Maputo Office), and Hitachi Construction Machinery Moçambique Ltd. (Tete Office), for their cooperation and kind support during our field trips in Mozambique. My gratitude also goes to Jindal Africa, JSPL Mozambique Minerais Lda, the coal mine in Tete (especially Mr N Vishwanadham), Exploratotor Ltd, the gold mine in Manica (especially to D'clay Mario Juta and Ildo Tsoca) for their cooperation, guidance and kind support during field trips in Mozambique.

I am grateful to my colleagues in the laboratory at Akita University and Hokkaido University for being such good friends and family: Brian, Kei, Yoshio, Jo, Jaewon, Zed, Ume, Taeyoo, Rempei, Rei, Takeuchi, Natsuo, Jun, Ellisha, Yuimi, Narihiro, Senjoba, Bashir, Tatsuki, Kursat, Yewuhalashet, Yuhei, Khan, Dorj, Taiga, Daniyar, Tomoya, Takaya, Shinchiro, Yosuke, Hiromassa, Yutaro, Yuta, Taichi, Armandine, Adrian, Yuna, Kaito, Takahiro. I deeply appreciate their love, encouragement, support in my studies and the great moments we shared.

To my close friends and family in Japan, Dércio Levi, Manuel, David, Dorcas, Euclésia, Watanabe San, Fyfy, Kay, Luisiana, Luisa, Márcia (Kanuca), Traifose, Renata, Solange, Mércia, Marina, Sheila, Inora, Amadeus, Joyce, and Sambo. I am thankful for their love and support in and beyond my student life, especially during tough times.

Special gratitude goes to my mother, Rosta Eugénio Machava, for her unconditional love, support, and encouragement throughout this journey. I did not forget to thank my father, André Jaime Manjate, who has also been supportive. To my girlfriend Maria, my sister Arlinda, my brothers Eugénio Dálvio, Epifánio, Eugénio (Dzacuta), and Gérson, my cousins Lidia (Lilly) and Elsa (Siminha), who have been my cheerleaders and unendingly supportive. I am very grateful for having you in my life, especially in this chapter and for your unconditional love.

I am deeply grateful to my entire family, friends and loved ones whom I could not name individually. You have been essential and very supportive. Much Love!

TO GOD, I am grateful for everything!

ABSTRACT

The success of a mining project highly relies on the feasibility of the adopted mining method(s) to recover mineral resources safely and efficiently from the earth. Therefore, mining methods selection (MMS) is one of the most critical and complex decision-making tasks in mine planning. The complexity of the MMS task is associated with the need to consider several influencing factors in the evaluations, including the orebody deposit geometry, geology and geotechnical properties, and economic, technological, and environmental factors. As such, the MMS process aims to select the most feasible method(s) to maximize profits and the recovery of mineral resources while minimizing mining costs and environmental impacts. MMS has been studied for many years, culminating in the development of different MMS systems. The first systems, termed qualitative systems, were introduced in the 1970s and 1980s, which were essentially flowcharts that served as guidelines for selecting the most suitable mining methods. The need for improvement and numerical methods led to the introduction of quantitative systems (the Nicholas approach and the UBC-MMS system) in the 1980s and 1990s, which are very useful and continue to be one of the most used for MMS. However, in quantitative systems, the relative importance of the influencing factors is not considered in the evaluations and selection process, plus the systems may offer obsolete solutions to today's requirements. Since the 2000s, multi-criteria decision-making (MCDM) techniques have been applied for MMS to overcome the shortcomings of the quantitative systems, whereby the relative importance of the influencing factors is considered in the evaluations. In MCDM-based systems, decision-makers are usually assigned to determine the weights or the relative importance of the factors subjectively (their opinion or judgment). However, using subjective (customized) judgement may introduce a certain level of bias, which inherently affects the eligibility and accuracy of the evaluations. Technological advancement, innovation and data availability have led to the growth of artificial intelligence (AI) and machine learning (ML) and their application in different fields of science and engineering. Recently, few studies have investigated the application of artificial neural network (ANN) algorithms in MMS, thus proving their effectiveness in solving the complexity of the MMS process.

In light of improving by addressing the challenges/gaps in the previous systems and extending the application of AI (and ML) in MMS, this study introduces the recommendation system approach in the MMS discipline. Recommendation systems are part of AI systems that helps users deal with information overload by filtering relevant information and making personalized recommendations based on users' historical information, thus improving users' decision-making ability. This study investigates the possibility of incorporating AI to explore available mining projects database for developing a system that can aid during the project development decision-making process, i.e., in the mine planning process. Therefore, the general aim of the study is to develop an AI-based mining methods recommendation system (AI-MMRS) based on mining projects' historical data. The study not only introduces the recommendation systems approach in MMS, but the proposed system integrates different strategies attempting to address the challenges/gaps in the previous MMS systems, which is explored by splitting the study into five main Chapters.

The study's first aim was to address the complexity of MMS associated with the relative importance of the influencing factors by determining the relative importance of the factors and identifying the most relevant factors without the direct involvement of decision-makers. As such, the MCDM Entropy method was applied to assess the relative importance of twenty factors by calculating their objective weights. The results showed that the ore strength, host-rock strength, orebody thickness, shape, dip, ore uniformity, mining costs and dilution have the higher objective weights (or higher relative importance), thus, identified as the most relevant factors in MMS. Using the Entropy method to determine objective weights of the factors eliminates the potential bias brought about by direct subjective (customized) decision-making, thus providing non-customized and generalized results. These results were the foundation for subsequent steps in developing the proposed AI-MMRS.

Because historical data availability is the backbone of developing AI systems using ML, the study's second aim was to create the input datasets used to evaluate the ML models for the AI-MMRS. The study's database is mainly based on mining projects' historical data (or technical reports) collected from an open-source database named *SEDAR*. In this study, the "data sparsity problem" was faced as one of the limitations caused by the lack of information about the required influencing factors in some projects' technical reports. The “data sparsity problem” forced the reduction of the quality of the input datasets (resulting in small and imbalanced datasets), which reflected in the quality of the ML models for the AI-MMRS. Two input datasets were created by filtering (and extracting) comprising the five relevant influencing factors (ore strength, rock strength, orebody thickness, shape, and dip) and five to seven underground mining methods. These datasets describe historical information on thirty to thirty-three mining projects regarding the orebody characteristics and the underground mining methods considered/selected to recover the orebody deposits. In the subsequent steps, these datasets were used to evaluate the models for developing the AI-MMRS.

The third aim was to develop a methodology to incorporate one of the well-known recommendation systems approach in mining methods selection (MMS): the memory-based collaborative filtering (CF) approach. Therefore, investigating the applicability of the memory-based CF approach in MMS through the k-nearest neighbours (KNN) with cosine similarity algorithm (KNN-cosine similarity algorithm). The essential step in the methodology involved creating an appropriate input dataset to evaluate the proposed model, which was done with the aid of the UBC-MMS system. The training dataset comprises thirty-three projects, the five input variables (ore strength, rock strength, orebody thickness, shape, and dip) and seven underground mining methods (block caving, cut and fill, room and pillar, longwall, shrinkage, sublevel caving and sublevel stoping). The proposed model was evaluated to predict and recommend the top-3 most relevant underground mining methods for a target project. The results showed that the memory-based CF approach is effective for MMS, given that the proposed model could predict and recommend the top-3 relevant underground mining methods with an accuracy ranging from 81.8% to 87.9%.

Acknowledging the “data sparsity problem” as one of the study's limitations, which forced the reduction of the quality of the input datasets, the fourth aim was to assess the capability of the nonnegative matrix factorization (NMF) algorithm to predict possible missing values from the sparse input dataset. The NMF algorithm was introduced to address the “data sparsity problem” to enable data augmentation to improve the quality of the input datasets. Using the input dataset comprising thirty projects, five input variables and five underground mining methods, the NMF model was evaluated to predict missing values in the sparse dataset. The results showed the NMF model’s effectiveness in predicting missing values from a sparse dataset with a moderate accuracy ranging from 60% to 70%.

The need for better models to address the limitations of the memory-based CF approach associated with the dependency on the UBC-MMS system led to the introduction of classification machine learning (ML) algorithms in MMS. Therefore, the fifth aim was to investigate the applicability of ML classification algorithms to predict (classify) underground mining methods. This aim involved training and evaluating different models to classify seven underground mining (block caving, cut and fill, room and pillar, longwall, shrinkage, sublevel caving and sublevel stoping) based on five input variables (ore strength, rock strength, orebody thickness, shape, and dip). The results demonstrated that the models could effectively classify the seven underground mining methods, with the best models (ANN, KNN and support vector machines) performing with moderate accuracy ranging from 60% to 70%.

In conclusion, the study expands the application of AI in MMS by introducing the recommendation system approach in the MMS discipline. Therefore, proposing the AI-MMRS by implementing the CF approach to recommend the most appropriate mining methods by learning from previous mining projects' procedures. The introduction of the recommendation systems approach in MMS possesses benefits associated with efficiency and the potential to learn from past experiences (mining projects' historical data). The results showed that the evaluated models effectively predicted (and classified) underground mining methods performing with moderate accuracy, which is considered realistic given the limitation associated with the limited size of the input datasets. Despite the limitations, the findings from this study demonstrated that the proposed AI-MMRS can be viable and practical for MMS. Continuous data collection and model optimization are required to improve the recommendations, thus building a robust system.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI: artificial intelligence

ML: machine learning

CF: collaborative filtering

MMS: mining methods selection

AI-MMRS: AI-based mining methods recommendation system

ANN: artificial neural network

NMF: nonnegative matrix factorization

WNMF: weighted nonnegative matrix factorization

KNN: k-nearest neighbours

SVM: support vector machines

CTGAN: conditional tabular generative adversarial network

DMS: document management software

UBC-MMS tool: UBC mining methods selection tool

MCDM: multi-criteria decision-making

AHP: analytical hierarchy process

FAHP: fuzzy analytical hierarchy process

TOPSIS: technique for order preference by similarity to ideal solution

PROMETHEE: preference ranking organisation method for enrichment evaluation

RMR: rock mass rating

RSS: rock substance strength

m: metres

^{o}: degrees

%: percentage

etc.: etcetera

GAR: global accuracy rate

MRR: mean reciprocal rank

1 Introduction

1.1 General Introduction

The success of a mining project highly relies on the feasibility of the adopted mining method (s) to recover mineral resources from the earth safely and efficiently. Therefore, mining methods selection (MMS) is one of the most critical and complex decision-making tasks in mine planning. The MMS process attempts to deduce the most feasible or appropriate method (s) to maximise profits and recover mineral resources while minimising mining costs and environmental impacts. Surface and underground mining methods are the most common mining types. This process is considered complex and somewhat problematic because the selection of the most feasible mining method (s) requires the consideration of several interconnected factors, including historical, social, and cultural factors, physical characteristics of the orebody, geotechnical properties, geological and geographical conditions, as well as technological, economic, and environmental factors [1], [2]. MMS has been studied for many years, culminating in the development of different MMS systems. The first MMS systems, termed qualitative systems, were introduced in the 1970s and 1980s, which were essentially flowcharts that served as guidelines for classifying and evaluating different surface and underground mining methods in terms of several factors [1], [3]. The need for improvement and numerical methods led to the introduction of quantitative systems [4], [5] in the 1980s and 1990s, which continue to be one of the most used systems for MMS. Since the 2000s, attention has shifted to applying multi-criteria decision-making (MCDM) techniques in the MMS field to overcome the shortcomings of quantitative systems. Different MCDM techniques [6]–[13] have been employed for selecting the most feasible or appropriate mining methods for different case studies. Technological advancement, innovation and data availability have led to the growth of artificial intelligence (AI) and its application in different fields of science and engineering. Recently, few studies have investigated the application of artificial neural network algorithms in MMS [14]–[17], thus proving their effectiveness in solving the complexity of the MMS process.

In light of improving by addressing the challenges/gaps in the previous MMS systems and extending the application of AI in MMS, this study introduces the application of recommendation systems [18], [19] technologies in the MMS discipline. Recommendation systems are part of AI systems aimed at helping users deal with information overload by filtering information and making personalized recommendations (of items that users might like), thus improving users' decision-making ability. This study investigates the possibility of incorporating artificial intelligence (AI) to explore available mining projects database for developing a system that can be used as a tool to aid in decision-making when planning a mining project (i.e., the mine planning process). As such, the general aim of the study is to develop an AI-based mining methods recommendation system (AI-MMRS) focusing on underground MMS. As such, this study investigates the applicability of collaborative filtering [20], [21] recommendation systems approach to develop a system that recommends the most appropriate underground mining methods by learning from previous mining projects' procedures. The introduction of the recommendation systems approach in MMS possesses benefits associated with efficiency and the potential to learn from past experiences (mining projects' historical data) [22].

1.2 Literature Review

1.2.1 Mining methods selection

During the mine planning and design processes, selecting the best or combinations of multiple mining methods is the most critical and complex decision-making task. Moreover, adopting a particular mining method can be an irreversible decision owing to the high costs involved in changing or replacing the mining method during mining production [8]. Therefore, the mining methods selection (MMS) task requires the engagement of experienced mining engineers. Given the complexity of an orebody deposit's physical characteristics (orebody geometry) and geological conditions, extracting the entire orebody using a single mining method is almost impossible [8]. Surface and underground are the most common mining types. Surface mining methods are usually applied to recover deposits near the earth's surface with a low stripping ratio: when the removal of the overburdened material (soil and rock covering the mineral deposit) is economically viable. Surface mining methods are usually cheaper and more large-scale than underground mining. The main types are open pit mining, open cast (strip), quarrying, and auger or highwall mining [1].

On the other hand, underground methods are usually applied in the recovery of orebody deposits with extreme depth and stripping ratio to apply surface methods. They are commonly classified into three classes (i.e., unsupported, supported and caving) based on the extent of support required. Unsupported are those methods that are naturally supported (or self-supported) by the surrounding natural rock (or host-rock) with no or less artificial support system required [1]. Room and pillar [23], shrinkage [24], and sublevel stoping [25] are the three main unsupported mining methods. Supported mining methods require a magnitude of artificial support systems to maintain the stability of openings and prevent subsidence of the surface material [1]. The most common unsupported methods are cut and fill [26], stull stoping and square set stoping [1]. Caving is when the ore, host rock or both cave naturally and/or are induced in a controlled manner after the materials are recovered/extracted [1]. The main caving methods are longwall [27], sublevel caving [28], and block caving [29]. The selection of underground mining methods is usually the most challenging task due to the complexity associated with orebody deposit characteristics. Therefore, the selection of underground methods highly relies on the orebody geometry, geology, and geotechnical properties, which then define the degree of ground support necessary to ensure safety and productivity [30]. The factors that highly influence the selection of both surface and underground mining methods are categorised as follows [1], [2]:

Physical characteristics of the orebody deposit (orebody geometry): this includes orebody size (height, width, and thickness), orebody shape, orebody dip, and depth of the orebody below the surface. These factors are considered critical in determining surface and underground methods as they affect the entire mine design and production.

Geotechnical properties, geological and hydrologic conditions: referring to the rock material properties (strength, deformation, and weathering characteristics), grade distribution/ore uniformity, mineralogy, and petrology. These factors include the orebody and host rock's mechanical and structural geological compositions. They play a significant role in selecting different classes (i.e., unsupported, supported, and caving methods) of underground mining methods, including the ground support system selection.

Economic factors: including the comparative capital and mining costs of suitable methods, reserves (tonnage and grade), mine life, production rate, and productivity. These factors play an essential role in the final decision-making process of MMS; they aid in determining the feasibility of the methods based on financial and economic analyses.

Technological factors: include recovery, selectivity, dilution, the flexibility of the method to changing conditions, mechanisation or automatization, and labour intensity. These factors are mostly related to the effects of mining methods on subsequent operations, such as processing requirements, treatment, and smelting.

Environmental considerations: include subsidence, stability of openings, health, and safety. These factors are interconnected to social, political, historical, and geographical factors; they assess the rejection or acceptance of said methods given the geographic location.

MMS has been a subject of study for many years [31], culminating with the development of different MMS systems described below in section 1.2.2.

1.2.2 Mining methods selection systems

The mining methods selection (MMS) systems developed over the years are here categorized into qualitative, quantitative, multi-criteria decision-making (MCDM)-based and machine learning-based systems. Table 1.1 presents the summary of the MMS systems approaches, including this study's proposed system. The MMS systems are described as follows:

- **Qualitative systems**

Various researchers, including Boscov and Wright in 1973, Morrison in 1976, Laubscher in 1981, and Hartman in 1987, proposed the first qualitative MMS systems [1], [3]. These systems were flowcharts or qualitative classification schemes providing guidelines to select the surface and underground mining methods based on the orebody geometry (size, dip, and shape) and geotechnical properties (ore and host rock strength). The systems proposed by Boscov and Wright, Morrison and Laubscher can be applied to underground mining methods but differ in the category of factors considered in each system. Boscov and Wright proposed a system based on the orebody geometry and geotechnical properties (thickness, orebody dip, and strengths of the ore and host rock). The system suggested by Morrison is based on the orebody width, underground mine support types, and strain-energy accumulation. Laubscher proposed a system based on geotechnical parameters (rock-mass classification) aimed at mass underground mining methods [3].

The system proposed by Hartman is relatively similar to that proposed by Boscov and Wright, which is based on the orebody geometry and the mechanical characteristics of the ore zone and rock (depth, shape, dip, size, and ore and rock strength) but targets both surface and underground methods [1].

- **Quantitative systems**

In 1981, Nichola's proposed the first quantitative MMS system, the Nichola approach [5]. The Nicholas approach is based on orebody geometry (shape, thickness, plunge/dip, and depth), grade distribution, and geotechnical properties (rock mechanics characteristics of the ore and host rock) to select the most suitable mining methods (open-pit and underground methods). In this system, numerical ranks are assigned to all factors determining how certain factors make a particular mining method less or more attractive. After that, the ranks of the factors for each mining method are summed up. The mining methods with the highest ranks are selected as the most suitable methods, aiding in the evaluation of economic viability. Later, in 1995 the University of British Columbia (UBC) [4] developed the UBC-MMS tool, a modified version of the Nicholas approach. UBC modified the Nicholas approach by introducing some mechanical properties and new values in the ranks. The UBC-MMS tool emphasizes underground stoping methods and best represents Canadian mine design practices. The Quantitative systems are handy; the UBC-MMS tool is still the most commonly used in MMS practice and as a base for scientific studies (research). However, in quantitative systems, the relative importance of the factors is not considered, implying that all factors have the same degree of importance in MMS.

- **MCDM-based systems**

Currently, the trend in the MMS discipline involves the application of multi-criteria decision-making (MCDM) techniques. These MCDM-based systems were introduced in an attempt to overcome the shortcomings of the quantitative systems and proved to be effective. As several factors must be considered in MMS, the formulation of definite criteria (or factors) for selecting methods that can simultaneously satisfy all conditions of the mining procedure becomes complicated [9]. Therefore, researchers developed MMS methodologies by applying MCDM techniques, wherein the relative importance of the factors is considered. The relative importance of the factors is usually determined subjectively based on the opinion or judgment of decision-makers (or mining engineers). Techniques such as fuzzy MCDM [7], analytical hierarchy process (AHP) [9], [10], the technique for order preference by similarity to ideal solution (TOPSIS) [12], [13], fuzzy analytical hierarchy process (FAHP) [8], [12], and preference ranking organisation method for enrichment evaluation (PROMETHEE) [11], among others, were applied in different scenarios to select the most suitable mining methods for different case studies. Bitarafan and Ataei [7] applied fuzzy decision-making tools (fuzzy dominance and fuzzy multiple attribute decision-making methods) to select the best mining method for anomaly No. 3 of the Gol-Gohar iron mine, where the weights of criteria (i.e., influencing factors) and alternatives (i.e., mining methods) are determined in a fuzzy environment (subjectively); block caving was the most suitable mining method. Ataei et al. [9] explored applying the AHP technique to develop a suitable mining method for the Golbini No. 8 deposit. Their technique was applied to determine criteria weights subjectively and the best alternative (conventional cut and fill). Therefore, the AHP was a unique model that could identify multiple criteria, minimal data requirement, and minimal time consumption. Namin et al. [13] discussed the application of a decision-making tool based on the fuzzy TOPSIS to develop the MMS tool for the Gol-e-Gohar anomaly No. 3 and Chahar Gonbad deposit. In this case, the weights of the criteria over the alternatives are determined by decision-makers to create a fuzzy decision matrix; and open pit mining method was identified as the best for the deposits, and the systematic evaluation of fuzzy TOPSIS of MMS was determined to reduce the risk of a poor choice. Alpay and Yavuz [10] developed a tool based on AHP and Yager's techniques to develop a computer program to analyse underground MMS problems for the Eskisehir–Karaburun chromite ore. The computer program could also enable decision-makers to perform sensitivity analyses after selecting the best method to observe the rate proposed method according to criteria weights. Azadeh et al. [8] developed a modified version of Nicholas' approach by using a FAHP to select the most feasible mining method for the anomaly of the Choghart iron mine. In their approach, FAHP was applied to determine and modify criteria weights according to Nicholas' approach and thus determine the most suitable method considering these criteria weights. Bogdanovic et al. [11] implemented an integrated approach that employed the AHP and PROMETHEE to select the most suitable mining method for the Coka Marin underground mine. In their approach, AHP was used to assign criteria weights, while PROMETHEE was used to complete the ranking of the alternatives; sublevel caving was identified as the most suitable method. Shariati et al. [12] developed an integrated model based on FAHP and TOPSIS to select the optimum mining methods for the Angouran Mine; criteria weights were determined based on FAHP, and the TOPSIS was applied to analyse the feasible alternatives, and the alternative with the highest score was selected followed by sensitivity analyses to determine the influence of criteria weights. The most significant advantage of MCDM-based systems over quantitative systems is the consideration of the relative importance of the factors during the selection process, where the decision-makers get to decide which criteria and/or alternatives are the most relevant for the projects' evaluation. However, determining the relative importance of the factors subjectively, in essence, introduces a certain level of bias, which inherently affects the eligibility and accuracy of the assessments. Furthermore, MCDM-based systems depend on the required criteria information, i.e., the optimum mining method can only be selected when all relevant determining criteria are available. Lastly, the subjective judgement from the decision-makers may be customised to a particular mining project, i.e., criteria that are important for a particular project might not apply to a different project.

- **Machine learning-based systems**

Technological advancement, innovation and big data have led to the growth of artificial intelligence (AI) and machine learning (ML) [32]. AI is a field of data science aimed at developing intelligent machine systems that simulate human intelligence for learning and complex problem-solving. ML is a subfield of AI that enables machines or computers to simulate human intelligence. ML is a process of using mathematical algorithms to train models that make future predictions without being explicitly programmed. These trained ML models learn from extensive historical data to make future predictions or decisions.

AI and ML have been applied in solving complex problems from different fields of science and engineering, including mining, especially in the MMS discipline. Few studies have evaluated the applicability of artificial neural networks (ANN) in MMS. ANN is an ML algorithm inspired by the biological neural network of humans (or animals) [33], [34]. Lv and Zhang [14] applied ANN to develop a model specialized in predicting mining methods for a thick coal seam. In their investigation, three coal seam mining methods were evaluated: the caving coal mining method, the large mining height mining method, and the slice mining method. The input parameters for their ANN model corresponded to ten factors (i.e., coal seam angle, coal seam thickness, roof condition, floor condition, gas condition, fault condition, stability of coal seam and workers) with three output factors (i.e., mining method, yields and ergonomics). From the thirty coal seam case studies, twenty-six datapoints were employed as training sets under a supervised learning environment, whilst the other four were employed as testing sets. Their ANN model could predict the outputs (i.e., mining method, yields and ergonomics) for the test samples with a considerable performance, hence demonstrating the effectiveness of ANN in MMS.

In 2018, Chen and Shixiang [15] evaluated the effectiveness of ANN in selecting mining methods for a thin coal seam. Their ANN model has been trained based on thirty-three samples from field investigations, literature, and questionnaires. Of the thirty-three samples, twenty-three were employed as the training dataset, eight as the testing dataset, and two as the validation dataset. The input parameters correspond to six factors: thickness, dip angle, variability, Provodnikov's hardness of thin seam, fault occurrence characteristics and length of the panel. The model outputs two factors: the mining method and the daily production of the panel. The studies by Lv and Zhang [14] and Chen and Shiaxiang [15] thus highlight the effectiveness of ANNs in underground MMS in thick and thin coal seams, respectively.

In 2020, Ozyurt and Karadogan [16] investigated the applicability of ANNs and game theory to develop a model for underground MMS. The study was based on a mixture of six different ANN models to evaluate orebody geometry, rock mass properties, environmental factors, and ventilation conditions to evaluate the technical feasibility of eleven underground mining methods. The six ANN models were trained using synthetic data and tested using real-world data from literature and the mining industry. The modified version of the UBC-MMS tool developed by the authors mentioned above was used to verify the level of the practicability of the synthetic samples. The output from the first to the fifth ANN models aimed to evaluate the mining methods in terms of oxidation risk, dust/gas explosion risk, caving methods, pillar methods and mechanization, respectively. The last ANN model outputs technical scores of eleven mining methods, including longwall mining, diagonal longwall, shrinkage stoping, cut and fill stoping, top slicing, sublevel caving, open-room, room and pillar, sublevel stoping, block caving and square set stoping. After that, the most viable mining method is selected using the ultimatum game theory, i.e., players (decision-makers) have a task to select the mining method (s) that satisfies both safety and economy. The input parameters for all six ANN models represent about nineteen factors such as ore type, orebody shape, orebody thickness, orebody plunge, depth of the orebody, grade, grade distribution, ore-RMR, footwall-RMR, hangingwall-RMR, ore-RSS, footwall-RSS, hangingwall-RSS, separation, between ore and rock, underground water flow velocity, risks of oxidation and dust/gas explosion, subsidence effect, and ore's economic value. Their study further demonstrated the effectiveness of ANNs in developing a robust system for underground MMS for different ore types and detailed evaluation of relevant criteria (influencing factors) using synthetic data. Moreover, according to the authors, their model can be applied even when there is a lack of information regarding the relevant criteria (the required input parameters).

Shohda et al. (2022) [17] also employed ANN in underground MMS. Their ANN was based on the input parameters of the UBC tool and data collected from a mine site. Their study compared the results of the ANN model with a commonly used MCDM TOPSISb technique. ANN model for MMS provided similar results as the TOPSIS more easily and accurately.

In light of improving and extending the application of AI and ML, this study introduces the application of recommendation system technologies in the MMS discipline. Recommendation systems are AI systems that use ML and big data to predict future users'/consumers' preferences and recommend the most relevant items.

1.2.3 Recommendation systems

Recommendation systems (also known as recommender systems) [18], [35], [36] are powered by the growth of Web-based business, big data and the availability of overwhelming content and products, which makes it difficult for consumers or users to make the best decision. Recommendation systems help users deal with information overload by filtering relevant information and making personalized recommendations. Recommendation systems have been successfully implemented in different domains such as retail, media and entertainment, web search, and e-learning. A good review of the different application domains of recommendation systems is done in [37]. Amazon.com

[38], [39] is a well-known retail company that has successfully implemented recommendation systems for personalisation products. Different models of recommendation systems power Netflix, YouTube, Spotify, and Facebook to offer personalized recommendations for movies, TV shows, videos, and social networks [40], [41]. The main goal of recommendation systems is to boost product sales by offering relevant, diverse, new products (or content) recommendations, thus boosting user (or consumers') retention (and satisfaction) and business profits. Figure 1.1 depicts the interconnection between artificial intelligence (AI), machine learning (ML) and recommendation systems under the data science umbrella. Figure 1.1 shows recommendation systems as a multidisciplinary field, including various AI and data science subfields such as machine learning, statistics, and data mining.

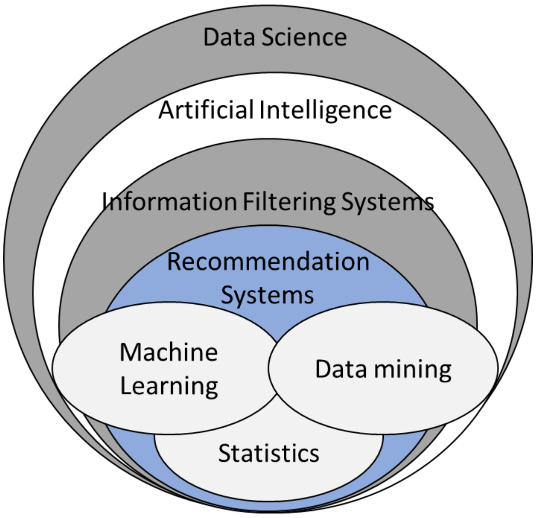


Figure 1.1: Interconnection between data science, artificial intelligence, information filtering and recommendation systems, including machine learning, data mining and statistics

There are different types of recommendation systems, and they can be classified based on the data source or type of data required to produce recommendations. Most recommendation systems rely on user-item interaction data, users' profiles, and the items' content descriptions. User-item interaction data refers to users' buying behaviour (implicit data), and the rating users give to the items. Users' profile and items' content are attributes or features that describes both users and items. Recommendation systems are commonly classified as collaborative filtering, content-based, knowledge-based, demographic-based, community-based and hybrid recommendation systems [18], [35], [42], [43].

Content-based and collaborative filtering are the two most common types of recommendation systems. Content-based recommendation systems make recommendations based on the items' “content” (i.e., attribute/features of items) combined with the ratings or preferences of users [21], [43]. Therefore, recommendations are predicted based on a user's historical information: users’ buying behaviour, the items’ content and ratings. In contrast, in collaborative filtering (CF) systems [21], [43], recommendations are generated based on collaboration between different users: the recommendations are made based on the preferences of other users. The “collaboration” is made through different users' ratings for items or products. Here, algorithms use the user-item interaction matrix as training data to find similar users, predict ratings for users that have not rated items, and then suggest similar products to similar users. CF systems are divided into two approaches: memory-based and model-based [44], [45]. The memory-based, also known as neighbourhood-based, recommendations are made by finding similarities between different users to predict a user rating (of unrated items). Cosine-similarity and Pearson correlation is used to measure and find users’ similarities. On the other hand, the model-based approach applies ML algorithms to train models to predict users (ratings of unrated items) and generate recommendations. Figure 1.2 shows the detailed classification of the two most common recommendation systems, stressing the CF systems.

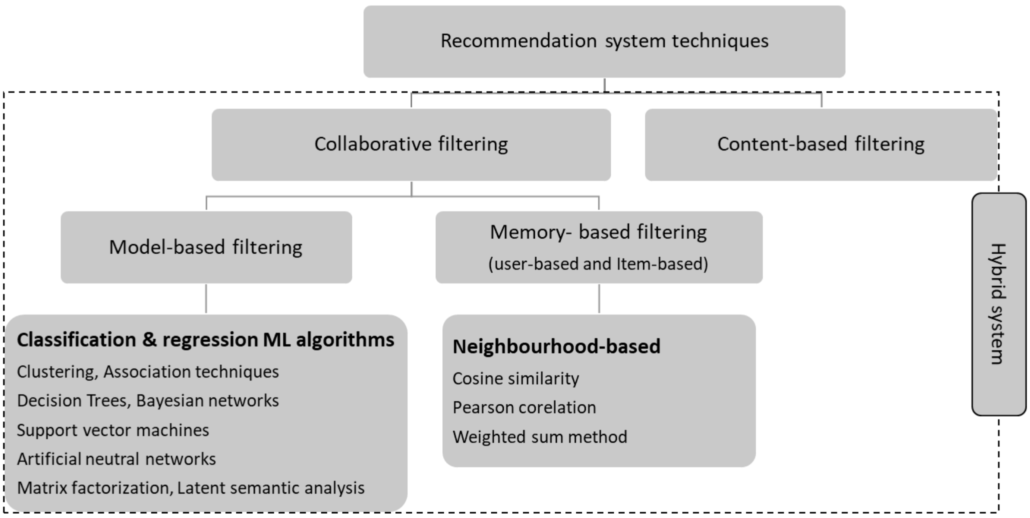


Figure 1.2: Classification of different types of recommendation systems

This study thus investigates the applicability of the collaborative filtering approach in underground mining methods selection by exploring mining projects' historical data from an open-source database.

Table 1.1: Summary and description of mining methods selection (MMS) systems, including this study's proposed system

System approach	Proposed systems
QUALITATIVE SYSTEMS Qualitative classification scheme or flowchart for MMS Factors: orebody geometry, geology, and geotechnical properties Methods: surface and underground	Boscov and Wright, 1973 [1], [3]: system for underground MMS Morrison, 1976 [1], [3]: a system for underground MMS Laubscher, 1981 [1], [3]: system for underground MMS Hartman, 1987 [1]: system for surface and underground MMS
QUANTITATIVE SYSTEMS Assign numerical ranks to the factors to evaluate the feasibility of the mining methods Factors: orebody geometry, geology, and geotechnical properties Methods: open pit and underground The relative importance of the factors is not considered in the evaluations	Nicholas, 1981 [5]: developed the Nicholas Approach Miller et al., 1995 [4]: developed the UBC-MMS tool: a modified version of the Nicholas Approach
MCDM-BASED SYSTEMS Apply different MCDM techniques in MMS Include a broad category of factors and methods <u>Considering the relative importance of the factors</u> Decision-makers subjectively determine the relative importance of the factors	Bitarafan and Ataei, 2004 [7]: Applied Fuzzy MCDM tools for MMS Ataei et al., 2008 [9]: Applied AHP technique for MMS Namin et al., 2008 [13]: Applied fuzzy TOPSIS technique for MMS Alpay and Yavuz, 2009 [10]: Applied AHP and Yager’s method for MMS Azadeh et al., 2010 [8]: Modified Nicholas approach based on FAHP Bogdanovic et al., 2012 [11]: Integrated AHP and PROMETHEE to develop a method for MMS Shariati et al., 2013 [12]: Integrated FAHP and TOPSIS to develop a method for MMS
MACHINE LEARNING-BASED SYSTEMS Use historical data to build artificial neural networks (ANN) models for MMS Factors and methods selected depending on the model	Lv and Zhihui, 2014 [14]: developed an ANN model for thick coal seam underground MMS Chen and Shixiang, 2018 [15]: developed an ANN model for thick coal seam underground MMS Ozyurt and Karadogan, 2020 [16]: Integrated ANN and game theory to build a model for selecting underground mining methods for different ore types
THIS STUDY’S PROPOSED SYSTEM: Expand the application of artificial intelligence (AI) and machine learning (ML) by introducing the recommendation system approach in MMS: **<u>Develop an AI-based mining methods recommendation system (AI-MMRS)</u>**	

1.3 Purpose of the Study

This study expands the application of artificial intelligence (AI) and machine learning (ML) in the mining methods selection (MMS) discipline by introducing the application of the recommendation system approach. Recommendation systems have been applied in different domains to help users deal with information overload, thus, improving users' decision-making ability/quality. This study investigates the possibility of incorporating AI to explore available mining projects database for developing a system that can aid in decision-making when planning a mining project (i.e., the mine planning process). The general aim of the study is to develop an AI-based mining methods recommendation system (AI-MMRS) by filtering relevant information from mining projects' historical data. As such, the study implements the collaborative filtering approach to develop a system that recommends the most appropriate mining methods by learning from previous mining projects' procedures focusing on underground mining methods, as the selection of underground mining methods is usually the most challenging task due to the complexity associated with orebody deposit characteristics. The study not only introduces the implementation of the recommendation systems approach in MMS, but the proposed system integrates different strategies attempting to address the challenges/gaps in the previous MMS systems. Therefore, the study has the following aims:

- Determine the weights and the most relevant factors in MMS objectively without the direct involvement of decision-makers.
- Strive to create input datasets by filtering relevant information from mining projects' historical data.
- Investigate the applicability of a memory-based collaborative filtering approach for predicting and recommending underground mining methods.
- Assess the capability of the nonnegative matrix factorization algorithm to address the data sparsity problem by predicting underground mining methods and other variables critical for MMS (for augmenting the input datasets).
- Investigate the capability of ML classification algorithms to predict (classify) underground mining methods.

Additionally, this study reviews the impact of scientific and technological advancement in implementing underground mining methods in the late 2000s.

1.4 Overview of the Study

This study comprises seven chapters; Chapter 1 provides a general introduction, including the literature review and the purpose of the study. Chapter 2 introduces the application of the Entropy method to estimate the relative importance of the factors influencing the MMS process to identify the most relevant factors that will be used as the main variables in the input datasets in Chapter 3. Chapter 3 gives an overview of study data from the data collection, management, and analysis to create the input datasets based on results from Chapter 2. These input datasets will be used as a base to evaluate different models in Chapters 4, 5 and 6. Chapter 3 additionally reviews the trend of the commonly implemented mining methods in the late 2000s following technological and scientific advancements. Chapter 4 investigates the applicability of the memory-based collaborative filtering approach to predict and recommend underground mining methods. Chapter 5 assesses the capability of the nonnegative matrix factorization (NMF) algorithm to address the data sparsity problem, which is one of the study's limitations (faced in Chapter 3). As such, Chapter 5 assesses the capability of the NMF to predict mining methods and other variables critical for mining methods selection (MMS). Chapter 6 further investigates the capability of machine learning classification algorithms to predict (classify) underground mining methods. Lastly, Chapter 7 presents the study’s conclusion, the significance of the proposed AI-MMRS and the study’s contribution. Figure 1.3 shows the overview of the study.

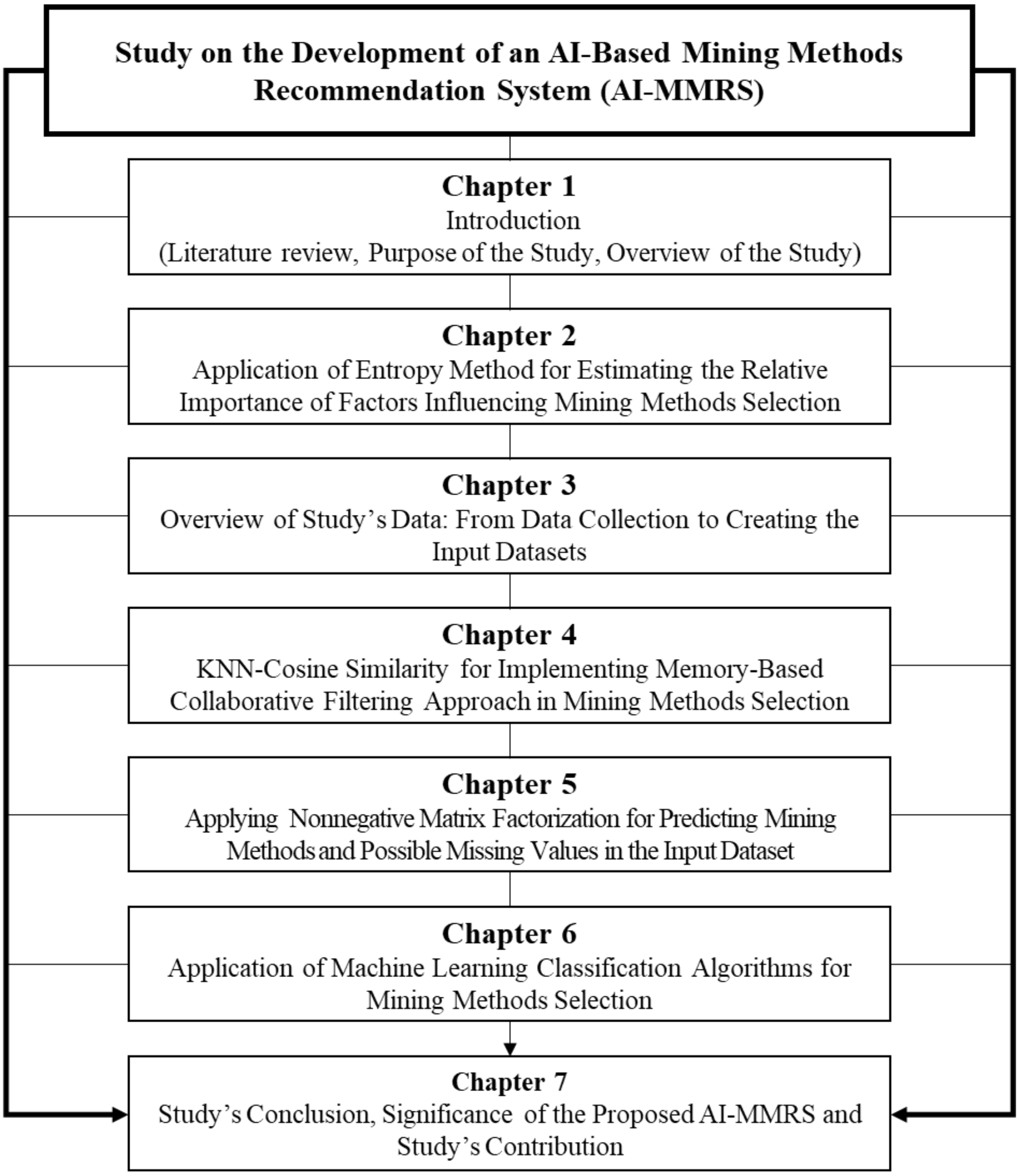


Figure 1.3: Overview of the study (KNN: k-nearest neighbours)

2 Application of Entropy Method for Estimating the Relative Importance of Factors Influencing Mining Methods Selection

In this chapter, the multi-criteria decision-making (MCDM) Entropy method is incorporated as a technique for feature selection. Feature selection is an essential pre-processing step for improving the performance and quality of the models for the AI-based mining methods recommendation system (AI-MMRS). This chapter aims to determine the weights and most relevant factors in mining methods selection (MMS) without the direct involvement of decision-makers. As such, the Entropy method is applied to estimate the relative importance of the factors influencing the MMS process to identify the most relevant factors that will be used as the main variables in the input datasets created in chapter 3.

2.1 Why Entropy Method for Estimating the Relative Importance of the Factors

Naturally, mining methods selection (MMS) is a complex task given the many factors that need to be considered as input variables. The factors that mainly influence the MMS process include the physical characteristics of the deposit (orebody geometry), geology, geotechnical properties, technological, economic, and environmental. In quantity, over twenty input variables (or factors) influence the MMS process. Usually, these input variables should be considered to train machine learning models for developing the AI-mining methods recommendation system (AI-MMRS). However, considering that many factors as input variables to train models may negatively affect the performance of the models owing to the noise and biases caused by redundant input variables and the complexity associated with computation time.

For this reason, the Entropy method is proposed as a feature selection technique to assess the relative importance of factors influencing the MMS process and identify the most relevant factors. In machine learning (ML), different methods are used for feature selection [46], [47] to reduce the number of features in a dataset by selecting the most relevant features, thus, improving the performance of the prediction models and reducing computation time. However, these methods usually require large datasets to analyze features correlation and identify the relevant features effectively. Entropy is a multi-criteria decision-making (MCDM) technique that determines factors (or criteria) objective weights only based on a decision matrix: it does not require large datasets as other ML methods. Furthermore, the Entropy method estimates the relative importance of the factors without the direct involvement of decision-makers (the opinion or judgement of mining engineering experts) [31], unlike most MCDM-based MMS systems, which are mainly based on subjective opinions from decision-markers (mining engineers). The Entropy method avoids a certain level of bias associated with the decision-makers' subjective (customized) opinions and conflicting views about the factors' relative importance, which may inherently affect the eligibility and accuracy of the assessments. In addition, the subjective judgement from the decision-makers is mainly customised to a particular mining project; in other words, factors or criteria considered relevant in one project may not be transferable to a different project. Using the Entropy method to estimate factors' objective weights, we will produce non-customized results that can be implemented in any case study, especially when the decision-makers' direct judgement is unavailable (totally or partially) or even not required [31].

2.2 MCDM-Entropy method to estimate criteria weights

Multi-criteria decision-making (MCDM) is a branch of operations research (OR) that attempts to solve real-life problems that involve different alternatives by considering several conflicting criteria to achieve specific goals. MCDM attempts to solve problems of selecting an alternative from a set of alternatives under several criteria, typically aiming at a single goal [7]. To address these problems, the decision maker's team performs the decision-making process based on the hierarchical structure model, wherein the first step is to define the goal and then identify the alternatives for achieving the goal and the criteria used to compare the alternatives [48]. MCDM techniques evaluate the performance of different alternatives based on the criteria weights, wherein the best alternative is selected as the one with the highest performance rates. The weights of each criterion express their relative importance for the decision. Typically, decision-makers may define and assign subjective weights to each criterion based on their intuition and judgement, commonly using methods such as the utility preferences function, analytic hierarchy process (AHP) and fuzzy version of classical linear weighted averages [7]. However, often, decision-makers have conflicting views on the values of weights or are simply uncertain of the relative importance of each criterion. In this case, the Entropy method [49] is applied to determine the objective weights of each criterion based on a decision matrix, wherein the preferences or judgement of decision-makers are entirely or partially unavailable or even not required [50].

The term Entropy is applied in different scientific fields (e.g., physics, chemistry, biology, mathematics, psychology, and information theory); in information theory, this term plays a vital role in measuring the uncertainty associated with random phenomena of the expected information content of a specific message [51]. The MCDM Entropy method also called Shannon's Entropy [49], is a technique applied in MCDM to estimate criteria objective weights. Figure 2.1 illustrates the flowchart of the overall procedures of the Entropy method, wherein the first step involves the generation of the decision matrix (DM) of the problem as follows:

$$\mathrm{DM} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ x_{m1} & x_{m2} & \cdots & x_{mn} \end{bmatrix} \tag{2.1}$$

Where x_{ij} is the criteria/sub-criteria rate, n is the number of criteria/sub-criteria, and m is the number of alternatives.

In the second step, the DM data are normalized by applying Equation (2.2) to make all the criteria comparable by transforming different scales and units among several criteria into standard measurable units [50]:

$$r_{ij} = \frac{x_{ij}}{\sum_{j=1}^{n} x_{ij}}, i = 1,2 \ldots, m;\ j = 1,2 \ldots n \tag{2.2}$$

Where r_{ij} is the normalized criteria/sub-criteria rate.

Then, the Entropy (E_j) values are computed by applying Equation (2.3). The entropy value measures the degree of uncertainty between the set of alternatives in the DM when no preference among criteria can be established [50], [52], [53].

$$E_j = -h \sum_{i=1}^{m} r_{ij} \ln{(r_{ij})},\ j = 1,2, \ldots, n, h = \frac{1}{ln\ (m)} \tag{2.3}$$

Where $r_{ij}ln(r_{ij}) = 0$ if $r_{ij} = 0$ and h is the entropy constant.

The fourth step is calculating the diversity (D_j) or the degree of diversification based on the entropy values using Equation (2.4). Diversity measures the level of diversity in the evaluation of a set of alternatives for the same criterion [50], [54], [55]. In other words, diversity measures the variation or the degree of dispersion between the rates of different alternatives for the same criterion. The higher the variation or dispersion, the higher the diversity, and the more valuable the criterion:

$$D_j = 1 - E_j \tag{2.4}$$

Finally, the relative importance of the criteria, measured by the objective weight, is calculated based on Equation (2.5). The relative importance of the criteria is directly related to the amount of data essentially provided by a set of alternatives for the same criterion [50], [52]:

$$w_j = \frac{D_j}{\sum_{j=1}^{n} D_j},\ j = 1,2,\dots,n \tag{2.5}$$

Where w_j is the degree of importance of criterion j or object weight of criterion j.

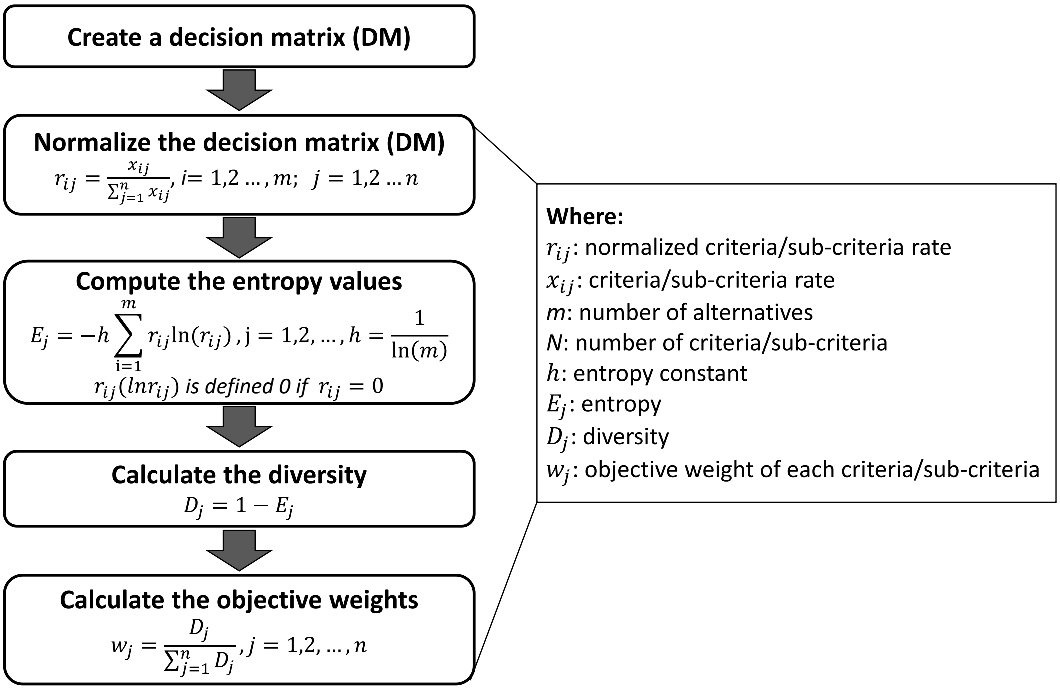


Figure 2.1: Procedures of the entropy method for calculating the objective weights of the criteria

2.3 Application of the Entropy Method to Estimate the Relative Importance of the Factors Influencing Mining Methods Selection

The MMS is an MCDM problem as it is a task that involves several conflicting factors (or criteria) to evaluate the feasibility of different mining methods (or alternatives). For this reason, different MCDM techniques have been applied in the MMS process, and several studies [8], [10]–[13], [54], [7], [9], [56], [57] have proven the applicability and advantages of different MCDM techniques in MMS. One of the critical steps in MCDM techniques is to define the relative importance of the selected criteria. In most MCDM techniques, the relative importance of the criteria is defined subjectively; criteria weights are defined based on the direct opinion and judgement of decision-makers (or mining engineering professionals). In essence, the subjective (customized) judgement introduces a certain level of bias, which inherently affects the eligibility and accuracy of the assessments. Furthermore, the subjective judgement from the decision-makers is mainly customised to a particular mining project; in other words, factors or criteria considered relevant in one project may not be transferable to a different project.

For the aforementioned reasons, the Entropy method is applied to estimate the relative importance of the factors influencing MMS by determining their objective weights without decision-makers' direct involvement. Then, based on objective weights, the most relevant ones are identified and used as main variables in the input dataset for the proposed AI-MMRS. The procedures for applying the Entropy method to determine the objective weights of the factors influencing the MMS are described below. The first step is to create the decision matrix.

Decision matrix (DM)

Table 2.1 presents the DM created based on the approaches proposed by Miller et al. [4], developers of the UBC-MMS tool, and Hartman and Mutmansky [1], who created a guideline base to compare different surface and underground mining methods. The DM describes the classification of different surface and underground mining methods based on the influencing factors. The DM comprises twenty factors classified in orebody geometry, geology and geotechnical properties, and economic, technological, and environmental factors. The factors (criteria) are used to evaluate twelve mining methods (alternatives), including surface and underground.

Table 2.1: DM, own development based on [2,17] approaches

	Host-rock strength	Ore strength	Ore uniformity	Dip	Shape
Block caving	Weak-fair	Weak-fair	Gradational	Moderate-deep	Tabular-equidimensional
Open pit	Any	Any	Any	Shallow	Any
Shrinkage stoping	Strong-very strong	Fair-strong	Uniform	Shallow-moderate	Tabular
Square set	Weak-fair	Very weak-weak	Erratic	Deep	Irregular
Longwall	Weak-fair	Very weak-weak	Uniform	Moderate-deep	Tabular
Solution mining	Weak-fair	Weak-fair	Erratic	Shallow	Any
Sublevel stoping	Strong-very strong	Fair-strong	Gradational	Moderate	Tabular
Sublevel caving	Weak-fair	Weak-fair	Gradational	Moderate	Tabular-equidimensional
Open cast	Any	Any	Gradational	Shallow	Tabular
Cut and fill	Weak-fair	Fair-strong	Erratic	Moderate-deep	Irregular-tabular
Stull stoping	Fair	Strong-very strong	Erratic	Moderate	Irregular-tabular
Room and pillar	Fair-strong	Weak-fair	Gradational	Shallow-moderate	Tabular

	Dip	Thickness	Health and safety	Stability of openings	Recovery
Block caving	Steep	Very thick	Good	Moderate	High
Open pit	Flat-intermediate	Thick-very thick	Good	High	High
Shrinkage stoping	Steep	Narrow-intermediate	Good	High	High
Square set	Any	Very narrow-narrow	Poor	High	Very high
Longwall	Flat	Very narrow-narrow	Good	High	High
Solution mining	Steep	Any	Good	Moderate	Very low
Sublevel stoping	Steep	Intermediate-thick	Good	High	Moderate
Sublevel caving	Steep	Thick-very thick	Good	Moderate	High
Open cast	Flat	Moderate	Good	High	High
Cut and fill	Intermediate-steep	Narrow-intermediate	Moderate	High	High
Stull stoping	Intermediate-steep	Narrow	Moderate	Moderate	High
Room and pillar	Flat	Narrow	Good	Moderate	Moderate

	Flexibility	Dilution	Selectivity	Depth capacity	Development rate
Block caving	Low	High	Low	Moderate	Slow
Open pit	Moderate	Moderate	Low	Limited	Rapid
Shrinkage stoping	Moderate	Low	Moderate	Limited	Rapid
Square set	High	Very low	High	Unlimited	Slow
Longwall	Low	Low	Low	Moderate	Moderate
Solution mining	Low	Very high	Low	Limited	Moderate
Sublevel stoping	Low	Moderate	Low	Moderate	Moderate
Sublevel caving	Moderate	Moderate	Low	Moderate	Moderate
Open cast	Moderate	Low	Low	Limited	Rapid
Cut and fill	Moderate	Low	High	Moderate	Moderate
Stull stoping	High	Low	High	Limited	Rapid
Room and pillar	Moderate	Moderate	Low	Limited	Rapid

	Productivity	Ore grade	Mining cost	Production rate	Capital investment
Block caving	High	Moderate	Low	Large	High
Open pit	High	Low	Very low	Large	High
Shrinkage stoping	Low	Moderate	Moderate-high	Moderate	Low
Square set	Low	High	Very high	Small	Low
Longwall	High	Low	Low	Large	High
Solution mining	Very high	Very low	Low	Moderate	Moderate
Sublevel stoping	High	Low-moderate	Moderate	Large	Moderate
Sublevel caving	Moderate	Moderate	Low	Large	Moderate
Open cast	High	Low	Low	Large	High
Cut and fill	Moderate	High	High	Moderate	Moderate
Stull stoping	Low	High-very high	High	Small	Low
Room and pillar	High	Low-moderate	Moderate	Large	High

The twelve mining methods or alternatives (A) in the DM include: block caving (A1), open-pit (A2), shrinkage stoping (A3), square set (A4), longwall (A5), solution mining (A6), sublevel stoping (A7), sublevel caving (A8), open-cast (A9), cut and fill (A10), stull stoping (A11), and room and pillar (A12). In addition, the factors or criteria (c) considered are described below [1], [3], [23], [58]–[62]:

<u>Geotechnical and geological properties</u>

- *Host-rock strength (c1):*

This factor is related to the properties of the rock surrounding the ore deposit, measuring the hardness or toughness of the rock against permanent deformation. The strength of the rock (host and ore) can be very weak, weak, fair, strong, and very strong. Understanding host-rock strength is crucial in mining methods selection (MMS) to ensure the safety and stability of openings (in surface and underground mining). Host-rock properties play a huge role in selecting the different classes of underground mining methods (supported, unsupported and caving). Furthermore, understanding rock conditions in surface or underground mining methods is crucial to determine the pit slope angle (in surface mining) and the support systems (in underground mining).

- *Ore strength (c2):*

Ore strength is related to the mechanics of the ore or even ore properties. In selecting both surface and underground mining methods is crucial to understand the properties of the ore to determine the extraction methods (mechanical or blasting), the support systems, equipment selection, and the stability of openings.

- *Ore uniformity (c3):*

Ore uniformity is a geological factor corresponding to ore grade distribution throughout the ore deposit. Ore uniformity is determined based on ore grade variation from the average grade within the ore deposit. The ore distribution can be variable/erratic, gradational and uniform. It is variable when the grade values within the deposit change radically over a short distance and do not show any perceptible pattern in their changes. Gradational is when grade values at any point within the deposit have zonal characteristics, and the grades change gradually from one to another. Uniform when grade values at any point within the deposit do not vary significantly from the average grade. It is essential to understand the ore distribution to select the most suitable mining method to ensure high selectivity, recovery, and low dilution. Additionally, this factor is directly related to the selectivity of a mining method, i.e., the poor the ore distribution, the more selective the mining method should be.

Physical characteristics of the orebody deposit (orebody geometry)

- *Depth (c4):*

This factor corresponds to the depth of the ore deposit relative to the surface ground. An ore deposit can be shallow (< 100 m), intermediate (100–600 m), and deep (> 600 m). Depth is usually a key factor in selecting between surface and underground methods. For surface, deposit depth is applied to decide between casting the waste (in open-cast) or haulage the waste to dump sites (in open-pit) and applying solution mining. Additionally, some underground methods are less suitable for deep deposits owing to the limited depth capacity.

- *Shape (c5):*

Shape refers to the form of the ore deposit, which can usually be tabular, equidimensional/massive and irregular. Tabular deposits extend at least hundreds of meters along two dimensions and substantially less along a minor dimension. Equidimensional (massive) deposits have all dimensions in the same order of magnitude. In irregular deposits, the dimensions vary over short distances. It is important to understand the ore deposit shape for mining methods selection as some methods (i.e., longwall, open cast, room and pillar) are more suitable for tabular deposits than others.

- *Dip (c6):*

The ore deposit dip is the angle of inclination of a plane measured downward, perpendicular to the strike direction. An ore deposit can be flat (< 20°), intermediate (20–55°) and steep (> 55°).

The dip is essential in selecting both surface and underground mining methods. In surface mining, the dip is usually applied to decide between the open cast/stripping mining (used for flat deposits), open pit or solution mining (usually for intermediate or steep). Moreover, some underground mining methods (shrinkage stoping, sublevel stoping, stull stoping and caving methods) are more suitable to exploit intermediate or steep deposits because they rely on gravity for material flow and cannot be applied in flat deposits.

- *Thickness (c7):*

This factor refers to one of the three dimensions of the ore deposit. The thickness can vary throughout ore deposits, being very narrow (< 3 m), narrow (3–10 m), intermediate (10–30 m), thick (30–100 m), and very thick (> 100 m). The thickness of the ore deposit determines the effectiveness of some mining methods, as some methods (open pit and caving methods) are less effective in narrow deposits. Additionally, this factor affects the mechanization (and equipment selection) and the selectivity of specific mining methods.

Environmental considerations

- *Health and safety (c8) and stability of openings (c9):*

The stability of openings determines the health and safety of mining operations. The health and safety of the mining operators should be a top priority objective in preventing hazards that unappropriated mining methods for a particular ore deposit can cause. Therefore, it is important always to consider mining methods with high stability of openings providing good health and safety conditions.

Technological Factors

- *Recovery (c10) and dilution (c12):*

Recovery is the capability of a mining method to extract valuable ore from the deposit entirely. Ore recovery is the percentage of mineable reserves extracted in the mining process. On the other hand, dilution is the waste material mixed with ore during the extraction, which is then sent to the processing plant. Dilution is the percentage of the waste mined and sent to the processing plant over the combined total ore and waste material milled. Recovery and dilution are usually interrelated, as some mining methods with high recovery usually involve contamination of the ore from the waste. Some mining methods have low recovery due to the need to leave the ore as structural support whilst providing moderate to low dilution.

- *Flexibility (c11) and selectivity (c13):*

Flexibility refers to a mining method's ability to adapt to changes related to mining conditions, market price and technology throughout the mine life. Selectivity refers to the separate extraction of ore and waste (or gangue), ensuring complete ore extraction with low dilution. Flexibility marries well with the selectivity of a mining method to determine the success of a project-the more flexible and selective, the more effective the mining method is.

- *Depth capacity (c14):*

This factor measures the capability of the mining method in terms of ore deposit depth. Mining methods with limited depth capacity (open-pit, open-cast, solution mining, room and pillar, stull stoping and shrinkage) are not suitable for extracting deep ore deposits; hence, the importance of considering depth capacity in MMS.

- *Development rate (c15):*

Mine development rate is the time (or speed) spent undertaking operations (tunnelling, sinking, crosscutting, drifting, raising, stripping, construction of mine infrastructures, etc.) that prepare the mine for ore extraction. This factor directly affects capital investment because the slower the development rate, the higher the capital costs or investment. Hence, it is crucial to consider this factor during MMS.

Economic factors

- *Productivity (c16):*

Productivity is the measure of the efficiency or performance in the mine in terms of how well/smart the inputs (labour, materials, equipment, capital investment, resources) are converted into outputs (gross output, value-added). This factor involves most of the parameters used to measure the efficiency of specific mining methods. Therefore, it is crucial to consider productivity during the MMS process.

- *Ore grade (c17):*

The grade is used to measure the quality of an ore deposit; the higher the grade, the more valuable the deposit is. It is essential to consider this factor during the MMS process to ensure the efficiency and effectiveness of mining operations. Mining methods with high operating costs are usually applied to high-grade deposits to be economical. Moreover, large-scale mining methods may be economically appropriate for low-grade deposits.

- *Mining costs (c18) and capital investment (c20):*

Mining costs are the expenses (mine development, rehabilitation, exploration and grade control activities, material and utility handling, maintenance, and labour cost) resulting from all operations or activities necessary to extract the ore. Mining costs are usually measured in terms of the money necessary to mine a tonne of material (ore and waste). At the same time, capital investment is the amount of money necessary to invest in the mining project to pursue the objectives (growing operations and generating revenue). It is crucial to consider these factors during the MMS process, and usually, underground mining methods require high capital investment.

- *Production rate (c19):*

The production rate corresponds to the quantity of material (ore and waste) extracted per hour, day, month, and year. The production rate of a mine highly relies on the selected mining method; thus, the need to consider this factor during the MMS process. Usually, large-scale mining methods have a higher production rate, and low-scale methods have otherwise.

In the DM, each row describes an alternative (A) or mining method, and each column describes the performance of each alternative (mining method) against each criterion (c) or factor. As shown in Table 2.1, the DM is composed of categorical/qualitative values, most of which are presented in the qualitative classification system.

The Entropy method is effective and accurate for numerical/quantitative criteria values, wherein some or all pertinent decision data are available [53]; hence, the qualitative classification values must be transformed into quantitative values. For this, an appropriate weighting system is used, as shown in Figure 2.2. The weighting system shown in Figure 2.2 is composed of 10 points, from 0 to 9. Before transforming the qualitative values in the decision matrix (shown in Table 2.1), first, the qualitative classification of the factors belonging to geotechnical properties and orebody geometry (physical characteristics) is transformed into an adequate qualitative classification compatible with the weighing system. Table 2.2 shows the approach implemented to transform the qualitative values in geotechnical properties and orebody geometry.

Table 2.2: Approach for transforming the qualitative classification of the geotechnical properties and orebody geometry

Factors	Transformation of the factors classification system
Ore and host rock strength	Very weak = very poor Weak = poor Fair = moderate Strong = good Very strong = very good
Orebody thickness	Very narrow = very small Narrow = small Intermediate = moderate Thick = large Very thick = very large
Orebody shape	Irregular = unfavourable Tabular = average Equidimensional = favourable
Ore uniformity	Erratic/variable = poor Gradational = moderate Uniform = good
Dip	Flat = low Intermediate = moderate Steep = high
Depth below the surface	Shallow = low Intermediate = moderate Deep = high

After adjusting or transforming the qualitative values using the approach shown in Table 2.2, the values in the decision matrix are then transformed into quantitative/numerical values. The weighing system depicted in Figure 2.2 is applied to transform all the qualitative values in the decision matrix (shown in Table 2.2) into quantitative values, resulting in a numerical DM, as presented in Table 2.3.

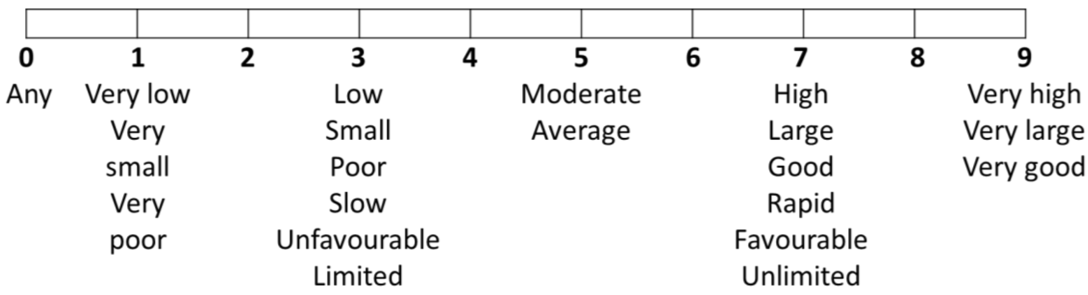


Figure 2.2: Weighting system to transform qualitative values in DM into qualitative values, where the values 0, 1, 3, 5, 7 and 9 are described, and 2, 4, 6 and 8 stand for intermediate values

Table 2.3: Transformed DM with quantitative values

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20
A1	4	4	5	6	6	7	9	7	5	7	3	7	3	5	3	7	5	3	7	7
A2	0	0	0	3	0	4	8	7	7	7	5	5	3	3	7	7	3	1	7	7
A3	8	6	7	4	5	7	4	7	7	7	5	3	5	3	7	3	5	6	5	3
A4	4	2	3	7	3	0	2	3	7	9	7	1	7	7	3	3	7	9	3	3
A5	4	2	7	6	5	3	2	7	7	7	3	3	3	5	5	7	3	3	7	7
A6	4	4	3	3	0	7	0	7	5	1	3	9	3	3	5	9	1	3	5	5
A7	8	6	5	5	5	7	6	7	7	5	3	5	3	5	5	7	4	5	7	5
A8	4	4	5	5	6	7	8	7	5	7	5	5	3	5	5	5	5	3	7	5
A9	0	0	5	3	5	3	5	7	7	7	5	3	3	3	7	7	3	3	7	7
A10	4	6	3	6	6	6	4	5	7	7	5	3	7	5	5	5	7	7	5	5
A11	5	8	3	5	6	6	3	5	5	7	7	3	7	3	7	3	8	7	3	3
A12	6	4	5	4	5	3	3	7	5	5	5	5	3	3	7	7	4	5	7	7

2.4 Analytical Results of the Application of the Entropy Method

The values in the original DM in Table 2.3 are normalized by applying Equation (2.2), resulting in a normalized matrix, as presented in Table 2.4. Then, by applying Equations (2.3)–(2.5), the entropy values (E_j), diversity (D_j), and objective weights (W_j) are computed, as presented in Table 2.5.

Table 2.4: Normalized DM

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10
A1	0.078	0.087	0.098	0.105	0.115	0.117	0.167	0.092	0.068	0.092
A2	0.000	0.000	0.000	0.053	0.000	0.067	0.148	0.092	0.095	0.092
A3	0.157	0.130	0.137	0.070	0.096	0.117	0.074	0.092	0.095	0.092
A4	0.078	0.043	0.059	0.123	0.058	0.000	0.037	0.039	0.095	0.118
A5	0.078	0.043	0.137	0.105	0.096	0.050	0.037	0.092	0.095	0.092
A6	0.078	0.087	0.059	0.053	0.000	0.117	0.000	0.092	0.068	0.013
A7	0.157	0.130	0.098	0.088	0.096	0.117	0.111	0.092	0.095	0.066
A8	0.078	0.087	0.098	0.088	0.115	0.117	0.148	0.092	0.068	0.092
A9	0.000	0.000	0.098	0.053	0.096	0.050	0.093	0.092	0.095	0.092
A10	0.078	0.130	0.059	0.105	0.115	0.100	0.074	0.066	0.095	0.092
A11	0.098	0.174	0.059	0.088	0.115	0.100	0.056	0.066	0.068	0.092
A12	0.118	0.087	0.098	0.070	0.096	0.050	0.056	0.092	0.068	0.066

	c11	c12	c13	c14	c15	c16	c17	c18	c19	c20
A1	0.054	0.135	0.060	0.100	0.045	0.100	0.091	0.055	0.100	0.109
A2	0.089	0.096	0.060	0.060	0.106	0.100	0.055	0.018	0.100	0.109
A3	0.089	0.058	0.100	0.060	0.106	0.043	0.091	0.109	0.071	0.047
A4	0.125	0.019	0.140	0.140	0.045	0.043	0.127	0.164	0.043	0.047
A5	0.054	0.058	0.060	0.100	0.076	0.100	0.055	0.055	0.100	0.109
A6	0.054	0.173	0.060	0.060	0.076	0.129	0.018	0.055	0.071	0.078
A7	0.054	0.096	0.060	0.100	0.076	0.100	0.0	0.091	0.100	0.078
A8	0.089	0.096	0.060	0.100	0.076	0.071	0.091	0.055	0.100	0.078
A9	0.089	0.058	0.060	0.060	0.106	0.100	0.055	0.055	0.100	0.109
A10	0.089	0.058	0.140	0.100	0.076	0.071	0.127	0.127	0.071	0.078
A11	0.125	0.058	0.140	0.060	0.106	0.043	0.145	0.127	0.043	0.047
A12	0.089	0.096	0.060	0.060	0.106	0.100	0.073	0.091	0.100	0.109

The Entropy is indirectly related to the objective weights and is typically measured from 0 to 1. Therefore, the closer the entropy value is to 1, the higher the level of uncertainty and the smaller the objective weight of that criterion. Additionally, diversity is directly related to the objective weight; thus, the higher the diversity in a criterion, the higher the objective weight of the same criterion. The objective weights reflect the relative importance of each factor (or criterion) in selecting the twelve mining methods (or alternatives). In this case, the results in Table 2.5 show that the factors possess a different degree of importance, with some more critical than others. Furthermore, mechanical properties, such as the strengths of the ore and host rock, have the highest diversity and, thus, the highest degree of importance among all factors. Environmental considerations, such as health and safety and the stability of openings, have the lowest diversity, i.e., the lowest degree of importance among all factors.

The results from the Entropy method emphasizes the different level of impact that the twenty factors have in selecting the twelve mining methods. Furthermore, to identify and select the most relevant influential factors, the deviation concept is then applied. The deviation was applied to determine the factors with the highest impact in MMS, i.e., the most relevant factors. The deviation of each criterion weight from the mean weight value is calculated using Equation (2.6).

$$\text{Deviation} = w_j - \bar{w};\ \bar{w} = \sum \frac{w_j}{n} \tag{2.6}$$

Where w_j is the weight of each criterion, $\bar{w}$ is the mean weight of the criteria set, and n is the number of criteria.

Table 2.5: Results of Entropy method application, showing the Entropy, Diversity, and Objective weights of the twenty factors/criteria

	Criteria (Factor)	Entropy	Diversity	Weights
c1	Host-rock strength	0.909	0.091	0.115
c2	Ore strength	0.895	0.105	0.132
c3	Ore uniformity	0.946	0.054	0.068
c4	Depth	0.985	0.015	0.019
c5	Shape	0.920	0.080	0.100
c6	Dip	0.943	0.057	0.072
c7	Ore thickness	0.917	0.083	0.104
c8	Health and safety	0.991	0.009	0.011
c9	Stability of openings	0.995	0.005	0.007
c10	Recovery	0.976	0.024	0.030
c11	Flexibility	0.982	0.018	0.022
c12	Dilution	0.955	0.045	0.057
c13	Selectivity	0.968	0.032	0.040
c14	Depth capacity	0.982	0.018	0.023
c15	Development rate	0.985	0.015	0.019
c16	Productivity	0.977	0.023	0.029
c17	Ore grade	0.961	0.039	0.048
c18	Mining cost	0.952	0.048	0.061
c19	Production rate	0.985	0.015	0.019
c20	Capital investment	0.981	0.019	0.024

The overall mean weight ($\bar{w}$) is 0.05. The deviation of each factor (criterion) weight from the mean weight is calculated based on this mean weight. Figure 2.3 depicts the results of the factors with the lowest and highest levels of impact in MMS based on the deviation concept. Based on the deviation concept, the criteria with an objective weight smaller than the mean weight produce negative deviation values and are considered to have the lowest level of impact. Furthermore, criteria with an objective weight higher than the mean weight produce positive deviation values and have the highest impact on MMS. Therefore, criteria with higher weights than the mean weight and with the smallest Entropy and the highest diversity were identified and selected as those with the highest level of impact. In this case, eight factors were identified, where ore strength had the highest weight of 0.132, followed by host-rock strength, thickness, shape, dip, ore uniformity, mining costs, and dilution with weights of 0.115, 0.104, 0.100, 0.072, 0.068, 0.061, and 0.057, respectively.

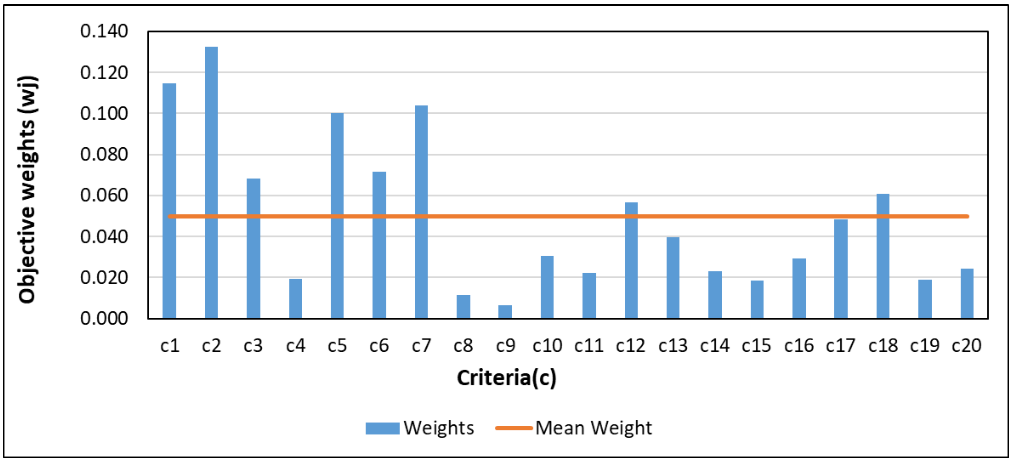


Figure 2.3: Results showing the least and the most relevant factors in MMS based on the deviation, where the most relevant are those with weights greater than the mean

2.5 Discussion on the Entropy Method Results

The results from this chapter suggest that ore strength, host-rock strength, orebody thickness, shape, dip, ore uniformity, mining costs and dilution are the most relevant in mining methods evaluation and selection, thus, limiting the use of most mining methods when compared to the rest of the factors. These results emphasize the significant impact of the physical characteristics or orebody geometry (thickness, shape, and dip of the orebody) and geotechnical properties (strength of the orebody and host rock) as well as ore uniformity on the mining methods selection (MMS) process, as also described in different MMS systems, including Nicholas' approach [5] and the UBC-MMS tool [4]. However, the factor of depth, considered necessary in the UBC-MMS tool when selecting between open-pit and underground methods, was not found to be highly important in these results because of its low diversity in the selection among the twelve mining methods (surface and underground methods). The recent implementation of mining methods also supports this fact; in practice, depth does not strictly limit the implementation of most mining methods, including the open-pit mining method. Thanks to technological and scientific advancements, there are examples of successful deep open-pit mines (in mineral deposits with depths of more than 500m). Furthermore, the results from this chapter suggest that economic and technological factors such as mining costs and dilution have a significant influence on the MMS process, thus, may limit the use of specific mining methods in the evaluations. Although, these factors are not highly emphasized during the first stage of some of the MMS systems (Nicholas' approach [5] and the UBC-MMS tool [4]).

This study objectively estimated the weights and the most relevant factors without requiring decision-makers direct involvement, unlike most MCDM-based MMS systems [14–20]. In most MCDM-based MMS systems, the relative importance of the influencing factors is determined based on the subjective opinions or judgement from decision-makers (mining engineering professionals); however, the use of subjective opinions, in essence, introduces a certain level of bias, which inherently affects the eligibility and accuracy of the assessments. In addition, the subjective judgement from the decision-makers is mainly customised to a particular mining project, i.e., decisions made in a particular project may not be transferable to a different project. The absence of subjective (customized) judgement in this study produces less biased and more generalized results, and it can be employed for different case studies assisting practical mining project decision-making and scientific studies (and research) on MMS when the opinion (or judgement) from mining engineering experts is not available or not required.

2.6 Summary of the Application of the Entropy Method to Estimate Factors Relative Importance

In this chapter, the multi-criteria decision-making Entropy method was incorporated as a technique for feature selection. Feature selection is an essential pre-processing step for developing the AI-based mining methods recommendation system (AI-MMRS). In this study, feature selection is required as a pre-processing step for improving the performance of the models by reducing the noise and bias that might be caused by many redundant factors and computation time when training the machine learning models. The Entropy method was employed to assess the relative importance of twenty factors influencing the mining methods selection (MMS) process to identify the most relevant factors in MMS. The twenty assessed factors are classified as orebody geometry, geology, geotechnical properties, technological, economic, and environmental factors. Using the Entropy method, it was possible to determine the objective weights of the twenty factors by calculating their weights without decision-makers' direct involvement. Then, based on these objective weights, factors such as ore strength, host-rock strength, orebody thickness, shape, dip, ore uniformity, mining costs, and the dilution of the mining methods were identified as the most relevant in MMS. Findings from this chapter are also supported by literature and the technological and scientific advancement in the selection and practical implementation of the mining methods. Furthermore, Using the Entropy method to determine factors' objective weights avoids any possible bias caused by the subjective (customized) judgement, thus more generalized results applicable for different case studies assisting practical mining project decision-making and scientific studies (and research) on MMS when the opinion (or judgement) from mining engineering experts is not available (fully or partially) or not required.

These results will be used as a foundation to create the input datasets in Chapter 3 and incorporated into the pre-processing data stages of Chapters 4, 5 and 6.

3 Overview of Study's Data: From Data Collection to Creating the Input Datasets

This chapter provides an overview of the study's data, from data collection, management, and analysis to creating the input datasets for evaluating different machine learning (ML) in Chapters 4, 5, and 6. As stated in Chapter 1, the availability of large historical datasets is the backbone for developing any artificial intelligence (AI) system using machine learning (ML) algorithms. In data mining, this stage is also called the data preparation process. Therefore, this Chapter strives to create the study's input datasets to train ML models for developing the AI-based mining methods recommendation system (AI-MMRS). This chapter additionally reviews the trend of the commonly implemented mining methods in the late 2000s following technological and scientific advancements.

3.1 Data Collection, Management and Analysis

The study data is mainly based on mining projects' historical data (i.e., technical reports: NI 43-101) collected from an open-source database named *SEDAR*[1]. *SEDAR* (the System for Electronic Document Analysis and Retrieval) is a mandatory document (in *pdf* format) filling and retrieval system for Canadian public companies. The National Instrument (NI 43-101) is a national instrument for the Standards of Disclosure for Mineral Projects within Canada, which sets the rules for reporting and displaying information regarding mineral properties owned or explored by companies reporting trade stock exchange within Canada. The technical reports collected for this study contain information about mining projects' development from the early exploration, advanced exploration, development (or construction) and production stages. From *SEDAR*, over 1,315 mining projects' technical reports were collected, stored, managed and analysed in a document management software (DMS). The technical reports are dated from 2000 to 2022, with mining projects in different places (in Africa, Asia, Europe, Oceania, and Central, North and South America). In the DMS, we created an appropriate searching system to optimise the efficiency of data analysis and data mining (relevant attributes extraction) processes. This searching system can also be considered an offline built-in recommendation system type, precisely a knowledge-based [18], [19] recommendation

[1] The System for Electronic Document Analysis and Retrieval (SEDAR): https://www.sedar.com/homepage_en.htm

system type. Knowledge-based [63] are more interactive recommendation systems that provide recommendations based on the user's explicit specification and attributes (or description) of the items. For example, a user might request a customised recommendation of cars based on specific requirements such as brand, model, colour, price, engine size, location of the shop or the number of doors. Figure 3.1. illustrates the user interface for the customised searching system in the DMS.

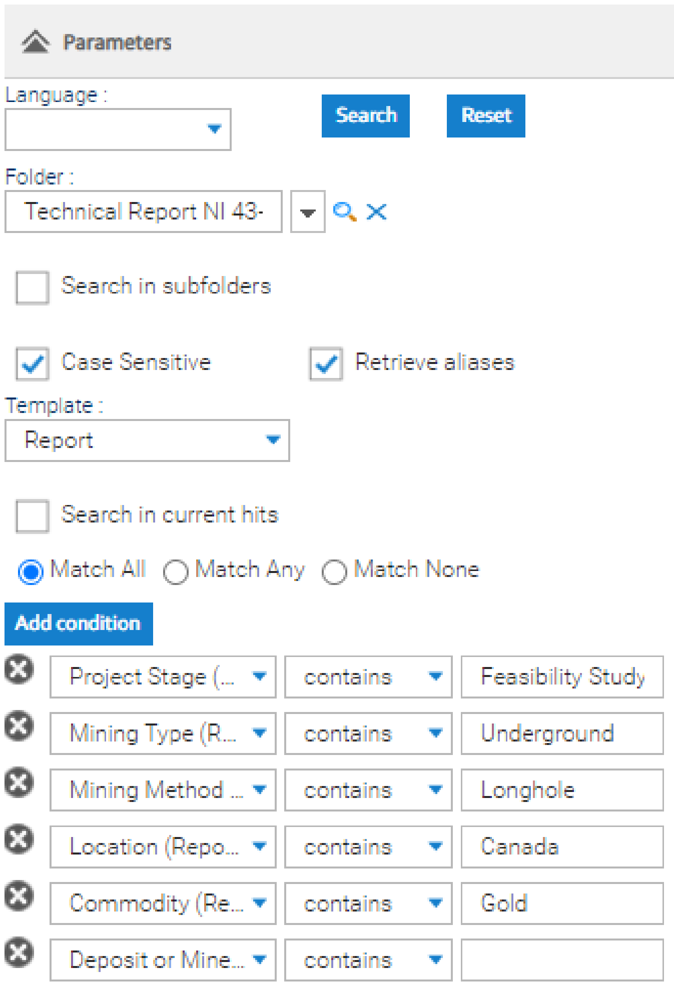


Figure 3.1: The user interface of the searching system in the DMS: allowing users to customise search by parameters such as Project Stage, Mining Type, Mining Method, Location, Commodity, Deposit or Mine

The searching system in the DMS was created using essential keywords and tags from the technical reports, such as the name of the mining company, the location of the project, the technical report issue date, commodity type, mining type, mining method, mining project stage (early exploration, advanced exploration, feasibility study, development, and production) among others. Using the searching system, a user can customise searches for relevant technical reports and quickly filter (and mine/extract) relevant attributes from the technical reports. Therefore, helping users deal with information overload by allowing easy and quick access to relevant information, which is one of the aims of recommendation systems. Through the searching system, the technical reports can be accessed using attributes such as project stage, company name, mining type, mining method, project location, name of the mine (or deposit), report issue date, report issuer, and commodity type.

Once we stored the technical reports in the DMS, the data analysis process proceeded. The data analysis process aims to analyse the quality of content in the technical reports, eliminating irrelevant and selecting the most relevant reports for the study. In a total of 1315 technical reports collected from *SEDAR*, about 113 were not useful for the study, thus, discarded. The useful data or technical reports are categorised and clustered into five classes based on project stages [1]: early exploration, advanced exploration, feasibility study, development, and production. For this study, the relevant reports are those categorised in the feasibility study, development, and production stages; thus, they are used as a base to mine (extract) relevant attributes for creating the input datasets for the study. Exploration (early and advanced) is the earliest stage of a mining project aimed at defining a potential mineral deposit. Feasibility studies are done after a potential mineral deposit to determine whether a project is viable enough to proceed (to be abandoned or wait until the commodity's price). Once a project is viable enough, the development stage and production proceed. The development stage aims to construct necessary infrastructures and roads to access the orebody and proceed with production. Finally, the production stage aims to recover mineral resources by applying different mining methods among surface or underground methods. Figure 3.2 shows the results of the data analysis process, illustrating the proportions of total collected data, not useful and useful data clustered into early exploration, advanced exploration, feasibility study, development, and production stages.

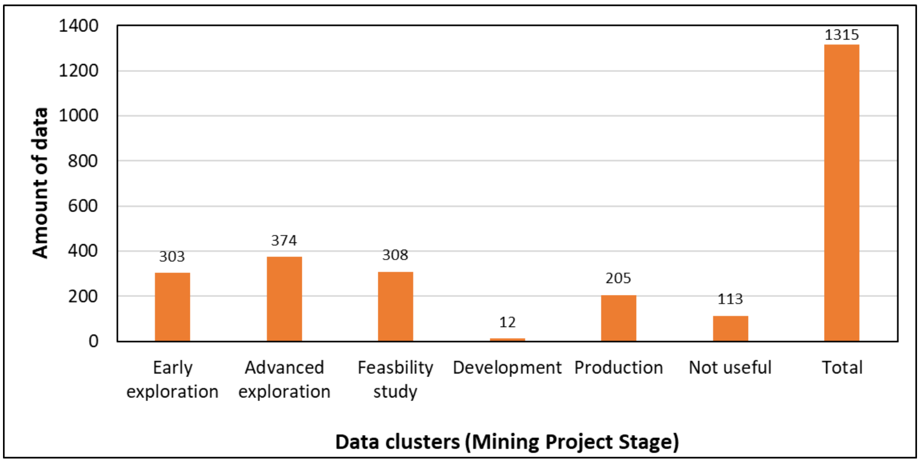


Figure 3.2: Results of data analysis, showing the proportion of the total collected data, not useful and the useful clustered

3.2 Creating Input Datasets, Data Cleaning and Data Validation

3.2.1 Creating input datasets

The input dataset for this study is created based on the results from chapter 2. Chapter 2 applied the Entropy method to estimate the relative importance of twenty factors influencing the MMS process and identify the most relevant factors. The Entropy method estimated the relative importance by calculating factors' objective weights without the direct involvement of decision-makers (judgment or opinion). Then based on the objective weights, ore strength, host-rock strength, orebody thickness, orebody shape, orebody dip, ore uniformity, mining costs, and the dilution of the mining methods were selected as the most relevant factors in MMS. These factors will then be used as the main variables in the input datasets in this chapter. To recap, Table 3.1 illustrates the results of the Entropy method where we have the twenty factors' entropy, diversity, and objective weights. Table 3.1 lists factors according to the objective weights in descent order; the first seven factors are selected as the most relevant in MMS.

Initially, all eight factors are used as main variables in the input dataset. The dataset is created by mining (extracting) information about the eight factors from the mining projects' technical reports in the DMS. The created initial dataset describes data from sixty-one projects and eight variables (ore strength, host-rock strength, orebody thickness, orebody shape, orebody dip, ore uniformity, mining costs, and the dilution) and the mining methods considered/selected in each project. However, this initial input dataset is very sparse because of the lack of information about the required factors in some project's technical reports. In other words, the initial input dataset has a lot of missing attributes, originating an incomplete dataset. Therefore, the “data sparsity problem” is one of the limitations of this study. Since data is the foundation of machine learning (ML) models, it is crucial to have clean and consistent data to train the models for artificial intelligence (AI) systems. Missing values in an input dataset can cause bias or noise to the results of the models and, in most cases, reduce the accuracy of the models. In this case, it is required to perform data cleaning to handle the missing values in the input dataset.

Table 3.1: Results from the Entropy methods showing Entropy, Diversity values and the objective weights of the factors: listed in descent order of the objective weights

Influential factor	Entropy	Diversity	Weights
Ore strength	0.895	0.105	**0.132**
Host-rock strength	0.909	0.091	**0.115**
Thickness	0.917	0.083	**0.104**
Shape	0.920	0.080	**0.100**
Dip	0.943	0.057	**0.072**
Ore uniformity	0.946	0.054	**0.068**
Mining cost	0.952	0.048	**0.061**
Dilution	0.955	0.045	**0.057**
Ore grade	0.961	0.039	0.048
Selectivity	0.968	0.032	0.040
Recovery	0.976	0.024	0.030
Productivity	0.977	0.023	0.029
Capital investment	0.981	0.019	0.024
Depth capacity	0.982	0.018	0.023
Flexibility	0.982	0.018	0.022
Depth	0.985	0.015	0.019
Production rate	0.985	0.015	0.019
Development rate	0.985	0.015	0.019
Health and safety	0.991	0.009	0.011
Stability of openings	0.995	0.005	0.007

3.2.2 Data cleaning

There are two most common ways of handling missing values in data mining and ML problems: removing (deleting) or imputing missing values [64]–[66]. Deleting the missing values involves removing variables or rows with missing values. While imputing involves replacing the missing values with arbitrary values, with mean or median values (for numeric values) of the column, replacing with mode (for categorical values) or even performing interpolation [67].

In this study, the data cleaning process consists of deleting or eliminating noisy or biased data or variables from the dataset. The variable "mining costs" is biased owing to the differences in currencies and exchange rates in each project: projects are located in different countries. Therefore, the "mining costs" variable was removed from the initial input dataset. Furthermore, missing values were handled by removing (or deleting) rows and variables with missing values, and variables such as "dilution" and "ore uniformity" were eliminated for having a lot of missing values.

Deleting missing values from the dataset implies reducing the size of the dataset. Therefore, from a noisy and biased dataset with sixty-one rows/projects and eight variables, the clean input dataset has thirty rows/projects and five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip). The clean dataset comprises five underground mining methods, with uneven distribution of projects among the mining methods. Figure 3.3 shows the distribution of the projects in each mining method in the dataset, with more projects in the sublevel stoping mining method.

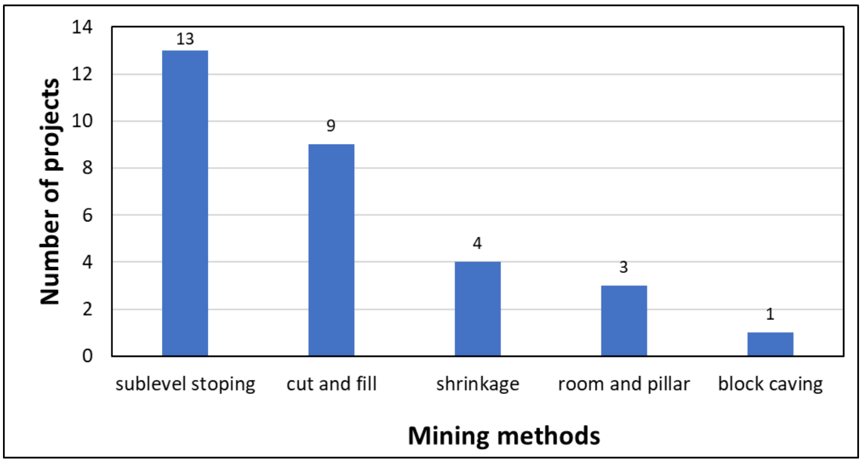


Figure 3.3: Distribution of the dataset after data cleaning, showing the number of projects in each mining method

This study's strength factors (of ore and host-rock) are mainly based on the rock mass rating (RMR) system. The RMR system is one of the most accessible geotechnical parameters in the mining projects' historical data (technical reports); hence, it was chosen as the main parameter to describe ore and host-rock strength. The orebody thickness, shape and dip factors are described based on Nichola's approach [5] and the UBC MMS tool [4].

3.2.3 Data validation

Since the whole dataset is based on information provided by mining projects, it would be too risky to solely trust the data because every company operates with different goals, rules, and regulations. For this reason, it was necessary to verify the integrity of the data or validate reliability of the data: verify how trustworthy the mining projects' procedures and decisions are. Then use the mining projects' data to provide future recommendations in the proposed AI-MMRS. The UBC-MMS system [4] was applied to validate the reliability of the mining methods selected for each project. In the UBC-MMS tool, we use the inputs from the project's technical reports to verify if the mining methods selected are appropriate for the orebody conditions. As described in chapter 1, the UBC-MMS tool is a quantitative MMS system and one of the most commonly used for MMS and as a base for studies in MMS. We used the "Excel and Visual Basic Implementation (version created by Jeff Breadner, 1999) of the UBC-MMS tool. Figure 3.4 shows the required inputs in the UBC MMS tool. As depicted in Figure 3.4, the UBC-MMS tool requires eleven variables as inputs. The UBC-MMS outputs ten mining methods with respective ranks in descent order, where the best mining methods are those with the highest ranks [4].

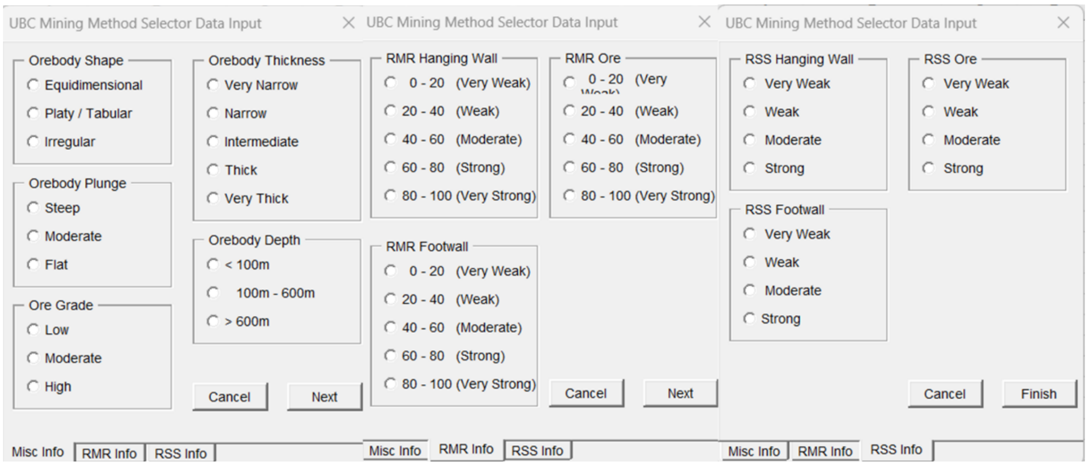


Figure 3.4: Showing the required inputs in the UBC-MMS system (in the Excell and Visual Basic Implementation tool)

We observed that in most technical reports, during the early stages of mining project development is difficult to get detailed information on the orebody characteristics, especially the geotechnical parameters required as inputs in the UBC-MMS tool. The lack of information about some required input variables for the UBC-MMS tool limits the validation (or verification) of the data of all thirty projects. Therefore, we used the literature review as a secondary data source by collecting information about underground mining case studies, mostly from the academic thesis and research papers. In addition to that, more reports were collected from the *SEDAR* database.

After additional data collection and validation using the UBC-MMS tool, the second input dataset comprises thirty-three projects (or case studies), five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip) and seven mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping). The seven mining methods include the three main classes of underground mining methods: unsupported, supported and caving, excluding the obsolete and extinct supported methods such as square set and stull stoping [1]. Sublevel stoping [25], shrinkage [24] and room and pillar [23] represent the unsupported methods. Cut and fill [26] is the only commonly applied supported mining method. The caving methods are represented by longwall [27] block caving [29] and sublevel caving [28]. Therefore, in terms of underground mining methods, this validated input dataset (depicted in Figure 3.5) is more complete than the non-validated input dataset (depicted in Figure 3.3). Figure 3.5 shows the distribution of projects (case studies) per mining method in the validated dataset.

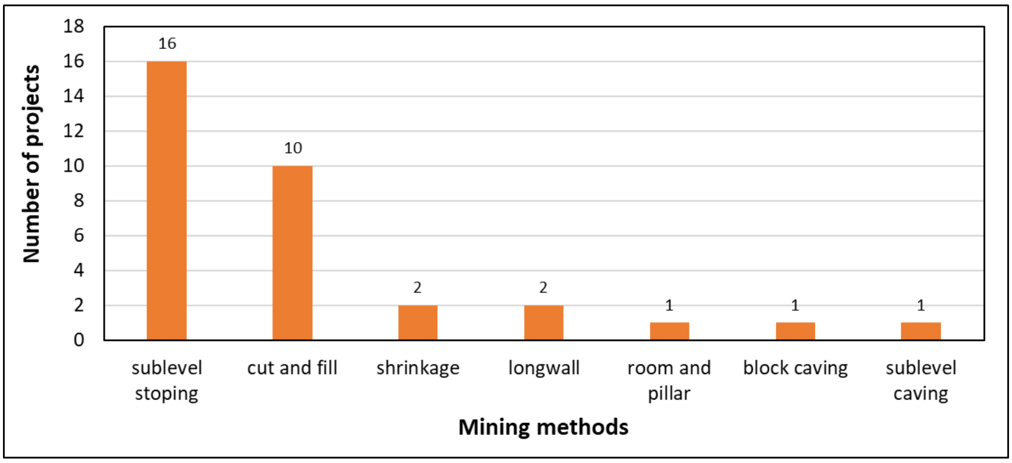


Figure 3.5: Distribution of the dataset after cleaning and validation using the UBC-MMS tool, showing the number of projects in each mining method

3.3 Review of the Most Preferred Underground Mining Methods in the 2000s

One of the objectives of this study on developing an AI-MMRS is to provide a review of the most applied underground mining methods in the late 2000s. The review is based on the study's database, i.e., on data collected from the *SEDAR* database. The technical reports in our dataset are dated from 2000 to 2022, with projects located in different countries or places of the world.

3.3.1 Most preferred underground mining methods

To show the statistics of the most common mining methods, we use the initial sparse dataset before the data cleaning process because it is the largest dataset. The initial sparse dataset comprises sixty-one projects, most located in Canada (~54%) and the minority in Europe. Figure 3.6 shows the distribution of the projects by location; the legend represents the number of projects in each country: 1, 3, 2, 4, and 7.

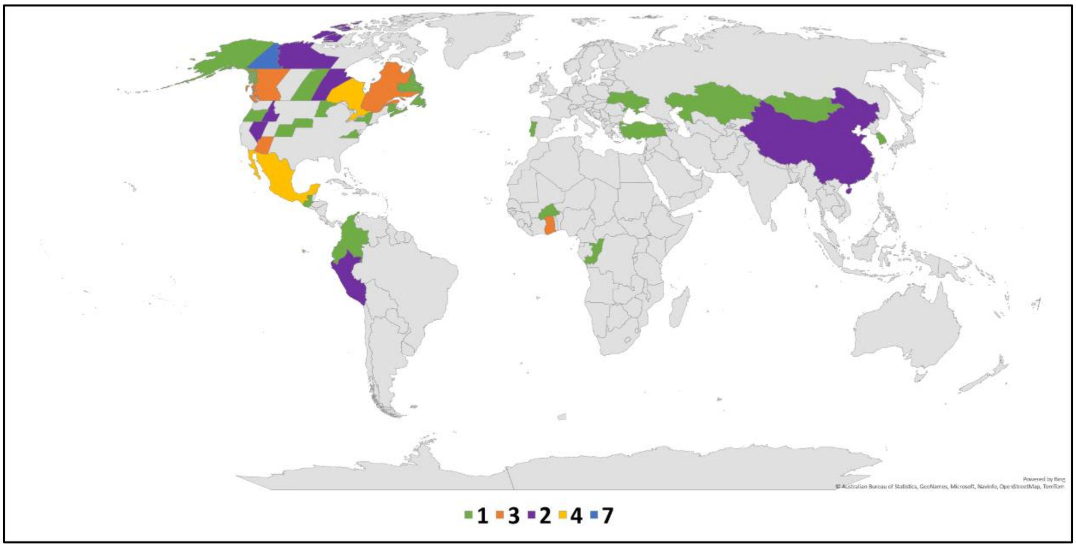


Figure 3.6: Distribution of the mining projects by location: the legend shows the number of projects in each country (identified by colours: 1, 3, 2, 4, and 7)

Figure 3.7 shows the statistics of the most preferred underground mining methods among the sixty-one projects. From our results, statistics show that sublevel stoping is the top preferred mining method. This fact may be attributed to the sublevel stoping [25] mining method having more variations, making it more versatile to implement in different conditions of the orebody. The most preferred variation of sublevel stoping in the projects is the longhole stoping. Furthermore, sublevel stoping is a large-scale method with low mining costs, high productivity, and mechanisation despite the low selectivity [1]. Cut and fill [26] is the second most preferred underground mining method, with drift and fill as the common variation. Cut and fill [26] is usually more selective, with the highest recovery rate than sublevel stoping, but cut and fill demands higher mining costs and slightly lower scale than sublevel stoping and requires backfilling material [1]. Room and pillar [23] and shrinkage [24] come in the third position. Room and pillar are highly limited to certain deposit conditions, and shrinkage can be labour-intensive with limitations to mechanisation and lower production rate [1]. Lastly, caving mining methods, including longwall [27] and block caving [29], are the least preferred. Like room and pillar, longwall can be very limited to certain deposit conditions despite the large production rate and high mechanizability [1]. Despite the low mining costs and high productivity and production rate, block caving and sublevel caving [28], [29] are the least preferred. The less preference for caving mining methods can be attributed to the fact that they destroy the surface and natural environment (causing surface subsidence), making them less environmentally friendly.

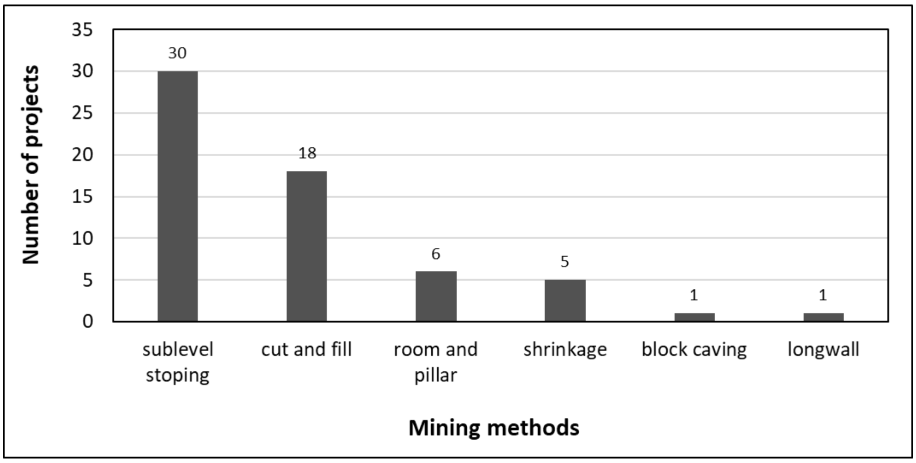


Figure 3.7: Distribution of mining projects by mining methods in the initial sparse dataset: showing sublevel stoping as the most preferred mining method

3.3.2 Orebody conditions of the preferred mining methods

This section provides an update on the orebody conditions of the most preferred mining methods in the projects. This review compares literature and practical choice by the mining projects in the study database (technical reports). To provide a comprehensive discussion, we will use the clean datasets pre and post-validation datasets depicted in Figures 3.3 and 3.4:

Sublevel Stoping:

This method is one of the unsupported mining methods, traditionally requiring no or less artificial support system. According to the literature [1], [25], this method is applied to deposits with moderate to strong ore and strong to very strong (sometimes moderate) host-rock. This method is mainly preferred for steep orebodies (dip higher than 50^o), but it can sometimes be applied in deposits with an intermediate dip (at least 40^o). In terms of thickness, this method can be a bit versatile: preferable for intermediate to thick (more than 6m thickness), but it can also be implemented in narrow deposits (less than 6m, more than 3m). Deposits with tabular and regular shapes are the most suitable for this method.

From the study's database (i.e., practical implementation), this method is preferred for deposits with ore strength following the literature, mostly moderate to strong (and very strong), with few cases with weak ore. Similar to ore strength, this method is preferred for deposits with moderate to strong host-rock, with few cases with weak and very strong host-rock. In terms of shape and dip, it follows the literature, mostly chosen for tabular orebodies with intermediate, mostly steep dip. However, there are rare cases where this method is considered for massive and flat dip (almost 20^o dip) deposits. In most cases, this method is considered for orebodies with narrow thicknesses and few cases for intermediate orebodies (rare cases with thick orebodies), which is slightly contrary to the literature. This situation can be attributed to the fact that this method has many variations that are benefited from technological advancement and easy mechanisation.

Shrinkage stoping:

Shrinkage stoping is also classified as an unsupported method requiring less artificial support. The shrinkage stoping is another unsupported mining method similar to sublevel stoping, room and pillar. Similar to sublevel stoping, this method [1], [24] is mostly preferred for steep orebodies (dip higher than 50^o), but it can sometimes be applied to deposits with intermediate dip (at least 40^o). The method favours deposits with strong ore and a moderate to strong host-rock. This method can be applied for deposits with any shape but is mostly preferred for tabular deposits with narrow to intermediate thicknesses (3m to 30m) and sometimes very narrow thicknesses (less than 3m).

Observations from the study database reveal that decisions from the mining projects follow the literature regarding host-rock strength, orebody thickness, shape, and dip. This method is considered for deposits with moderate to strong host-rock (rare cases with very strong), in most cases, for deposits with a steep dip and narrow thickness (rare cases with very narrow). The shape of the orebodies is mostly tabular, with few cases of irregular shape. The ore strength of the deposits does not follow the literature, as this method is mostly considered for deposits with weak to moderate ore strength (rare cases for strong ore).

Room and pillar stoping:

The room and pillar is another unsupported mining method similar to sublevel stoping and shrinkage stoping. According to the literature [1], [23], this method is usually suitable for deposits with moderate to strong host-rock in order to maintain the pillar and stability of openings. The ore strength usually does not limit the application of the method. It is mostly preferred for tabular deposits with a flat dip (less than 15°), in some cases, for deposits with an intermediate dip (up to 45°). This method is preferred for orebodies with very narrow to narrow thicknesses (less than 5m); in some cases, it is implemented for orebodies with an intermediate thickness (up to 30m).

Observations from the study's database suggest that the choice for this mining method in mining projects follows the literature in terms of shape, thickness, dip, and ore strength. This method is preferred for tabular orebodies, mostly narrow thickness (few cases with very narrow thickness) and mostly flat dip (few cases with intermediate dip) and variable ore strength (very weak to strong). However, contrary to the literature, this method is considered in deposits with weak to moderate host-rock strength. This situation is likely because of technological advancements and the possibility of using better artificial support systems.

Cut and fill stoping:

Cut and fill stoping is the only supported mining method that is still common nowadays. This method demands a large extent of artificial support systems. This method [1], [26] is versatile, easily adaptable, and flexible. Because artificial filling is necessary to maintain the stability of openings, this method is preferred for deposits with very weak to weak host-rock and moderate to strong ore. It can be applied for deposits with irregular and tabular shapes, with dip ranging from intermediate to steep and sometimes flat dip.

This method can be applied for various thicknesses, from narrow (sometimes very narrow) to intermediate to thick.

From the study's database, this method is mainly considered for deposits with ore strength, thickness, shape, and dip following the literature. However, the host-rock strength slightly differs from the literature review, as this method is mostly considered for deposits with moderate to strong host-rock (few cases for weak host-rock).

Longwall

Longwall is one of the caving mining methods with controlled, induced, or massive caving of the ore and/or host-rock. According to the literature [1], [27] this method is applied in tabular, narrow (and very narrow) and flat deposits (less than 12^o), especially coal. It can be suitable for deposits with any ore strength and weak to moderate host-rock strength.

From the study's database, this method is considered for most orebody conditions under the literature. The thickness of the orebody is mostly narrow (in some cases very narrow), with a tabular shape and weak to moderate ore and host-rock strength. However, it is sometimes considered in cases where the orebody has an intermediate dip (more than 12^o) and strong host rock.

Block caving and sublevel caving

These two methods are applied in similar orebody conditions. The main difference is that both the ore and the host-rock are involved in the caving process in block caving, while in sublevel caving, only the host-rock is involved in caving. According to the literature [1], [28], [29], these two methods are applied for massive deposits, but sublevel caving can be suitable for tabular deposits. In terms of thickness, these methods are suitable for thick to very thick orebodies; however, sublevel caving can be applied in deposits with an intermediate thickness (usually more than 6m or 10m). Both methods are more suitable for deposits with steep dips but can be applied in flat dips if the deposits are thick. Sublevel caving is suitable for deposits with moderate to strong ore and weak to strong host-rock. Block caving is more suitable for deposits with weak to moderate host-rock and ore (sometimes in strong ore).

Observations from the study's database indicate that practices from the mining projects are under the literature. In both mining methods, the conditions of the orebody, ore and host-rock strength, orebody thickness, shape and dip are in accordance with the literature review, as block caving is considered in thick, massive, and flat deposits with moderate to strong ore and moderate host-rock. Similarly, sublevel caving is considered for intermediate to thick, tabular, and steep deposits with strong ore and host-rock.

3.4 Summary of the Study's Data

This chapter provided an overview of the study data, from data collection to creating input datasets that will be used for training the models to develop the AI-based mining methods recommendation system (AI-MMRS). The study's data is mainly based on mining projects' historical data (i.e., technical reports: NI 43-101) collected from the *SEDAR* database. The secondary data source is a literature review from an academic thesis and research papers on mining methods selection (MMS). The technical reports collected from the *SEDAR* database (around 1,315) were stored in document management software (DMS) for management and analysis. In the DMS, an appropriate searching system was created using important attributes from the technical reports. This searching system is also an offline knowledge-based recommendation system used to customise searches for relevant technical reports and quickly filter (and extract) relevant attributes from the technical reports.

The input datasets for this study were created based on the results from chapter 2, in which the Entropy method was applied to estimate the relative importance of twenty factors influencing the MMS process to identify the most relevant factors then. The main variables in the input datasets were the most relevant factors, namely ore strength, host-rock strength, orebody thickness, shape and dip. Two input datasets were created for the study. The first comprises thirty projects (or case studies) from *SEDAR* with five underground mining methods (block caving, cut and fill, room and pillar, shrinkage, and sublevel stoping). The second comprises thirty-three projects (or case studies) from *SEDAR* and the literature review, with seven types of underground mining methods (sublevel stoping, shrinkage, room and pillar, cut and fill, longwall, block caving, and sublevel caving). The second dataset is more complete than the first dataset because it contains all main underground mining methods except those in extinction, such as square-set and stull stoping. Furthermore, the data in this dataset was validated using the UBC-

MMS tool. The first dataset will be used in Chapter 4, and the second will be used in Chapters 3 and 5.

This chapter also reviewed the most preferred underground mining methods in the late 2000s based on the database from *SEDAR*. According to the study's database, most projects' preferred mining method is the sublevel stoping, followed by the cut and fill, room and pillar, shrinkage, and longwall. Caving mining methods such as block caving and sublevel caving seem to be the least preferred lately, probably due to the environmental impacts of destroying the surface. Square set stoping and stull stoping methods seem to be in extinction because they were not observed in the database. In terms of the orebody conditions of the preferred mining methods in the database, most mining methods are selected for orebody conditions following the existing literature. However, there is a noticeable change in conditions such as thickness, ore strength, host-rock strength and dip for sublevel stoping, shrinkage, room and pillar, cut and fill and longwall. This change can be associated with technological advancement, enabling easier and more flexible mechanisation and better support systems, thus improving the versatility of the mining methods. This way, reviewing the impact of the scientific and technological advancement associated with the change of orebody conditions in which underground mining methods are implemented. Thus, updating the literature on the MMS discipline.

4 KNN-Cosine Similarity for Implementing Memory-Based Collaborative Filtering Approach in Mining Methods Selection

This chapter investigates the applicability of the memory-based collaborative filtering (CF) approach for predicting and recommending a set of top-N underground mining methods into the proposed AI-based mining methods recommendation system (AI-MMRS). The KNN-cosine similarity algorithm is used for computing similarities among the projects for the predictions using one of the datasets created in chapter 3. The dataset comprises thirty-three mining projects' historical data described by orebody characteristics and the selected mining methods to recover the deposits.

4.1 Why Collaborative Filtering Approach for the AI-MMRS

Artificial intelligence (AI) systems rely on the availability of historical data, i.e., it is necessary to have large datasets to effectively train machine learning (ML) models for AI systems. Recommendation systems [18], [19], [35], [36] are part of AI systems that provide recommendations to users about products or items they might like. The purpose of recommendation systems is very similar to the information retrieval systems [68], [69] like the google search engine. Both recommendation and information retrieval systems are aimed at helping users deal with information overload by filtering personalised and useful information. Recommendation systems rely on users' historical data in order to predict users' future interests. User historical information refers to user purchase history and the ratings the users give to purchased items. Collaborative filtering (CF) is one of the most common types of recommendation systems. This type of system generates recommendations by evaluating the similarities between different users, i.e., based on the approach that users interested in similar items in the past might have similar tastes in the future. CF can be implemented in two approaches, memory-based and model-based. Model-based uses machine learning algorithms to train models to predict users' ratings of unrated items and generate recommendations. There is no need for training or optimisation in the memory-based, also known as neighbourhood-based CF algorithms. The memory-based approach was among the earliest CF filtering algorithms and is easy to implement, understand and interpret. In the memory-based approach, unknown users or item ratings are predicted based on their nearest neighbours (most similar users). This approach relies on similarity measures to find a group of neighbours or similar users or items. The k-nearest neighbours (KNN) algorithm is commonly used for implementing memory-based CF to compute similarities among users and items.

As stated in the introduction, this study investigates the possibility of incorporating artificial intelligence (AI) to explore available mining projects database to develop a system that can aid decision-making when planning a mining project. As such, the proposed system will attempt to recommend the most appropriate mining methods by learning from previous mining projects' procedures; thus, the proposed system is directly linked to collaborative filtering problems. This chapter investigates the applicability of the memory-based collaborative filtering approach to predicting and recommending underground mining methods. Using the proposed memory-based collaborative filtering approach, we will build a model to predict and recommend not just one but a set of top-N underground mining methods. The KNN-cosine similarity algorithm is used to compute similarities among the projects. Thereafter, the weighted sum method is used to predict the ratings of the mining methods for generating the top-N recommendations. The dataset for this chapter is created in Chapter 3 (described in Figure 3.5), comprising thirty-three projects, five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip), and seven underground mining methods (sublevel stoping, shrinkage, room and pillar, cut and fill, longwall, block caving, and sublevel caving). The memory-based approach is proposed as one of the algorithms to be implemented in the proposed AI-MMRS for generating top-N recommendations of the most appropriate mining methods by analysing the similarities among the projects regarding orebody deposit characteristics. In other words, recommend mining methods based on practices from other similar deposits or projects.

4.2 Memory-Based Collaborative Filtering Recommendation System

Recommendation systems are aimed at helping users deal with information overload by making personalised suggestions or suggesting the most relevant/popular items. In order to make personalised suggestions, recommendation systems [21], [43] collect different types of data about the users and the items. In this study, users are defined as the objects receiving the recommendations and items as the recommended objects. Therefore, datasets for recommendation systems are composed of three main objects: users, items and the interaction between users and items. The recommendations systems collect various information about the objects, for example, the user's attributes/features (age, location, gender, sex); the item's attributes/features (description of the items: colour, price); and the interaction between users and items or the evaluation/feedback from users about the items (implicitly or explicitly ratings). Implicit feedback can be collected through purchase records, time logs, cart history, wishlist, and web hyperlinks; explicit feedback is collected through rating scores (numerical, ordinal, binary or unary ratings). Collaborative filtering (CF) [20], [70], [71] recommendation systems make personalised suggestions mainly based on the user-item interaction matrix that contains ratings that different users give to evaluate items. CF make recommendations to a target user based on the preferences/ratings of other users considered similar to the target user. A target user is a user to whom the recommendations are made. Recommendations can be made by predicting the ratings the target user will like to give to unrated items and/or by providing a list of top-N items that the target user will like the most [72]–[74]. In this study, we use the terms unrated, missing ratings or unknown ratings interchangeably; and the terms active and target users interchangeably. CF recommendation systems can be developed through two approaches, model-based and memory-based. In the model-based [75]–[77] approach, machine learning (ML) algorithms are used to train models that automatically learn user-item interaction patterns. Memory-based, mostly known as the neighbourhood-based approach [76], [78]–[80], is simple, straightforward, and easy to understand and interpret the results. In this approach, there is no need for training a model; the algorithms utilise the entire user-item matrix to generate predictions directly. This approach is based on neighbourhood methods to compute similarities among users and items and generate recommendations. Memory-based CF can be further divided into user-based (or user-user) and item-based (or item-item) [35], [72]–[74], [78]–[82].

Let **X**, in Figure 4.1, be a user-item interaction matrix:

n-items

m-users

	I1	I2	I3	I4	I5	I6
U1	1	5	3	?	?	5
U2	2	?	4	?	5	1
U3	5	2	3	?	1	?
U4	?	5	?	4	?	5
U5	?	1	5	3	?	2

Figure 4.1: **X** user-item interaction matrix composed of *m-users* and *n-items* with ratings ranging from 1 to 5, "?" unknown or missing rating

- User-based: find a set of most similar users to the active user 1, U1: similar users will rate the same items similarly. Then, use the ratings from a set of similar users to predict ratings that a target U1 would give to unrated items 1 and 2(I1 and I2). Finally, recommend the top-N items with the highest predicted rating to the U1.
- Item-based: To predict ratings that the active user, U1, would give to unrated items 4 and 5(I4 and I5), first find a set of most similar items to I4 and I5 based on items that U1 has rated: similar items are rated similarly by the active user U1 and similar users. Then, predict ratings that the U1 would give to target items I4 and I5 based on the ratings that U1 gave to similar items. Finally, recommend a list of top-N most relevant items to U1: the higher the predicted rating, the most relevant the item is.

The main difference between item-based and user-based is in user-based recommendations are made based on other users' ratings. In contrast, item-based recommendations are based on ratings from the same user [78]–[80]. In matrix **X** in Figure 4.1, user-based, the similarity is measured between the rows, while item-based similarity is measured between the columns. In this chapter, we employ the user-based CF approach; thus, the following steps will be explained mainly based on the user-based approach.

There are two main steps to building a memory-based CF recommendation system: 1) measure similarity and 2) predictions and recommendations:

- <u>Step 1: Measuring similarity</u>

The first and key step to building a memory-based recommendation system is to find similarities between users and items. Different methods can be applied to compute similarities; here, we show Pearson Correlation and Cosine similarity [78]–[80], [83]:

<u>Pearson correlation or correlation coefficient</u> measures the strength of the linear relationship between two variables. The Pearson coefficient (in a user-item matrix) is calculated between a target user and all other users. The similarity measure by Pearson correlation is shown in Equation (4.1):

$$sim(a,b) = r_{a,b} = \frac{\sum_i (r_{ai} - \overline{r_a})(r_b - \overline{r_b})}{\sqrt{\sum (r_{ai} - \overline{r_a})^2}\sqrt{\sum (r_{bi} - \overline{r_b})^2}} \tag{4.1}$$

<u>Cosine similarity</u> measures similarities between two vectors, defined as the cosine of the angle between the two vectors. Equation (4.2) shows the computation of cosine similarity to measure the similarity between two users, *a* and *b*:

$$sim(a,b) = \cos(\theta) = \frac{r_a . r_b}{\|r_a\| \|r_b\|} = \sum_i \frac{r_{ai} . r_{bi}}{\sqrt{\sum_i {r_{ai}}^2}\sqrt{\sum_i {r_{bi}}^2}} \tag{4.2}$$

- <u>Step 2: Predictions and Recommendations:</u>

Once we find a set of most similar users, using one of the methods in Equations (4.1) and (4.2), the next step is to use this set of most similar users to predict missing/unknown ratings for an active/target user. Different methods are used to predict the unknown ratings; here, we will show the weighted sum method which is applied in this study:

<u>Weighted sum method:</u>

This method predicts the ratings of unrated items for a target user by computing the weighted average of the ratings that the most similar users gave to those unrated items. Equation (4.3) shows the weighted sum method used to predict the ratings:

$$r_{ui} = \frac{\sum_{i \in users} sim(u,i) * r_{ui}}{\sum_{i \in users} |sim(u,i)|} \tag{4.3}$$

Apart from predicting users' ratings of unrated items, recommendation systems are also tasked with recommending good items to users [84]. Commonly, recommendation systems recommend the top-N items [73], [74], [85]–[90] to users: items are those with the highest predicted ratings.

4.3 KNN-Cosine Similarity for Implementing Memory-based Collaborative Filtering Approach in Mining Methods Selection

In this study, we use the k-nearest neighbours (KNN) with cosine similarity to measure the similarity among the projects in the dataset. KNN is one of the low-level supervised classification machine learning (ML) algorithms [91], [92]. Supervised ML learns from labelled datasets, while unsupervised ML learns from unlabelled data. Labelled datasets refer to datasets in which each sample/datapoint is tagged to a label/class; on the contrary, unlabelled data do not have a tag/label. In unsupervised learning, algorithms use unlabelled datasets to discover information or patterns. In the memory-based CF approach, the KNN is implemented as an unsupervised learner that is incorporated to search for nearest neighbours [93]–[96]. Therefore, KNN is applied as a pre-processing stage for computing similarities before ratings prediction and recommendations. The steps to generating top-N recommendations using KNN are as follows:

- Step1: identify target/active user
- Step 2: define the k-number of neighbours for predictions
- Step 3: use KNN to find the k-nearest neighbours to the target user
- Step 4: find items with missing ratings (or unrated items) for the target user
- Step 5: predict the missing ratings for the target user
- Step 6: recommend top-N items: items with the highest predicted ratings

In order to incorporate user-based collaborative filtering (CF) techniques in underground mining methods selection (MMS), it is required to prepare an appropriate dataset. In the CF recommendation system, the dataset is usually filled with ratings given by the users to evaluate an item, as shown in Figure 4.1. On the other hand, the dataset used in this study is more suitable for classical ML [32] classification problems. This study's dataset contains datapoints denominated as projects described by their attributes/dependent variables (the five input variables) and class labels (the seven underground mining methods). Therefore, one of the steps in data pre-processing is to transform the input dataset to an appropriate format for CF problems to have a dataset similar to the user-item interaction matrix, with users represented in the rows and items in the column, as shown in Figure 4.1. Figure 4.2 shows the workflow of the proposed methodology for practical experiments to evaluate the applicability of the memory-based collaborative filtering (CF) approach for predicting and recommending top-N recommendations of underground mining methods.

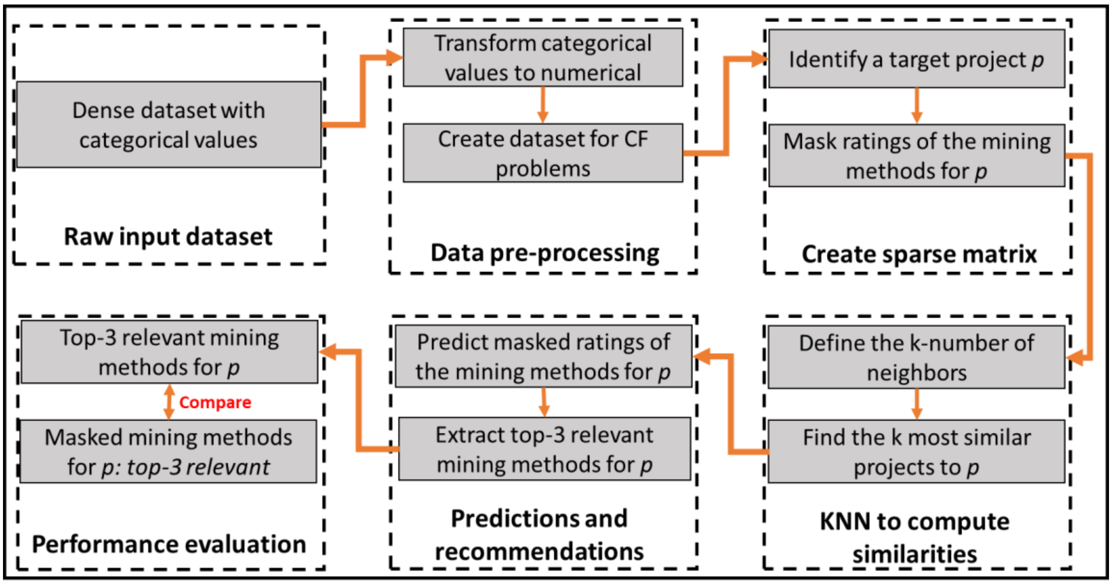


Figure 4.2: Workflow of the proposed methodology for practical experiments: the memory-based collaborative filtering approach for predicting and recommending top-N mining methods

Recommendation systems can be evaluated both online and offline [84], [97]–[100]. Offline evaluations are usually performed during the design phase to test and filter different algorithms. In this study, we perform an offline evaluation. To evaluate algorithms offline is necessary to simulate users' behaviour in online recommendation systems [84], [97]–[100]. Offline evaluation experiments are done using a user-item interaction matrix which contains users' preferences for the items (ratings or likes). The user behaviour can be simulated by first hiding some users' ratings in the user-item interaction matrix; then using the algorithms to predict the hidden ratings. Finally, the hidden ratings are used as ground truth data to evaluate the quality of recommendations given by the algorithm [84], [99], [101]. Selecting the ratings to be hidden highly depends on the practical situation in which the system will be employed.

In the practical application of the proposed AI-MMRS, we assume that the target project will provide the system with all five input variables/attributes: ore strength, host-rock strength, orebody thickness, shape, and dip. These input variables/attributes are used to search for a set of nearest projects in the dataset, i.e., for projects with similar attributes to the target project. After that, the set of nearest projects is used to generate the recommendations of the top-N most appropriate mining methods for the target project. In the experiments, we simulate the practical application of the proposed methodology in the AI-MMRS as follows. For a target project, we mask/hide the known ratings of all seven mining methods in the dataset. Thus, creating a sparse dataset for the target project with seven missing ratings. Then, the proposed algorithm is used to predict the ratings of the mining methods for the target project. A list of top-N most relevant mining methods is provided based on the predicted ratings. The hidden/masked ratings are kept as ground truth data and used for evaluating the accuracy and quality of the top-N recommendations provided by the proposed algorithm.

4.4 Practical Experiments on the Implementation of the Memory-based Collaborative Filtering Approach for Mining Methods Selection

4.4.1 Dataset

This chapter is based on one of the datasets created in chapter 3, section 3.3, the dataset described in Figure 3.5. As previously mentioned, the dataset is suitable for ML classification problems. Table 4.1 illustrates the short representation of the input dataset used in this chapter, where the first column indicates the mining projects id (denoted as PJ001, PJ002…, PJ0033); from the second to the fifth column are the input variables/independent variables and the last column the class labels/mining method. This dataset shows historical information on thirty-three mining projects regarding the selected or considered mining methods to recover the orebody deposits. The orebody deposits in each project are described by their orebody characteristics: geotechnical properties (ore and host-rock strength) and orebody geometry (orebody thickness, shape, and dip). The mining methods tagged to each project were considered/selected based on the orebody characteristics. There are seven class labels/underground mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping).

Table 4.1: Short representation of the raw input dataset for experiments

Project	Ore strength	Host-rock strength	Thickness	Shape	Dip	Mining method
PJ001	Strong	Moderate-strong	Intermediate-thick	Tabular	Steep	Sublevel stoping
PJ002	Moderate	Moderate-strong	Very narrow-narrow	Tabular	Intermediate	Longwall
PJ003	Moderate	Strong	Narrow	Tabular	Steep	Cut and fill
PJ004	Strong	Moderate-strong	Thick	Irregular	Intermediate	Cut and fill
PJ005	Weak	Moderate	Very narrow	Irregular	Steep	Cut and fill
PJ006	Moderate	Weak-moderate	Very narrow	Tabular	Flat	Room and pillar
…	…	…	…	…	…	…
PJ033	Strong	Strong	Intermediate-thick	Tabular	Steep	Sublevel caving

4.4.2 Data pre-processing

The study's dataset in Table 4.1 is unsuitable for a memory-based CF approach because the dataset is composed of categorical values. Usually, memory-based approach algorithms perform better on numeric data; thus, the first step consists of transforming the categorical values into numerical values. First, we transform the values of the five input variables using the label encoding strategy [102]. In this study, categories of the five input variables are transformed based on their objective weights calculated using the Entropy method in chapter 2. The first step is assigning numerical values to each variable's categories (as shown in Table 4.2) using the weighing system applied in chapter 2 (refer to Figure 2.2).

Table 4.2: Label encoding to transform the qualitative values of the five input variables

Input variables	Categories	Numerical values
Ore strength	Very weak, weak, moderate, strong, very strong	1,3,5,7,9 2,4,6,8: intermediate values
Host-rock strength	Very weak, weak, moderate, strong, very strong	1,3,5,7,9 2,4,6,8: intermediate values
Thickness	Very narrow, narrow, intermediate, thick, very thick	1,3,5,7,9 2,4,6,8: intermediate values
Shape	Irregular, tabular, massive	1,3,5 2, 4: intermediate values
Dip	Flat, intermediate, steep	1,3,5 2, 4: intermediate values

Thereafter, the numerical values of the five input variables are multiplied by their objective weights obtained from the Entropy method results. Table 4.3 shows the objective weights of the five input variables (or influencing factors) and the minimum and maximum values for each input variable.

Table 4.3: Description of the independent variables transformation method showing the objective weights, minimum and maximum values

Input variables	Objective weights	Minimum value	Maximum value
Ore strength	0.132	13.2	118.8
Host-rock strength	0.115	11.5	103.5
Thickness	0.104	10.4	93.6
Shape	0.100	10.0	50
Dip	0.072	7.2	36

Once the five input variables are transformed, the next step involves transforming the class labels or the mining methods. There are seven class labels (mining methods) which are categorical values. Using the one-hot-encoding [102], [103] strategy, we transform the mining methods from class labels into items to be recommended (refer to Figure 1.1). One-hot-encoding quantifies the categorical data by producing a vector with a length equal to the number of categories of the variable, with binary values 0 and 1: 1 assigned if the datapoint belongs to the category and 0 otherwise.

For the input dataset shown in Table 4.1, we create a vector for the mining method column with seven elements corresponding to the mining methods: block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping. However, we do not use binary values as commonly used in classical one-hot encoding [102], [103]. Instead, we use a 9-point rating system (ranging from 1 to 9) for the seven mining methods. The rating system is created with the aid of the results from data validation using the UBC-MMS tool [4] in chapter 3. In chapter 3, we used the UBC-MMS tool to validate or verify the integrity of the data from the mining projects' technical reports. The UBC tool outputs ten mining methods ranked in descent order, where the most appropriate mining methods are those with the highest ranks. Therefore, with the aid UBC-MMS tool, the rantings for the mining methods are produced as described in Table 4.4.

Table 4.4: Approach for transforming the mining methods into items to be recommended showing the ratings assigned to the seven underground mining methods

Ratings	Description
9	Assigned if the mining method is the primary option for the mining project: in the technical reports
5	Assigned, if the mining methods are in the top-3 rank in the output from the UBC-MMS tool
1	Assigned if the mining method is not in the top 3 ranks from the UBC-MMS tool

As a result, we have the dataset ready for memory-based CF problems, as shown in Table 4.5. This way, we have a dataset similar to the traditional user-item interaction matrix (refer to Figure 4.1) with users in the rows as mining projects and the mining methods in columns as the items to be recommended. In the dataset in Table 4.5, OS, RS, TH, SH, and DP are abbreviations for ore strength, host-rock strength, thickness, shape, and dip, respectively. The abbreviations for the mining methods are SS (sublevel stoping), RP (room and pillar), SH (shrinkage), CF (cut and fill), LW (longwall), BC (block caving) and SC (sublevel caving). The employed 9-point rating system ensures the diversity of the proposed AI-MMRS. In the input dataset, each project has three relevant mining methods: one primary mining method with a rating of 9 and two secondary methods with a rating of 5. Therefore, the task of the proposed model is to predict the top-3 most relevant mining methods for each project: one primary and two secondary methods. This way, the proposed AI-MMRS can be used to recommend not just one mining method but a set of top-3 relevant mining methods for a target project.

Table 4.5: Short representation of the transformed input dataset to be used in the practical experiments

Project	OS	RS	TH	SH	DP	SS	RP	SH	CF	LW	BC	SC
PJ001	92.4	69.9	62.4	30	36	9	1	1	5	1	1	5
PJ002	66	69.9	20.8	30	21.6	5	1	1	5	9	1	1
PJ003	66	80.5	31.2	30	36	5	1	5	9	1	1	1
PJ004	92.4	69.9	72.8	10	21.6	1	1	1	9	1	5	5
PJ005	39.6	57.5	10.4	10	36	5	1	5	9	1	1	1
PJ006	66	46	10.4	30	7.2	1	9	1	5	5	1	1
⋮	…	…	…	…	…	…	…	…	…	…	…	…
PJ033	92.4	80.5	62.4	30	36	5	1	1	1	1	5	9

4.4.3 Creating sparse matrices

The recommendation systems aim to predict ratings that users might give to unrated items [18], [19]. Therefore, datasets in recommendation systems are usually sparse, i.e., containing several missing or unknown ratings. The input dataset for practical experiments in Table 4.5 is a dense matrix, i.e., all values are filled or known; thus, there are no missing or unknown values. In these experiments, sparse datasets are created by masking/hiding the ratings of all seven mining methods for a given target project to simulate the practical application of the proposed methodology in the AI-MMRS. The hidden/masked ratings are kept as ground truth data that will be used for evaluating the accuracy and quality of recommendations.

4.4.4 KNN-cosine similarity to compute similarities

The KNN algorithm with cosine similarity measure is implemented as an unsupervised learner to search for the nearest neighbours of the target project: to find projects with similar attributes (input variables) to the target project. In the experiments, similarities are computed with different neighbourhood sizes ranging from 2 to 10, i.e., find 2, 3, 4, …10 projects similar to the target project. Then we evaluate the prediction accuracy in each k-number of neighbours (number of projects). We used the scikit learn library for computing the KNN [104]; the parameters of the algorithm are shown in Table 4.6.

Table 4.6: Parameters of the KNN algorithm in the proposed methodology

Parameters	Description
k, number of projects	2, 3, 4, 5, 6, 7, 8, 9, 10
Metric	Cosine
Optimisation algorithm	Brute

4.4.5 Predictions and recommendations

The ratings of the mining methods are predicted using the weighted sum method shown in Equation (4.3). For each target project, we compute predictions for using a different number of projects (different neighbourhood sizes), as shown in Table 4.6.

The task of the proposed approach in the AI-MMRS is not just to predict the ratings but to provide a list of top-N most relevant mining methods for a target project. The top-

N recommendations are provided based on the predicted ratings for the seven mining methods. The priority of recommendations is given by the highest predicted rating, i.e., the higher the predicted rating of a mining method, the most relevant the method.

4.4.6 Performance evaluation metrics

There are different ways of evaluating recommendation systems depending on the goal or task of the system, whether it is to predict the ratings or to recommend good/relevant items [21], [84], [105]. Recommendation systems that recommend good items usually provide a list of top-N recommended items to users [73], [74], [86]–[90]; thereafter, users can decide if or which items are relevant to them. The evaluation metrics for recommendation systems tasked to recommend relevant items are very close to metrics used to evaluate ML classification algorithms and information retrieval systems [68]–[71], [98], [106]. The performance is evaluated by measuring the decision-making capacity of the recommendation systems: how useful/relevant the recommended items are to the users. In our experiments, we use classification metrics such as global accuracy rate, decision support metrics and ranking-based metrics to evaluate the quality of top-N recommendations.

- ***<u>Global accuracy rate (GAR):</u>***

In these experiments, we use the GAR to evaluate how well the proposed approach predicts (or classifies) the primary mining methods (with a rating of 9 in the input dataset) as the top-1 (with the highest predicted rating) most relevant method for the projects. This metric is defined as the ratio of the number of correct predictions (or classification) divided by the total number of predictions (or classification) as depicted in Equation (4.5):

$$GAR(\%) = \frac{number\ of\ correct\ classifications}{total\ number\ of\ classification} \times 100 \tag{4.5}$$

- ***<u>Decision support metrics:</u>***

Decision support metrics are based on a confusion matrix to evaluate information retrieval systems. Here the exactly predicted rating is ignored, and the recommendation system problem is transformed into a binary classification problem where one is assigned if the item is recommended in the top-N list and 0 otherwise, as shown in Table 4.7 [42], [68], [84], [97]–[100], [105]–[107].

Table 4.7: Confusion matrix used to evaluate the quality of the top-N recommended underground mining methods

	Recommended top N	**Not recommended**
Relevant	True Positive (TP)	False Negative (FN)
Not relevant	False Positive (FP)	True Negative (TN)

From the confusion matrix in Table 4.7, decision support metrics, namely Precision (@N), Recall (@N) and F1-score, can be computed as shown in Equations (4.5) to (4.7):

Precision (@N) can be defined as the ratio of the number of relevant items in the top-N list and the number of items recommended to the user. Usually, for a fixed number of items in the top-N list, we have Precision as shown in Equation (4.5).

$$Precision(@N) = \frac{TP}{TP+FP} \quad (4.5)$$

Recall (@N) is the ratio of the number of relevant items in the top-N list to the number of all relevant items. Similar to precision, recall is usually computed for a fixed number of items in the top-N list, as shown in Equation (4.6).

$$Recall(@N) = \frac{TP}{TP+FN} \quad (4.6)$$

F1-score is used to combine or balance the precision and the recall into one metric. F1-score is the harmonic mean of precision(@N) and recall(@N), as shown in Equation (4.7).

$$F1score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \quad (4.7)$$

- ***Ranking-based metric:***

Decision support metrics evaluate the quality of the recommendations regardless of the ranking order. However, if the intention is to evaluate the quality of recommendations based on the rank order of the top-N recommended items, different metrics are used. This way, it is possible to measure the recommendation system's ability to recommend items in the correct order. Mean reciprocal rank is one of the rank-based metrics used to evaluate the quality of information retrieval systems; it will be used as one of the evaluation metrics in this chapter.

Mean Reciprocal Rank (MRR) is the average reciprocal rank over users. The reciprocal rank is used to calculate the reciprocal of the rank at which the first and prioritised relevant item was retrieved [108], [109]:

$$MRR = \frac{1}{|U|}\sum_{u \epsilon U}\frac{1}{k_u} \tag{4.8}$$

Where U is the total number of users and, k_u is the rank position of the first relevant item in the top-N list for a user u.

In these experiments, we use the MRR to evaluate the quality of the top-N recommended mining methods prioritising the rank order of the primary mining methods. Here, we assume that only the primary mining method is the relevant recommendation in the top-N list; therefore, we care about the primary mining methods of each project. In other words, we intend to know if the primary mining method is included in the top-N list of recommended mining methods. Further, in which rank position the primary mining methods is placed in the top-N list. The reason for prioritising the primary mining methods is that the primary mining method is the one that defines the quality of the model. Suppose the primary method is not present in the top-N list or placed in the top-1 position. In that case, the quality of top-N recommendations provided by the model will be significantly reduced.

4.5 Experimental Results of the Implementation of the Memory-Based Collaborative Filtering Approach in Mining Methods Selection

This chapter investigates the applicability of the memory-based collaborative filtering (CF) approach for predicting and recommending underground mining methods for developing an AI-based mining methods recommendation system (AI-MMRS). This section presents the results of the practical experiments on implementing the memory-based CF approach for predicting and recommending top-N underground mining methods. The dataset used in the experiments comprises thirty-three projects described by five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip) and seven underground mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping). The memory-based approach is implemented using the KNN-cosine similarity with the weighted sum algorithm. The k-nearest neighbours (KNN) algorithm with cosine similarity measure is used to compute similarities among the projects and find a set of k-nearest projects. Then the set of nearest projects is used for predictions and recommendations using the weighted sum method.

In the proposed approach, we create an input dataset suitable for collaborative filtering problems, where each project has ratings for the seven mining methods. A rating of 9 is given to the projects' primary mining method (from the projects' technical reports), a rating of 5 to the two projects' secondary methods (found using the UBC-MMS tool), and a rating of 1 is assigned to the remaining four methods (which are not relevant to the project). Therefore, each project has three relevant mining methods (one primary and two secondaries). In the experiments, we use the approach for offline evaluation of recommendation systems [84], [97]–[100] by simulating the practical implementation of the proposed approach in the AI-MMRS as follows. We mask/hide the known ratings of all seven mining methods for a given target project. Thus, creating a sparse dataset for the target project with seven missing ratings. After that, the proposed algorithm is used to predict the masked ratings of the seven mining methods. Then, we extract a list of the top-3 most relevant mining methods is provided based on the predicted ratings (the higher the predicted rating, the more relevant the method). The hidden/masked ratings are kept as ground truth data and used for evaluating the accuracy and quality of the top-3 recommendations.

To evaluate the quality of the recommendations, we first evaluate the proposed model's capability to place each project's primary mining method in the top-1 position or as the most relevant among all three recommended methods. For that, we use evaluation metrics such as the global accuracy rate and the mean reciprocal rank. Furthermore, we evaluate the capability of the proposed model to include all three relevant mining methods of each project (one primary and two secondary methods) in the top-3 list of recommendations using the F1-score.

4.5.1 Prediction accuracy for the primary mining methods

Given the input dataset with project input variables, the proposed model is tasked to provide a list of top-3 most relevant mining methods for a target project.

To start, we use the global accuracy rate (GAR) to evaluate the model's capabilities to predict and recommend the primary mining method of each project as the top-1 most relevant method. The reason for prioritising the primary mining methods is that the primary mining method is the one that defines the quality of the model. Suppose the primary method is not present in the top-N list or placed in the top-1 position. In that case, the quality of top-N recommendations provided by the model will be significantly reduced, implying that the proposed methodology is not effective enough.

The global accuracy rate (GAR) is defined by the ratio of the number of times that the model correctly predicted (or classified) the primary mining method (as a top-1 method) and the total number of predictions (or classifications). Using the KNN-cosine similarity algorithm to find the nearest projects, it is required to set the parameter *k*, which is the number of projects used for predictions. In our experiments, we evaluate the model's accuracy for the number of projects (neighbourhood size) ranging from 2 to 10. Figure 4.3 shows the model's performance results regarding the global accuracy rate (GAR) using different projects for predictions. The GAR ranges from 45.5% to 63.6%. The results show that the highest GAR of 63.6% is achieved when we use the two nearest projects for predictions and recommendations. The lowest accuracy is observed when using 45.5% seven nearest projects for predictions. These results suggest that the accuracy tends to decrease as the number of projects used for predictions (neighbourhood size) increases up to 7, then increases slightly from 9 to 10 nearest projects.

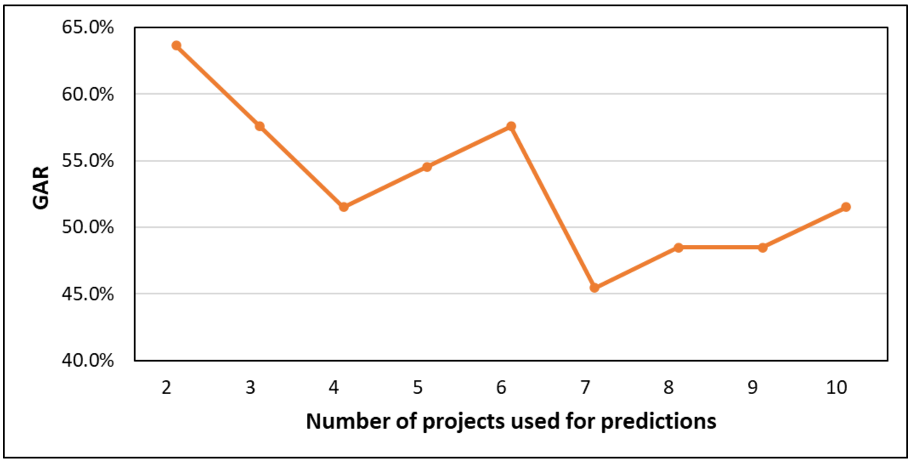


Figure 4.3: Performance of the proposed model in predicting primary mining methods using Global accuracy rate (GAR)

These results suggest that using the two nearest projects (the most similar projects to the target project), the model can correctly predict the primary mining method as the top-

1 with an accuracy of 63.6%. In common machine learning (ML) algorithms, this accuracy is considered moderate, perhaps not optimum. However, in our study, this accuracy is realistic, given the dataset size, which is significantly small for common ML algorithms. Additionally, the dataset in this study is a class imbalance (i.e., with uneven distribution of projects in each mining method) which might hinder the model's performance. Despite the moderate accuracy, the results suggest that the proposed memory-based CF approach is effective for underground mining methods selection.

The results depicted in Figure 4.3 focuses on the prediction of the primary mining method in the top-1 position of the recommendations. However, from these results, we cannot understand the model's capabilities always to include the primary mining methods in the top-3 list not just as the top-1 method but as the top-2 or top-3 method. This evaluation can be done using the mean reciprocal rank (MRR) to calculate the average rank accuracy of the primary mining method. The reciprocal rank (rank accuracy) calculates the reciprocal of the rank/position at which the primary mining method is placed in the top-3 list. For a target project, the rank accuracy can be 1, 0.5 and 0.33 when the primary mining method is placed in rank/positions 1, 2 and 3, respectively. If the primary mining method is not included in the top-3 list, the project's rank accuracy will be 0. The MRR is all projects' average reciprocal rank (rank accuracy). Figure 4.4 shows the results of the MRR (average rank accuracy) for the different number of projects used for predictions (neighbourhood size), which ranges from 67.2% to 77.8%. The results in Figure 4.4 follow the same trend as those shown in Figure 4.5 regarding the number of projects used for predictions. Similarly, the highest MRR of 77.8% is achieved with two nearest projects, the lowest of 67.2% for seven nearest projects, and a slight visible increase from 9 to 10 nearest projects.

An MRR close to 1 means that the model is more accurate/precise in placing the primary mining method in the top-1/rank-1 position among the top-3 relevant recommendations. These results suggest that when we use just the two nearest projects (most similar projects to the target project) for predictions and recommendations, the model can provide a list of top-3 recommended mining methods where the primary mining method is accurately included (77.8% accuracy). Again, this highlights the effectiveness of the proposed approach to handling the complex underground mining methods selection process.

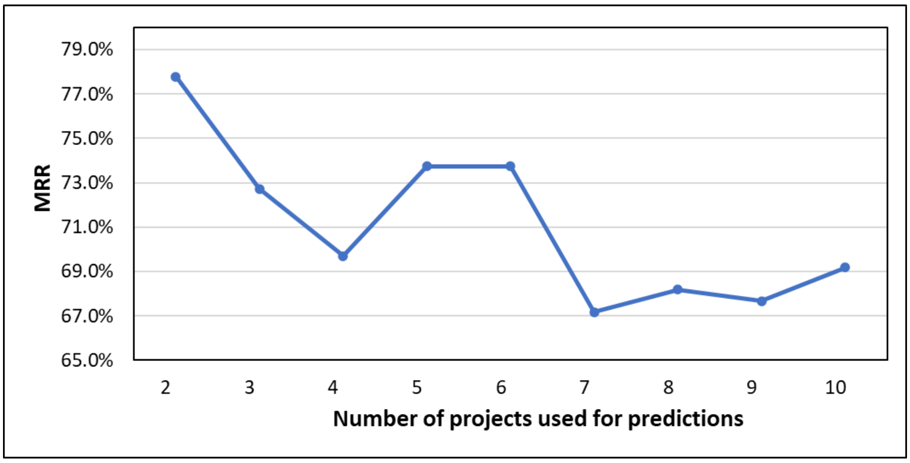


Figure 4.4: Performance of the proposed model in predicting the primary mining methods, showing the average rank accuracy or the mean reciprocal rank (MRR)

4.5.2 Relevance of the top-3 recommended mining methods

As stated previously, in the original input dataset, each project has top-3 most relevant mining methods, with one primary and two secondary methods. Therefore, the proposed model is tasked to predict and output a list of top-3 most relevant mining methods for each project. To evaluate the relevance of the top-3 recommendations, we use the F1-score, which is a decision support metric commonly used to evaluate machine learning (ML) classification models and information retrieval systems [98]–[100], [105]. F1-score is used to combine and balance the precision and the recall rates. Using the F-1 score is possible to measure how accurate and precise the model is to include the three most relevant mining methods of each project in the top-3 list of recommendations. The F1-score only cares if the model includes the three most relevant mining methods in the top-3 list of recommendations regardless of the rank/position of the primary mining method. Thus, this evaluation metric ignores the rank order of the mining methods. Figure 4.5 shows the results of the F1-score for the different number of projects (neighbourhood size), which ranges from 81.8% to 87.9%. The highest F1-score of 87.9% is achieved using the two nearest projects for predictions and recommendations. The lowest score is achieved when using the seven nearest projects for predictions, 81.8%. The trend of the F1-score results resembles the trend from the GAR and the MRR depicted in the previous section, in which the best performance of the model is achieved with the two nearest projects. These results suggest that the model provides the best results when using the two nearest projects (most similar projects to the target project) for predictions and recommendations. Using the proposed model to recommend the top-3 most relevant mining methods, the model will effectively include all three relevant mining methods with good accuracy of 87.9%.

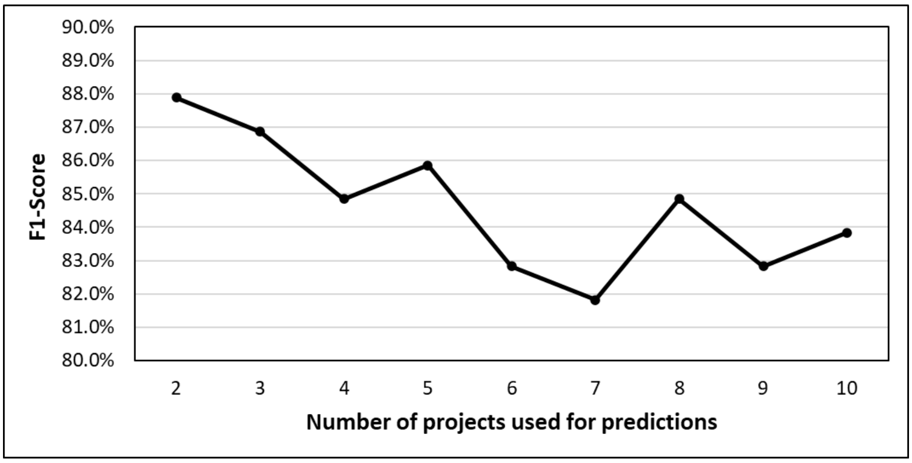


Figure 4.5: Results evaluating the relevance of the top-3 predicted and recommended mining methods using the F1-score

These results in sections 4.4.1 and 4.4.2 reveal that the proposed model can predict and recommend the primary mining methods with moderately good accuracy. On the other hand, the model performs even better when it comes to including each project's three most relevant mining methods in the top-3 list of recommendations. Thus, the results suggest that the memory-based collaborative filtering approach can be applied in selecting mining methods, revealing the approach's effectiveness in developing the proposed AI-mining methods recommendation system (AI-MMRS).

With continuous data collection to improve the quality and size of the dataset and optimisation of the model, it will be possible to build a robust AI-MMRS. The AI-MMRS can be practically implemented to aid in the decision-making task to provide recommendations of the top-3 underground mining methods (by learning from previous projects' mine planning procedures) that can be submitted for further evaluation (economic, environmental, political, and social).

4.6 Discussion of the Memory-based Collaborative Filtering Approach in Mining Methods Selection

4.6.1 Effect of number of projects used for predictions

In these experiments, we investigate the applicability of the memory-based collaborative filtering (CF) approach for predicting and recommending underground mining methods for the development of an AI-based mining methods recommendation system (AI-MMRS). The k-nearest neighbours (KNN) [93]–[96] algorithm with cosine similarity measure is chosen as the key algorithm to search for the k-nearest projects (most similar projects) used for predictions and recommendations. When using the KNN algorithm to compute similarities selecting the most optimum number of neighbours (number of projects) is an essential step. The number of neighbours used for predictions significantly affects the quality of predictions and recommendations [72], [94], [96], [110]. In our experiments, we evaluate the model's sensitivity by using a different number of projects (number of neighbours or neighbourhood size) for predictions and recommendations. Then we evaluate the quality of the results for the number of projects (neighbourhood size) using the global accuracy rate (GAR), mean reciprocal rank (MRR) and the F-1 score. Figure 4.6 shows the results of the model's performance using a different number of projects for predictions, combining the GAR, MRR and the F1-score. The results highlight the impact of the number of projects (neighbourhood size) on the quality of predictions and recommendations. All evaluation metrics used in the experiments have similar trends regarding the effect of the number of projects used for predictions. The best performance of the model is observed when we use two projects for predictions (and recommendations) with a tendency to decrease as the neighbourhood size increase until the lowest accuracy at the seven nearest projects. However, from 9 to 10 number of projects, the model's performance shows some improvement.

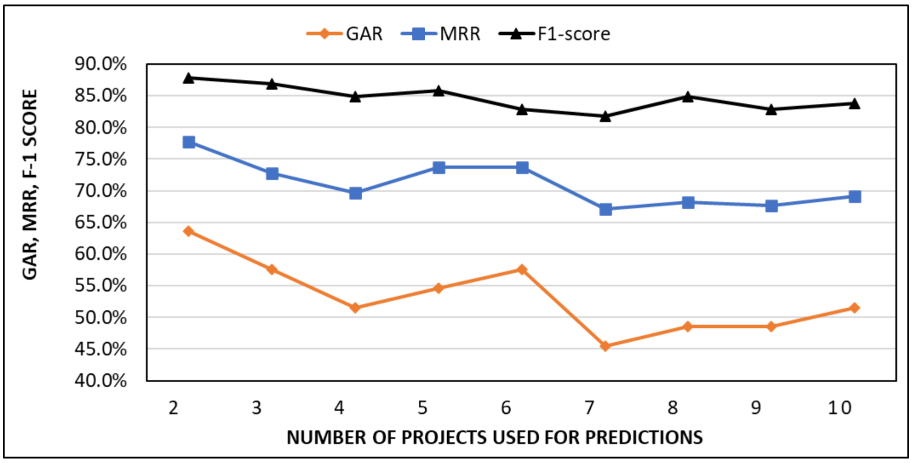


Figure 4.6: Effect of the number of projects used for predictions in the KNN algorithm showing GAR, MRR, F-1 score

According to the results, using two to three nearest projects for predictions and recommendations is the optimum choice, with the best performance at the two nearest projects. The best performance in smaller neighbourhood sizes may be related to the intrinsic characteristics of the dataset used for experiments. Some studies [110]–[112] have discussed the neighbourhood size's dependence on the datasets' intrinsic characteristics. The dataset for the experiments comprises thirty-three projects (datapoints), described by attributes such as ore strength, host-rock strength, orebody thickness, shape, and dip, and seven underground mining methods. First of all, the size of the dataset used in the experiments is considered small for common machine learning (ML) applications; usually, ML algorithms perform better on large datasets.

Furthermore, the dataset is class imbalanced [113]–[115], with an uneven distribution of projects (datapoints) in each mining method, as shown in Figure 4.7. The hypothesis on the KNN algorithm in this study's dataset is that projects included in the same mining method have similar attributes or similar orebody characteristics. Since some mining methods have only one project (datapoint), using more projects for predictions would probably add bias or noise. For this reason, the proposed model performs best using a smaller number of projects (neighbourhood size) for predictions.

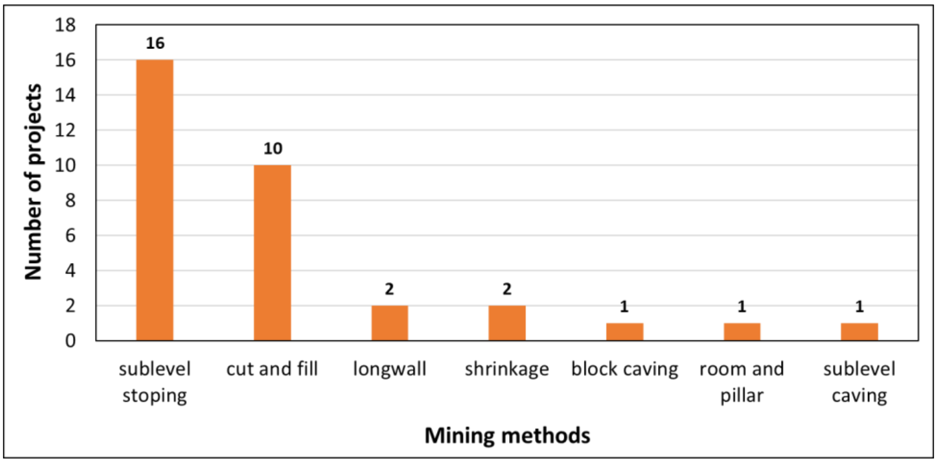


Figure 4.7: Description of the dataset used for experiments: distribution of projects (datapoints) in each underground mining method

We limited these experiments to 10 nearest projects because of the size and the characteristics of the dataset used for experiments. However, further implementation of the proposed algorithm on larger datasets is required to select the most optimum neighbourhood size by following the same steps designed in this study.

4.7 Relevance of the Proposed Memory-based Collaborative Filtering Approach for Mining Methods Selection

This study introduces the application of the recommendation system concept, specifically collaborative filtering (CF), in mining methods selection (MMS) for developing an AI-based mining methods recommendation system (AI-MMRS). As such, in this chapter, we investigated the applicability of the memory-based CF approach to predicting and recommending top-N underground mining methods. As previously demonstrated, the memory-based CF approach performs well in predicting and recommending top-3 relevant mining methods, thus, proving effective in MMS. Therefore, revealing the effectiveness of the approach to developing the proposed AI-MMRS. Despite the limitations on getting a fair amount of data for training the machine learning (ML) models, these results demonstrated that the proposed approach performs quite well in generating recommendations of top-3 most relevant underground mining methods. With continued data collection and training, we will be able to improve the quality of the proposed system, thus building a robust system. The achievements of the proposed memory-based approach are as follows:

- The proposed approach recommends mining methods based on the similarity between different projects in terms of orebody characteristics (ore strength, host-rock strength, orebody thickness, shape, and dip) based on practical procedures of previous mining projects (in the study's database), including all seven currently applied underground mining methods (sublevel stoping, cut and fill, shrinkage, room and pillar, longwall, block caving and sublevel caving).
- The model can recommend not just one but the top-3 most relevant mining methods with good accuracy by providing only five factors as input variables (which are easily accessible). Therefore, the AI-MMRS can be practically implemented in cases with limited access to detailed information about the orebody characteristics. Especially during the early stages of mine project development for projects requiring recommendations of a set of mining methods that will be submitted for further economic, technological, environmental, and political analysis during the mine planning process.
- Since the system is developed based on data from mining projects dating from the 2000s, the system aids the benefit of providing up-to-date solutions following the change in factors-mining methods classification and technological advancement.

Perhaps the only limitation of the model is the dependency on the UBC-MMS tool [4] to find the two secondary mining methods in the input dataset.

4.8 Summary of the Proposed Memory-based Collaborative Filtering Approach for Mining Methods Selection

In this chapter, we investigated the applicability of memory-based collaborative filtering (CF) to predicting and recommending top-3 underground mining methods for developing an AI-based mining methods recommendation system (AI-MMRS). The dataset used for experiments is composed of thirty-three projects which are described by five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip) and seven underground mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping). With the aid of the UBC-MMS tool to find the two secondary mining methods for each project, we designed a diversified system to output the top-3 most relevant mining methods for a target project. The proposed memory-based approach uses the k-nearest neighbours (KNN) with the cosine similarity algorithm to measure the similarity among the projects and the weighted sum method for predictions and recommendations. As such, we evaluate the capability of the proposed model to accurately predict and recommend the top-3 most relevant mining methods: one primary and two secondary methods for each project. The results show that the model can predict the projects' primary mining methods with an accuracy of 63.8%. Furthermore, the model performs even better in predicting and recommending the top-3 relevant mining methods, with an accuracy of 87.9%.

Despite the limitations on getting a fair amount of data for training the machine learning (ML) models, these results reveal the effectiveness of the memory-based approach for underground mining methods selection, thus, applicable in developing the AI-MMRS. The model can effectively predict and recommend the top-3 most relevant mining methods based on only five input variables. However, continuous data collection and model training is required to improve the quality of the recommendations, thus, building a robust system. The proposed AI-MMRS can be practically implemented in cases with limited access to detailed information about the orebody characteristics, especially during the early stages of mining project development, to recommend a set of mining methods that will be submitted for further evaluations (economic, technological, environmental, and political).

5 Applying Nonnegative Matrix Factorization for Predicting Mining Methods and Possible Missing Values in the Input Dataset

This Chapter assesses the capability of the nonnegative matrix factorization (NMF) algorithm to address the “data sparsity problem” by predicting mining methods and other variables critical for mining methods selection (MMS). As mentioned in Chapter 3, one of the limitations of this study on developing an AI-based mining methods recommendation system (AI-MMRS) is the “data sparsity problem” caused by the lack of information about the required input variables from the mining project database. As most ML algorithms are not very effective on very sparse datasets, the “data sparsity problem” forced the reduction of the size of the input datasets, resulting in small input datasets. The NMF algorithm is proposed as a pre-processing algorithm to address the “data sparsity problem” by predicting possible missing information from the input datasets, which can then enable data augmentation (increase the size of the input datasets). By evaluating the applicability of the NMF algorithm to predicting mining methods, we simultaneously evaluate the capabilities of the same algorithm to predict possible missing values from the sparse dataset, thereby addressing the “data sparsity problem”. The dataset used in this chapter is described in Chapter 3, which comprises thirty projects’ historical data.

5.1 Why Propose Nonnegative Matrix Factorization Algorithm

Collaborative filtering (CF) is a recommendation system that collaboratively predicts users' future preferences based on other users' preferences. CF is based on the concept that users who purchased/liked similar items in the past might likely be interested in purchasing similar items in the future; therefore, an item that was purchased/liked by a user (or group of users) in the past might be recommended to a similar user (or users) in the future. CF can be implemented in memory-based and model-based approaches[19]. Memory-based, also known as a neighbourhood-based approach [76], [78]–[80], is simple, straightforward, and easy to understand or interpret. In this approach, there is no need for training a model: for every recommendation to be generated, the algorithms utilize the entire user-item matrix to generate predictions directly. In the model-based approach, [75]–[77] machine learning (ML) algorithms are used to train models that automatically learn user-item interaction patterns to then predict users' ratings of unrated items and/or generate recommendations. In chapter 4, we evaluated the applicability of memory-based CF for underground mining methods selection (MMS). Results from chapter 4 demonstrated the effectiveness of the proposed memory-based approach for predicting and recommending top-3 relevant underground mining methods, thus, applicable for developing the proposed AI-MMRS. However, the memory-based CF approach is not very effective on very sparse datasets because it depends on the ratings of other users to provide recommendations. As already highlighted in previous chapters, the central concept of this entire study is to explore available mining projects database to extract useful information for developing the AI-MMRS. This way, it is possible to develop a system that will recommend the most appropriate mining methods by learning from previous mining projects' procedures. To recall that one of the limitations of this study is the lack of information about the required input variables from the mining project database. The lack of the required information caused the data sparsity problem, and most ML algorithms, including the memory-based CF approach proposed in chapter 4, are ineffective on sparse datasets. For this reason, the entire study is based on small and biased datasets (i.e., class imbalanced datasets), which negatively affect the quality and models' performance.

This chapter proposes the nonnegative matrix factorization (NMF) algorithm as a pre-processing algorithm to solve the data sparsity problem. As such, we assess the NMF algorithm's capability to predict missing values from the sparse dataset to enable data augmentation, thus improving the datasets' quality (increasing the input datasets' size). The main focus of this chapter is to evaluate the applicability of the NMF in predicting underground mining methods. Then propose the same strategy to predict possible missing values about other required input variables in the dataset.

5.2 Nonnegative Matrix Factorization for Predicting Missing Data in Recommendation Systems

Nonnegative matrix factorization (NMF) is a type of matrix factorization (MF) algorithm [116], [117] in CF recommendation systems [35], [70], [71], [118] that is based on the nonnegativity constraint. NMF was popularized by Lee and Seung [119], [120], who applied the NMF to learn parts of faces and semantic features. Since then, it has been applied for different purposes: recommendation systems [22], [121], astronomy [122], and audio signal processing [123]–[125], among others. In recommendation systems, MF (and NMF) are usually applied to predict missing (or unobserved) ratings from a sparse user-rating matrix in order to recommend the most relevant items (usually, items with the highest ratings are recommended to the users).

Recommendation systems are aimed at helping users deal with information overload by making personalized suggestions or suggesting the most relevant/popular items. In order to make personalized suggestions, recommendation systems [21], [43] collect different types of data about the users and the items. CF [20], [70], [71] recommendation systems make personalized suggestions mainly based on the user-item interaction matrix that contains ratings that different users give to evaluate items. A sparse user-item matrix originated when there are missing ratings in the matrix: some users do not rate items or even because users have not checked the item. Usually, the task of matrix factorization algorithms is to predict the missing ratings from the sparse user-item matrix. Let **X** be the user-item interaction matrix with missing ratings, and the task is to generate recommendations for user 3 (U3), who has watched movies 1 and 3. First, the NMF algorithm is trained to predict the missing ratings in **X**. In NMF, **X** is the input used to generate an output matrix **X'** with all missing ratings predicted. Based on the predicted/output matrix **X',** movie 2 (or I2) would be recommended to U3 as it has the highest predicted rating of 5.1 compared to I4, which has a predicted rating of 1.1.

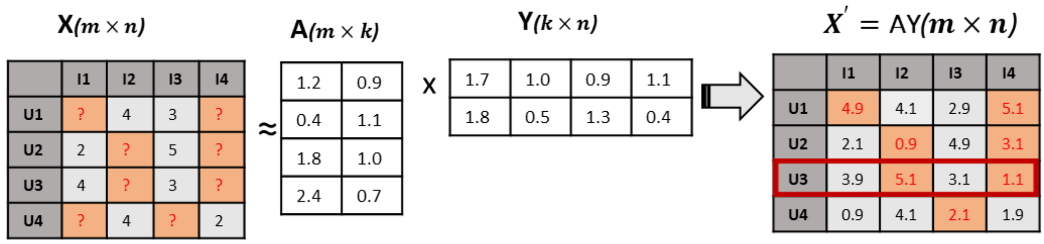


Figure 5.1: NMF algorithm for predicting missing values from a sparse matrix

As shown in Figure 5.1, the problem of NMF [121], [126], [127] works by decomposing a nonnegative sparse matrix $\mathbf{X} \in \mathbb{R}_{+}^{m\times n}$ into two smaller matrices $\mathbf{A} \in \mathbb{R}_{+}^{m\times k}$ and $\mathbf{Y} \in \mathbb{R}_{+}^{k\times n}$, such that $\mathbf{X} \approx \mathbf{AY}$ by minimising a cost function through optimization algorithms. Matrices **A** and **Y** are called "basis" and "coefficient", and **k** is the rank of matrices **A** and **Y**. The k-rank is usually chosen to be smaller or equal to the number of rows and columns in the original matrix **X** so that the originated matrix **AY** is not bigger than the original matrix **X**: $\mathbf{k} \leq \mathbf{min}\,(\mathbf{m}, \mathbf{n})$. In the NMF approach to decompose the original matrix **X,** the first **A** and **Y** matrices are usually randomly assigned. Then in every iteration, the values of **A** and **Y** matrices are updated so that **X** ≈ **AY**. The most optimum matrices, **A** and **Y,** are found by minimizing the error or cost function between the original matrix **X** and the approximated/output matrix **AY** through optimization algorithms. The most common cost functions are the Euclidean distance (or Frobenius norm) and the Kullback-Leibler Divergence [119], [128], [129]. Alternating least squares and gradient descent [126], [127] are the most commonly applied algorithms to optimize the cost functions and multiplicative updating rules.

In this study, we implement one of the latest variants of matrix factorization, the weighted nonnegative matrix factorization (WNMF). WNMF [22], [130]–[133] is a variant of NMF known for its powerful ability to deal with a sparse dataset (i.e., a dataset with known and unknown/missing values) and effectively predict unknown values. The problem of WNMF works similarly to the NMF by decomposing a nonnegative matrix $\mathbf{X} \in \mathbb{R}_+^{\mathrm{m}\times\mathrm{n}}$ into two smaller matrices $\mathbf{A} \in \mathbb{R}_+^{\mathrm{m}\times\mathrm{k}}$ and $\mathbf{Y} \in \mathbb{R}_+^{\mathrm{k}\times\mathrm{n}}$, such that **X** ≈ **AY** by minimising a cost function through optimization algorithms. In WNMF, a binary matrix **W** was introduced as the weight of matrix **X** [133]. Where one is assigned if the value $\mathbf{X_{ij}}$ is observed/known; otherwise, 0 is assigned. The Euclidean distance and the Kullback-Leibler Divergence [133] are the most common cost functions optimized through multiplicative updating rules algorithms.

Equation (5.1) shows the Weighted Euclidean Distance cost function under multiplicative updating rules in Equations (5.2) and (5.3).

$$\frac{1}{2}\|X - AY\|_W^2 := \frac{1}{2}\sum_{ij}[W \circ (X - AY) \circ (X - AY)]_{ij} \tag{5.1}$$

$$A \leftarrow A \circ \left({[(W \circ X)Y^T]} \middle/ {[(W \circ (AY))Y^T]} \right) \tag{5.2}$$

$$Y \leftarrow Y \circ \left({[A^T(W \circ X)]} \middle/ {[A^T(W \circ (AY)]} \right) \tag{5.3}$$

The Weighted Kullback-Leibler Divergence function is shown in Equation (5.4), under multiplicative updating rules in Equations (5.5) and (5.6):

$$D_W\left({}^{X}\!/_{AY}\right) := \sum_{ij}[\mathrm{W} \circ \left(\mathrm{X} \circ \log_\circ {}^{[X]}\!/_{[AY]} - \mathrm{X} + \mathrm{AY}\right)]_{ij} \tag{5.4}$$

$$AA \leftarrow \left({}^{[A]}\!/_{[Y^TW]}\right) \circ \left(Y^T\left[{}^{[(W \circ X)]}\!/_{[AY]}\right]\right) \tag{5.5}$$

$$Y \leftarrow \left({}^{[Y]}\!/_{[WA^T]}\right) \circ \left(\left[{}^{[(W \circ X)]}\!/_{[AY]}\right]A^T\right) \tag{5.6}$$

Where $\mathrm{B} \circ \mathrm{C}$ denotes Hadamard multiplication (or element-wise multiplication) of matrices B and C. ${}^{\mathrm{B}}\!/_{\mathrm{C}}$ is the Hadamard division of the matrices B and C. $\log_\circ \mathrm{B}$ is the element-wise logarithm of B.

5.3 Applying WNMF Algorithm for Predicting Mining Methods

The input dataset used in this chapter is described in chapter 3, Figure 3.2. The dataset is similar to the one used to implement the memory-based CF approach in Chapter 4. The only difference is in the number of projects/samples and the underground mining methods. The dataset used in this chapter comprises thirty projects/samples and five underground mining methods such as block caving, cut and fill, room and pillar, shrinkage, and sublevel stoping. The input variables/attributes are the same for all datasets: ore strength, host-rock strength, orebody thickness, shape, and dip. In this chapter, we use a similar approach used in Chapter 3. In order to incorporate the WNMF into predicting underground mining methods, we use a similar approach as the one used in the previous chapter 4. In chapter 4, we performed experiments to evaluate the applicability of the memory-based approach in underground MMS based on the approach used for offline evaluations of recommendation systems [84], [97]–[100].

As the first task of the WNMF algorithm is to predict the mining methods: in the practical application of the AI-MMRS, we assume that the target project will have information about the attributes (i.e., ore strength, host-rock strength, orebody thickness, shape, and dip). Thus, only one value will be missing/unknown for the target project: the mining method. In the experiments, we simulate the practical application of the proposed methodology in the AI-MMRS by masking/hiding the known value of the mining method for a given target project. Thus, creating a sparse dataset for the target project with one unknown/missing value. After that, the sparse dataset is used as input to train the WNMF model for predicting the missing/masked value. The hidden/masked mining methods values are kept as ground truth data and used to evaluate the WNMF model's prediction accuracy. The accuracy is used to measure how well WNMF predicts the missing values (the masked/hidden mining methods values) by comparing the predicted values with the original masked values. Figure 5.2 shows the workflow of the proposed methodology for practical experiments to evaluate the applicability of the WNMF algorithm in MMS.

The WNMF algorithm is proposed as a pre-processing approach in the AI-MMRS for predicting possible missing values from the input dataset. However, it can also predict and recommend underground mining methods. In Figure 5.2, ore strength, rock strength, thickness, shape, dip, and mining method are abbreviated as OS, RS, TH, SH, DP, and MM, respectively. In step 3, the highlighted squares (in red/black) in the matrices represent the masked/hidden values in the MM column. In step 5, the highlighted value (in bold red/black) is the MM value predicted by the WNMF model. In step 6, the highlighted value (in bold red/black) in the output matrix and original dataset represents the predicted (and denormalized) and original MM values, respectively.

The workflow details depicted in Figure 5.2 are explained in detail in section 5.3 on practical experiments.

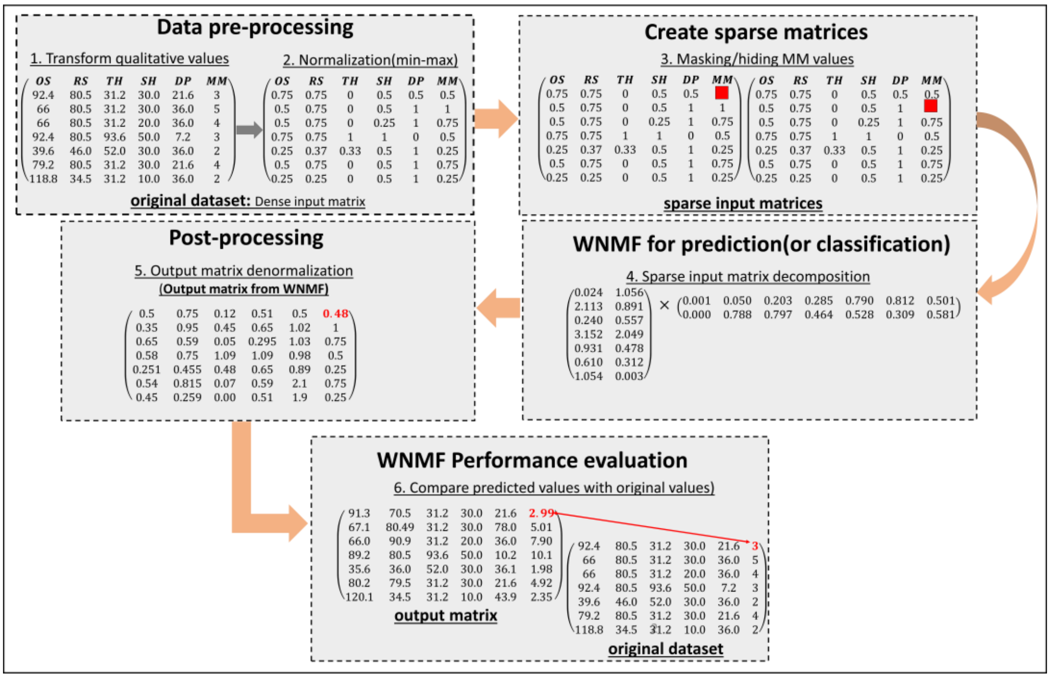


Figure 5.2: Workflow of the proposed methodology for practical experiments: WNMF algorithm for predicting mining methods and possible missing values of the input variables

5.4 Practical Experiments on the Implementation of the WNMF Algorithm for Predicting Mining Methods

5.4.1 Dataset

This chapter is based on one of the datasets created in chapter 3, described in Figure 3.2. The dataset for this chapter contains historical data of thirty mining projects/samples (or case studies) and five mining methods/classes: block caving, cut and fill, room and pillar, shrinkage, and sublevel stoping. Similar to the dataset used in chapter 4, there are five input variables: ore strength, host-rock strength, orebody thickness, shape, and dip. Table 5.1 illustrates the short representation of the input dataset used for experiments, where the first column contains the mining projects id (denoted as PJ001, PJ002…, PJ0030). From the second to the fifth column are the five input variables/factors, and in the last column are the mining methods selected/considered in each project. This dataset shows historical information on thirty mining projects regarding the selected or considered mining methods to recover the orebody deposits. The orebody deposits in each project are described by their orebody characteristics: geotechnical properties (ore and host-rock strength) and orebody geometry (orebody thickness, shape, and dip).

Table 5.1: Short representation of the raw input dataset used for experiments

Project	Ore strength	Host-rock strength	Thickness	Shape	Dip	Mining method
PJ001	Strong	Strong	Narrow	Tabular	Intermediate	Room and pillar
PJ002	Moderate	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ003	Moderate	Strong	Narrow	Irregular-tabular	Steep	Shrinkage
PJ004	Strong	Strong	Intermediate	Tabular	Flat	Cut and fill
PJ005	Weak	Weak-moderate	Narrow	Tabular	Flat	Cut and fill
PJ006	Moderate-strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
…	…	…	…	…	…	…
PJ030	Weak	Moderate	Narrow	Tabular	Steep	Shrinkage

5.4.2 Data Pre-processing

Similar to the raw dataset used in chapter 4, the dataset in Table 5.1 is composed of qualitative or categorical values. The first step in pre-processing data consists of transforming the categorical values to numerical values because the WNMF is only effective in datasets with numerical/quantitative values. Similar to chapter 4, categorical values are transformed using the label encoding strategy [102]. The five input variables are also transformed based on their objective weights calculated using the Entropy method in chapter 2. In this chapter, the five mining methods/classes are encoded using five numerical values from 1 to 5. Table 5.4 shows the transformed numerical input dataset for experiments.

Table 5.2: Transformed input dataset used for experiments

Project	Ore strength	Host-rock strength	Thickness	Shape	Dip	Mining method
PJ001	92.4	80.5	31.2	30	21.6	3
PJ002	66	80.5	31.2	30	36	5
PJ003	66	80.5	31.2	20	36	4
PJ004	92.4	80.5	52	30	7.2	2
PJ005	39.6	46	31.2	30	7.2	2
PJ006	79.2	80.5	31.2	30	36	5
…	…	…	…	…	…	…
PJ030	39.6	57.5	31.2	30	36	4

Once the categorical values are transformed into numerical values, the next step involves normalizing the values. Normalization [134] of values is a pre-processing step consisting of rescaling values in the dataset in order to adjust values on different scales to a common scale. Here, normalization is performed on the overall dataset by applying the Min-Max normalization method [18], with new values ranging from 0 to 1.

5.4.3 Creating sparse matrices

Matrix factorization [116], [117], [121] algorithm was introduced in the recommendation to handle the sparsity problem by effectively predicting missing values from sparse matrices. In this chapter, we employ the weighted nonnegative matrix factorization (WNMF) for predicting missing values of mining methods. The input dataset for practical experiments in Table 5.1 is a dense matrix, i.e., all values are filled or known; thus, there are no missing or unknown values. In these experiments, to simulate the practical application of the proposed methodology in the AI-MMRS, sparse datasets are created by masking/hiding the value of the mining method for a given target project. The hidden/masked ratings are kept as ground truth data that will be used for evaluating the accuracy and quality of recommendations. Since there are thirty projects in the input dataset, masking/hiding one mining method value at a time results in thirty sparse datasets with one missing/unknown mining method for each project. The thirty sparse datasets are then used as inputs to the WNMF algorithm. In this case, the task of the WNMF model is to predict the missing masked mining methods values for each project.

5.4.4 Predictions (classification) using the WNMF algorithm

For the experiments, in order to decompose the input sparse matrix **X,** the WNMF algorithm is based on the random initialization of matrices **A** and **Y** [129], [135]. The Euclidean distance is used as a cost function under the multiplicative updating rules optimization algorithm. The k-rank parameter ranges from 1 to 6, following the rule $\boldsymbol{k} \leq \mathbf{min}(\boldsymbol{m}, \boldsymbol{n})$**.** For a given target project, WNMF predicts the missing masked/hidden mining method value in the respective input sparse dataset (created in section 5.3.2). We run the WNMF algorithm for each target project by changing the k-rank parameter from 1 to 6. Therefore, for each target project, we will have six predicted outputs (i.e., outputs from rank-1, rank-2, rank-3, rank-4, rank-5 and rank-6). The random initialization method of the initial matrices **A** and **Y** results in unstable results, i.e., the model produces different outputs at every experiment. To measure the variation caused by the random initialization, we run the WNMF algorithm twenty times for each sparse dataset. Table 5.5 shows the parameters of the WNMF algorithm used in the practical experiments.

Table 5.3: Parameters of the WNMF algorithm for prediction of underground mining methods

Parameters	Description
k-rank	1, 2, 3, 4, 5, 6
Cost function	Euclidean distance
Optimization algorithm	Multiplicative updating rules

After the predictions, a post-processing step is performed to denormalize the overall values in the output matrices from the WNMF to scale the normalized values to their initial/original scales.

5.4.5 Performance evaluation metrics

Usually, matrix factorization algorithms are evaluated using metrics used to evaluate ratings-based recommendation systems to optimize the predicted rating to be closer to the masked/hidden ratings. The evaluation metrics used for rating prediction are also similar to those used in machine learning (ML) regression models, such as the Root Mean Square Error (RMSE), Mean Square Error (MSE) and Mean Absolute Error (MAE) [42], [84], [97], [99]. In these experiments, the intention is not to minimize the error of the predicted rating. The task of the WNMF model is to predict labels or classes corresponding to mining methods. For this reason, we employ classification accuracy metrics to evaluate the performance of the WNMF model. The global accuracy rate (GAR) is the main evaluation metric. The GAR was also used as one of the metrics in Chapter 4. Here, the GAR is the number of correctly predicted classes divided by the total number of predictions (or classifications), as shown in Equation (5.7).

$$GAR(\%) = \frac{number\ of\ correct\ classifications}{total\ number\ of classifications} \times 100 \tag{5.7}$$

5.5 Experimental Results of the Implementation of the WNMF Algorithm to Predict Mining Methods

This chapter aims to assess the capability of the nonnegative matrix factorization (NMF) algorithm to address one of the limitations of the study: the data sparsity problem. As such, we evaluate the capability of the NMF to predict missing values of mining methods and other possible missing values in the dataset. This section presents the results of the practical experiments on implementing the weighted nonnegative matrix factorization (WNMF) algorithm to predict missing values of mining methods. The input dataset used in the experiments comprises thirty projects described by five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip) and five underground mining methods (block caving, cut and fill, room and pillar, shrinkage, and sublevel stoping).

In the experiments, we simulate the practical application of the NMF algorithm in the AI-MMRS by masking/hiding the known value of the mining method for a given target project. Thus, creating a sparse dataset for the target project with one unknown/missing value. After that, the sparse dataset is used as input to train the WNMF model for predicting the missing value. The hidden/masked mining methods values are kept as ground truth data and used to evaluate the WNMF model's prediction accuracy. The global accuracy rate (GAR) measures how well WNMF predicts mining methods' masked/hidden values.

5.5.1 General prediction accuracy of the missing values of mining methods

Given the input dataset of thirty projects with five variables, the WNMF model is tasked to predict each project's missing mining method value. As a result of the six k-ranks parameters, the WNMF model generates six output predicted values of mining methods for each target project. For each project, the global accuracy rate (GAR) is evaluated from the output of k-rank 1 to k-rank 6. Generally, for one round of experiments, we start by computing the accuracy rate by counting the number of correct predictions for each target project regardless of the k-rank. Then, we compute the GAR by counting the number of correct predictions divided by 30, which corresponds to the total predictions (there are 30 projects in the dataset). The random initialization method of the initial matrices **A** and **Y** results in unstable results, i.e., the model produces different outputs at every experiment. To measure the variation caused by the random initialization, we run the WNMF algorithm twenty times for each sparse dataset. Figure 5.3 shows the results of the GAR of the twenty rounds of experiments, with GAR ranging from 60.0% to 76.7%, an average GAR of 67.5% and a standard deviation of 0.05%. The standard deviation value shows a relatively low variation of the accuracy from the mean accuracy among the twenty rounds of experiments.

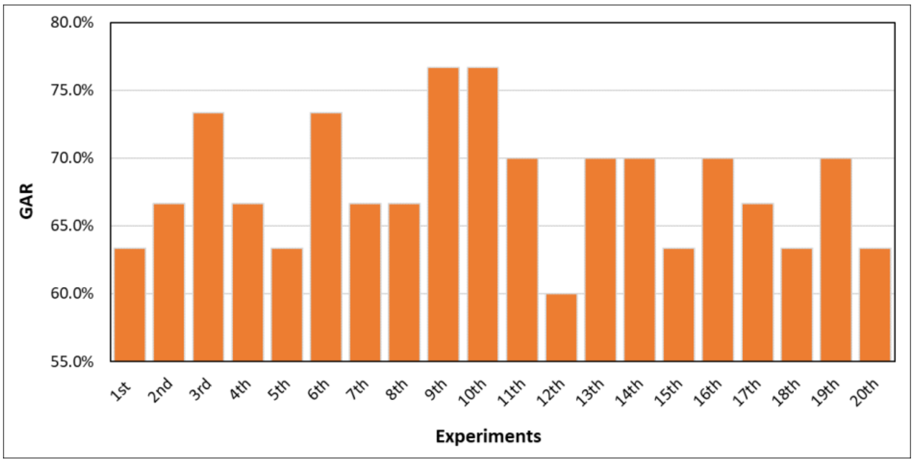


Figure 5.3: Prediction accuracy of the WNMF algorithm showing Global accuracy rate (GAR)

Despite the variation of the GAR of the WNMF model, the average accuracy of 67.5% is considered reasonable, though not optimum. In a dataset with missing values, the WNMF model can effectively predict the missing value with moderate accuracy.

5.5.2 Prediction accuracy of the projects' mining methods

We evaluate each mining project's global accuracy rate (GAR) for the twenty rounds of experiments to understand the WNMF model's capabilities to predict each project's mining methods. The results are shown in Table 5.6: the first column represents the mining projects (denoted as PJ001, PJ002, PJ003). As we can observe, the model correctly predicted most of the project's mining methods with high accuracy. However, there are twelve mining projects with low accuracy (accuracy lower than 55%), highlighted in Table 5.6 (bold and underlined). The results suggest that the WNMF model predicted most of the mining methods well but failed to correctly predict the twelve projects' mining methods. Consequently, the low accuracy in the twelve projects negatively affects the model's overall performance.

Table 5.4: The accuracy rate of each mining project for the twenty rounds of experiments

Mining project	**GAR**	**Mining project**	**GAR**
PJ001	**30%**	**PJ016**	**0%**
PJ002	100%	**PJ017**	**55%**
PJ003	100%	PJ018	100%
PJ004	**45%**	PJ019	100%
PJ005	100%	**PJ020**	**25%**
PJ006	100%	**PJ021**	**50%**
PJ007	100%	**PJ022**	**25%**
PJ008	100%	PJ023	100%
PJ009	100%	PJ024	100%
PJ010	**5%**	**PJ025**	**35%**
PJ011	70%	PJ026	100%
PJ012	100%	PJ027	100%
PJ013	**0%**	PJ028	100%
PJ014	90%	**PJ029**	**0%**
PJ015	**0%**	PJ030	100%

These results suggest that the proposed NMF algorithm can be effectively implemented for predicting underground mining methods with further improvements. Therefore, it is required to understand the reasons behind the low performance of the model to predict some mining methods.

5.5.3 Relationship between data distribution and prediction accuracy

Table 5.6 clearly shows that the proposed model predicted most of the project's mining methods well. However, we observe that model failed to predict some projects' mining methods and some projects with close to 0%. To understand the reason behind the low performance of the model in the twelve projects, we investigate the relationship between the global accuracy rate (GAR) and intrinsic characteristics of the dataset. For that, we first try to understand the relationship between the GAR and the distribution of projects (datapoints) in each mining method. We calculate the average GAR of the projects labelled in each mining method. Figure 5.4 shows the relationship between the distribution of the data and the GAR in each mining method. The left vertical axis in Figure 5.4 depicts the data distribution of the mining projects (datapoints) in each mining method. As it is notorious, the input dataset is class imbalanced [113], [136], [137]. Class imbalance happens when the distribution of datapoints among the classes is uneven, with more samples in the majority class and fewer in the minority class. We can observe from Figure 5.4 that in a total of thirty projects, thirteen are classified in sublevel stoping, nine in cut and fill, four in shrinkage stoping, three in room and pillar and one in block caving. In this case, the input dataset is multi-class imbalanced with sublevel stoping mining method as the majority class.

The global accuracy rate (GAR) of the projects in each mining method in Figure 5.4 is on the left vertical axis. We can observe that mining methods with a single (block caving: 0%) or a few projects (room and pillar: 20%) have the lowest GAR. Whilst mining methods with a higher number of projects have higher accuracy (shrinkage: 81.3%, cut and fill: 77.2%, and sublevel stoping: 73.1%). Furthermore, sublevel stoping is the majority class with the highest number of projects, yet the accuracy is lower than that of the shrinkage and cut-and-fill methods. These results suggest that the WNMF model performs better predicting (classifying) mining methods with more projects. However, the model might be sensitive to outliers, i.e., mining methods with a single (block caving) and highest (sublevel stoping) number of projects.

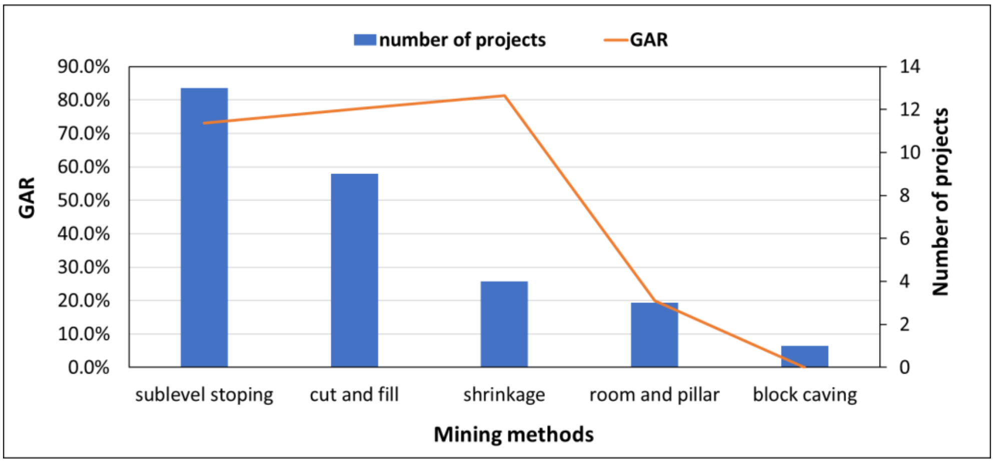


Figure 5.4: Relationship between the global accuracy rate (GAR) and the distribution of projects in each mining method

These results generally suggest that the WNMF model is sensitive to imbalanced datasets. Studies done in imbalanced datasets support the findings from this chapter. Most studies [22], [113]–[115], [136]–[140] done on machine learning (ML) models with class imbalance dataset suggest that class imbalance dataset negatively affects the performance of most ML models. The findings from this study are a good indicator for analysing and improving the dataset's quality to improve the model's overall performance. Therefore, it is important to keep collecting data from the mining project's database to create good-quality datasets.

5.6 Discussion on the WNMF Algorithm for Predicting Mining Methods

5.6.1 K-rank and initialization of the WNMF algorithm

In matrix factorization algorithms [116], [117], [119], [121], [126], [127], [141], [142], the k-rank is one of the required parameters, and the user must set it. Usually, the k-rank must be chosen to be smaller or equal to the number of rows and columns of the original input matrix, i.e., $\boldsymbol{k \leq \min(m, n)}$. The k-rank can affect the quality and accuracy of the predictions. For our experiments, the k-rank parameter was set to be from 1 to 6, meaning that the WNMF model was run six times for each sparse dataset, changing the k-rank. To investigate the performance of the WNMF model in each k-rank, we describe the results of the first ten rounds of experiments. Table 5.7 shows the prediction accuracy in each k-rank in terms of the average global accuracy rate (GAR).

Table 5.5: Average global accuracy rate (GAR) in each k-rank

	k=1	**k=2**	**k=3**	**k=4**	**k=5**	**k=6**
Average GAR	37%	40%	35%	27%	23%	24%

In our experiments, the WNMF model produces different outcomes among k-rank, similar to what is observed in most studies using matrix factorization algorithms [133], [143], [144]. In most cases, including recommendation systems, the optimization of the model is based on the k-rank. The most optimum k-rank is the one that produces the prediction with the lowest error. Usually, the optimization of the NMF algorithm is done by choosing the most optimum k-rank; the higher the accuracy, the most optimum the k-rank. As shown in Table 5.7, the average accuracy shows that rank-2 is the most optimum, with the highest average accuracy of 40%, followed by rank-1 and rank-3 with 37% and 35%, respectively. The lowest performance is observed in rank-5 and rank-6, with average accuracies of 23% and 24%, respectively. In this case, rank-1 to rank-3 would be the most optimum for the dataset. However, selecting rank-1 to rank-3 as the most optimum for the entire dataset might be a biased optimization method because the optimum k-ranks are different for each project, i.e., the k-rank with the best accuracy rate varies in each project. For some projects, the best performance is observed in rank-4 to rank-6 rather than in rank-1 to rank-3. Therefore, in this study, the proposed evaluation approach is not solely based on the k-rank with the best accuracy. Here we analyse the results based on the most frequently predicted classes from rank-1 to rank-6 in 10 rounds of experiments. For example, for a target project, we collect the output classes for rank-1 to rank-6 in 10 rounds of experiments, resulting in 60 predictions. After that, from the 60 predicted classes, we compute the three or four most frequently predicted classes in a ranking order; say: sublevel stoping, shrinkage, cut and fill, and sublevel caving are the four most frequently predicted in a ranking order. Therefore, sublevel stoping, shrinkage, cut and fill, and sublevel caving will be recommended as the top-4 methods for the project. This strategy will be further developed in future studies on the improvement of the performance of the model.

The other aspect worth analysing is the random initialization method of the WNMF algorithm used in the experiments. The random initialization method [135] causes instability of the predictions, which can negatively affect the model's performance. In our experiments, we expect the WNMF model to output values from 1 to 5 corresponding to the five mining methods (block caving: 1, cut and fill: 2, room and pillar: 3, shrinkage: 4, and sublevel stoping: 5). However, in some cases, the WNMF model can output values that are out of range of the expected values: values over 5 to the hundred scales. Therefore, it is required to investigate different initialization methods, some of which are suggested in some studies on the NMF algorithms [129], [135], [142], [145]–[152].

In general, the results of assessing the capability of the WNMF algorithm for addressing the data sparsity problem reveal a moderate model performance, requiring further improvements. The same approach applied for predicting missing values of mining methods can be effectively implemented to predict possible missing values of other required variables in the dataset, thereby addressing the "data sparsity problem". Thus, enabling data augmentation and the practical implementation of the proposed AI-MMRS system when information about some input variables is not available or accessible. The WNMF algorithm is proposed as a pre-processing approach in the AI-MMRS for predicting possible missing values from the input dataset. However, it can also be implemented to predict and recommend underground mining methods. Therefore, with continuous improvement, the WNMF algorithm can be incorporated into developing the proposed AI-based mining methods recommendation system (AI-MMRS). The improvements focus on improving the initialization method and continuous data collection to improve the quality and size of the dataset.

5.7 Summary of the Implementation of The NMF Algorithm for Predicting Mining Methods

This chapter assessed the capability of the nonnegative matrix factorization (NMF) algorithm to address the data sparsity problem, which is one of the study's limitations. The NMF was proposed to address the data sparsity problem enabling input datasets augmentation to improve the quality and size of the datasets, thus, improving the quality and performance of the models in the proposed AI-based mining methods recommendation system (AI-MMRS). As such, we investigated the capability of the WNMF to predict missing values of mining methods. The results reveal that the WNMF model can effectively predict missing values of mining methods correctly with a moderate accuracy of 67.5%. Further findings suggest that the WNMF model is sensitive to class imbalance datasets which negatively affects the model's performance. The accuracy of 67.5% is considered reasonable and realistic, reflecting the sensibility of the model and the size and quality of the dataset (small and imbalanced dataset).

In this chapter, the WNMF algorithm is proposed as a pre-processing approach in the AI-MMRS for predicting possible missing values from the input dataset, thus enabling data augmentation and the practical implementation of the proposed system in a situation when information about some input variables is not available or accessible. Apart from predicting missing values, the WNMF model can also be used to predict and recommend underground mining methods. Therefore, with continuous improvement, the WNMF algorithm can be incorporated into developing the proposed AI-MMRS. The improvements focus on improving the initialization method and continuous data collection to improve the quality and size of the dataset.

6 Application of Machine Learning Classification Algorithms for Mining Methods Selection

This chapter aims to investigate the capability of machine learning (ML) classification algorithms to predict (classify) underground mining methods to build powerful models to be incorporated into the AI-based mining methods recommendation system (AI-MMRS). The dataset comprises mining projects' historical data with thirty-three projects described by orebody characteristics and the selected mining methods to recover the deposits.

6.1 The Need for Supervised Classification Machine Learning Algorithms

This study investigates the possibility of incorporating artificial intelligence (AI) to explore available mining projects database for developing a system that can be used as a tool to aid in decision-making when planning a mining project (i.e., the mine planning process). As such, we investigate the capability of collaborative filtering (CF) recommendation systems for developing the AI-MMRS. Collaborative filtering (CF) recommendation systems [20], [70], [71] predict users' future preferences in a collaborative way among different users. The concept of CF is that users who purchased/liked similar items in the past might likely be interested in purchasing similar items in the future. In Chapter 4, we investigated the applicability of the memory-based CF approach for underground mining methods selection (MMS) based on the k-nearest neighbours (KNN) [91], [96] algorithm with cosine similarity and the weighted sum method. Results from chapter 4 proved the effectiveness of the proposed memory-based approach in underground MMS. However, the memory-based approach is not very practical on highly sparse datasets and may be computationally expensive for large datasets. Additionally, the approach implemented in chapter 4 relies on the UBC-MMS tool [4] for generating diversified ratings for the mining methods in the dataset. The dependency on the UBC tool may affect the novelty of the proposed AI-MMRS, given that the UBC tool is an old system. In Chapter 5, we introduced a model-based CF approach [75]–[77], the nonnegative matrix factorization (NMF) algorithm. In Chapter 5, we evaluated the applicability of the NMF algorithm for addressing the data sparsity problem by predicting missing values in the dataset. Results from Chapter 5 reveal the effectiveness of the NMF in handling sparse datasets and can be effective for predicting and recommending underground mining methods. The shortcomings of the NMF as an unsupervised algorithm and the limitations memory-based approach led to the need for powerful models to be incorporated in the proposed AI-MMRS.

This chapter proposes supervised machine learning (ML) classification algorithms to address the shortcomings of the memory-based approach and NMF algorithms. We investigate the capability of different ML classification algorithms to predict (classify) underground mining methods. The dataset for this chapter is the same as applied in Chapter 3, composed of thirty-three projects, five input variables and seven underground mining methods.

6.2 Machine Learning Classifiers for Mining Methods Selection

Memory-based collaborative filtering (CF)[76], [78]–[80] is easy to use, understand and interpret mainly because there is no need for model training. However, memory-based are considered lazy learning methods because the prediction is specific and relative to the target user or item. While in the model-based [75]–[77] approach, machine learning (ML) algorithms are used to train models that automatically learn user-item interaction patterns. ML models are trained to learn the patterns/information in the datasets to map the relationship between the input variables (or predictor/independent variables) and the dependent variables (or labelled responses). ML and the model-based approach is commonly implemented through unsupervised and supervised learning environment [153]–[155]. In supervised learning, the algorithms learn from labelled datasets, while in unsupervised the algorithms learn from unlabelled datasets. Labelled datasets refer to datasets in which each sample/datapoint is tagged to a class label (or response); on the contrary, unlabelled data do not have a tag/label. Unsupervised learning algorithms are meant to discover information/patterns in the dataset without labelled responses; thus, unsupervised learning is commonly used as a data mining technique for pre-model training. In supervised learning, the algorithms get some help during the learning process. The algorithms are trained to learn patterns or relationships between input variables (independent variables) and the responses (dependent variables) to generate predictions on new datasets [33], [155], [156].

The k-nearest neighbours (KNN) and the nonnegative matrix factorization (NMF) implemented in Chapters 4 and 5 are unsupervised algorithms. In this chapter, we implement supervised classification algorithms [155]–[159]. Classification ML models are trained to predict class labels of the datapoints/samples based on the input variables [155], [160]–[162].

To incorporate ML classification algorithms in underground mining methods selection (MMS), we use the dataset used in Chapter 4. The dataset contains thirty-three projects (datapoints), five input variables (independent variables) and seven class labels (dependent variable). The five input variables correspond to the orebody characteristics in each project, such as the ore strength, host-rock strength, orebody thickness, shape, and dip. The seven class labels are the underground mining methods selected/considered in each project: block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping. We use the dataset to evaluate the capability of different ML classification algorithms (classifiers) to predict (classify) underground mining methods. For experiments, we use the classification learner application in MATLAB to train five classifiers such as decision trees, k-nearest neighbours (KNN), support vector machines (SVM), kernel approximation (SVM kernel) and artificial neural network (ANN) [34], [91], [156], [157].

6.3 Practical Experiments on the Implementation of Classification Machine Learning algorithms for Mining Methods Selection

6.3.1 Dataset

The dataset for the experiments is the same applied in Chapter 4. This dataset shows historical information on thirty-three mining projects regarding the selected or considered mining methods to recover the orebody deposits. The orebody deposits in each project are described by their orebody characteristics: geotechnical properties (ore and host-rock strength) and orebody geometry (orebody thickness, shape, and dip). The mining methods tagged to each project were selected based on the orebody characteristics. There are seven class labels/underground mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping).

6.3.2 Data Pre-processing

The pre-processing stage aims to prepare the dataset for training the selected machine learning (ML) models [163]. The original dataset is composed of variables (independent and dependent) with categorical or qualitative values. It is required to transform categorical values of the input variables to use the classification learner application. Therefore, in the pre-processing step, we only convert the categorical values of the five input variables based on their objective weights calculated using the Entropy method in Chapter 2 (the same method used in Chapters 4 and 5). The seven class labels of the dependent variable (the mining methods) will remain categorical.

6.3.3 Training ML classifiers for underground mining methods selection

The pre-processed dataset is used as input to train machine learning (ML) classifiers (classification models) for predicting (classifying) underground mining methods. Therefore, the models are trained to learn patterns or the relationship between the mining projects (datapoints) and underground mining methods (class labels/dependent variable) based on the five input variables (independent variables). The experiments are conducted in MATLAB R2022a, Statistics and Machine Learning Toolbox using classification learner application. The classification learner application allows users to train (and test) multiple classifiers on the same dataset enabling the comparison of the performance of the models side-by-side. In these experiments, we evaluate the performance of five classifiers, namely decision trees, support vector machine (SVM), k-nearest neighbour (KNN), kernel approximation, and artificial neural network (ANN) [160]–[162]. In supervised ML, it is usually required to perform model validation, also known as model assessment [164]–[166]. The purpose of the validation process is to assess the capability of the models to perform prediction on new/test data (data not used or not seen by the model during the training process); in other words, to assess the generalization capability of the model on future or unseen data [165], [166]. Therefore, the original dataset is usually split or resampled into different sets in which one of the sets is used for training and the remaining for validation and testing. The dataset splitting or resampling method depends on the type of validation method used. We evaluate/validate all five classifiers in these experiments using the k-fold cross-validation method [164].

6.3.4 Performance evaluation metrics

To measure the performance of the classifiers, we use classification metrics such as global accuracy rate and decision support metrics from the confusion matrix as described below:

The most common and straightforward metric for classification problems is the accuracy score [167], referred to as the global accuracy rate (GAR) in this study which represents the ratio of correct predicted datapoints (projects) to the total datapoints (projects) as shown in equation (6.1).

$$GAR(\%) = \frac{number\ of\ correct\ classifications}{total\ number\ of classifications} \times 100 \qquad (6.1)$$

To further evaluate the performance of the classifiers and get a more insightful performance evaluation, we use decision support metrics from the confusion matrix [160], [167] shown in Table 6.2:

Table 6.1: Confusion matrix for evaluating the performance of the classifiers

	Predicted Positive	**Predicted Negative**
Actual Positive	True Positive (TP)	False Negative (FN)
Actual Negative	False Positive (FP)	True Negative (TN)

From the confusion matrix in Table 6.2, decision support metrics such as Precision or Positive Predictive Value (PPV), Recall or True Positive Rate (TPR) and F1-score can be computed as shown in Equations (6.2) to (6.4):

$$Precision = PPV = \frac{TP}{TP+FP} \qquad (6.2)$$

$$Recall = TPR = \frac{TP}{TP+FN} \qquad (6.3)$$

$$F1 - score = 2 \times \frac{Recall \times Precision}{Recall + Precision} \qquad (6.4)$$

6.4 Experimental Results on the Implementation of Machine Learning Classification Algorithms for Mining Methods Selection

In this chapter, we evaluate the capability of machine learning (ML) classifiers (classification algorithms) for predicting (classifying) underground mining methods for developing the proposed AI-based mining methods recommendation system (AI-MMRS). The input dataset comprises thirty-three projects (datapoints), five input variables (independent variables) and seven class labels (dependent variable). The input variables correspond to five MMS influencing factors: ore strength, host-rock strength, orebody thickness, shape, and dip. The class labels are the seven underground mining methods selected or considered in each project to recover the orebody deposit: block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping. The experiments were conducted in MATLAB R2022a, Statistics and Machine Learning Toolbox, using the classification learner application. In the classification learner application, we trained multiple classifiers simultaneously on the same dataset, thus, being possible to compare them side-by-side and choose the best model. The five classifiers are decision tree, support vector machine (SVM), k-nearest neighbours (KNN), kernel approximation, and artificial neural network (ANN). We used the 10-fold cross-validation for validating all five classifiers. The k-fold cross-validation method is known to perform better on small datasets (datasets with limited datapoints) compared to other validation methods [164], [168]. In this validation method, the original dataset is randomly split into k number of folds (or subsets) of roughly equal sizes where each fold gets the chance to appear in training and testing sets [165], [169]. As shown in Figure 6.2, the total datapoints in the original dataset are split into 10-fold, each fold with roughly three datapoints (total datapoints of 33 divided by 10). The k-fold cross-validation ensures that every datapoint in the dataset can appear in training and testing sets, resulting in models that learn well and evenly the underlying patterns, creating more generalized and less biased models [166].

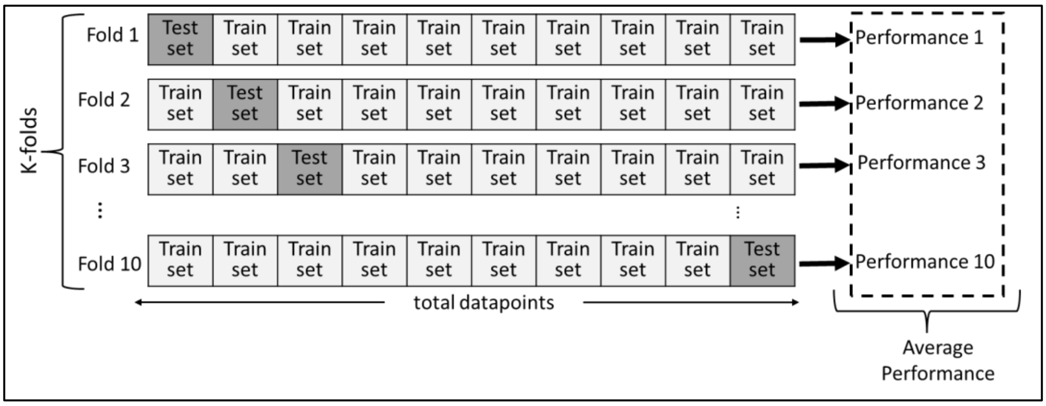


Figure 6.1: K-fold cross-validation scheme used to assess the performance of the five classifiers: 10-fold cross-validation was used to evaluate all classifiers

6.4.1 Prediction (classification) of underground mining methods

The results of the performance of the five classifiers (classification models) in terms of the global accuracy rate (GAR) are shown in Figure 6.2. The GAR is the percentage of projects (datapoints) correctly classified by a classifier. The results in Figure 6.2 shows that among the five classifiers, artificial neural network (ANN) has the best performance with 66.7% accuracy, which is followed by the k-nearest neighbours (KNN) and support vector machine (SVM), both with 63.6% accuracy. Decision tree and kernel classifiers have the lowest performance at 57.6% and 54.5%, respectively.

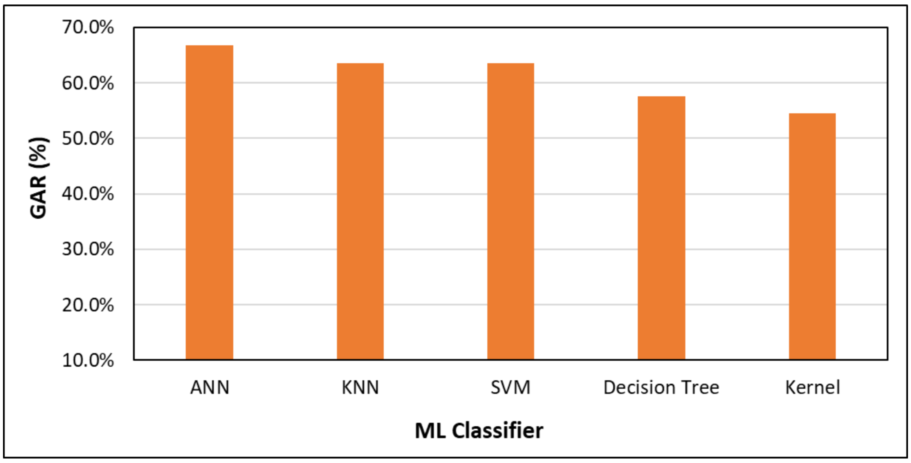


Figure 6.2: Prediction (classification) performance of the five classifiers in terms of global accuracy rate (GAR)

The results in Figure 6.2 highlight the ANN's effectiveness in mining methods selection (MMS). Previous studies [14]–[17] have investigated the applicability of the ANN in MMS and proven its effectiveness in solving the complexity of MMS. These results further highlight the effectiveness of the KNN algorithm for underground MMS. As a recap, the KNN was used in Chapter 4 for implementing the memory-based CF approach in MMS. Chapter 4 showed that the KNN-cosine similarity algorithm produced solid and effective results for predicting and recommending underground mining methods. Furthermore, the support vector machine (SVM) classifier is shown to be as effective as the KNN classifier in this chapter.

To further grasp the performance of the five classifiers in predicting (classifying) underground mining methods, we use the decision support metrics such as Precision or Positive Predictive Value (PPV), Recall or True Positive Rate (TPR) and the F1-score. The Precision (PPV) and Recall (TPR) are used to evaluate the per-class performance of the classifiers: to understand how the classifiers perform to predict projects (datapoints) in each mining method (each class label) or how the classifiers learn the underlying patterns in each mining method. The F1-score combines and balances the Precision and Recall rates [160], [167]. Table 6.3 shows the results of the performance of the classifiers in terms of the global accuracy rate (GAR), average per class of Precision, Recall and F1-score. The Precision, Recall and F1-score are first calculated for each mining method (each class) and then averaged.

Table 6.2: Performance evaluation results of the classifiers in terms of global accuracy rate (GAR), average per-class Precision, Recall and F1-score

Classifier	**GAR (%)**	**Average per-class Precision (%)**	**Average per-class Recall (%)**	**Average per-class F1-score (%)**
ANN	66.7	19.0	22.9	20.6
KNN	63.6	19.3	22.0	20.3
SVM	63.6	20.5	22.0	21.2
Decision Tree	57.6	16.6	20.2	18.2
Kernel	54.5	14.4	18.2	16.0

Usually, we expected to have close GAR and F1-score reflecting the performance of the models. Table 6.3 shows a big gap between the GAR and the average per-class F1-score of each classifier. This situation may be caused by the intrinsic characteristics of the dataset used to train the classifiers. The original dataset comprises thirty-three projects (datapoints) and seven mining methods (class labels). In Figure 6.3, we show the distribution of the projects (datapoints) in each mining method (class label). As is notable in Figure 6.3, the original dataset is class imbalance. Class imbalanced datasets happen when the distribution of datapoints among the classes is uneven, with majority and minority classes [113], [114], [136], [139]. In our dataset, we have sublevel stoping and cut and fill as the majority classes and the remaining mining methods as minority classes. Training machine learning (ML) models on class imbalance datasets have been a hot topic and the subject of several studies in different fields. It is believed that imbalanced datasets negatively affect the performance of most ML models. ML models trained from imbalanced datasets are said to be biased towards the majority class; such models will overlearn the patterns of the majority class, ignoring the minority class [113], [115], [137], [139]. In this situation, the majority classes will have higher Recall or TPR, and most of the time, the minority classes tend to have recall rates close to 0%. Thus, resulting in a lower average per-class Recall, Precision, and F1-score, as we can observe in our results in Table 6.2. The big gap between the GAR and the F1-score of the classifiers may imply that the models overlearn from majority classes (sublevel stoping and cut and fill), resulting in high GAR; however, the models perform poorly in predicting (classifying) the minority classes.

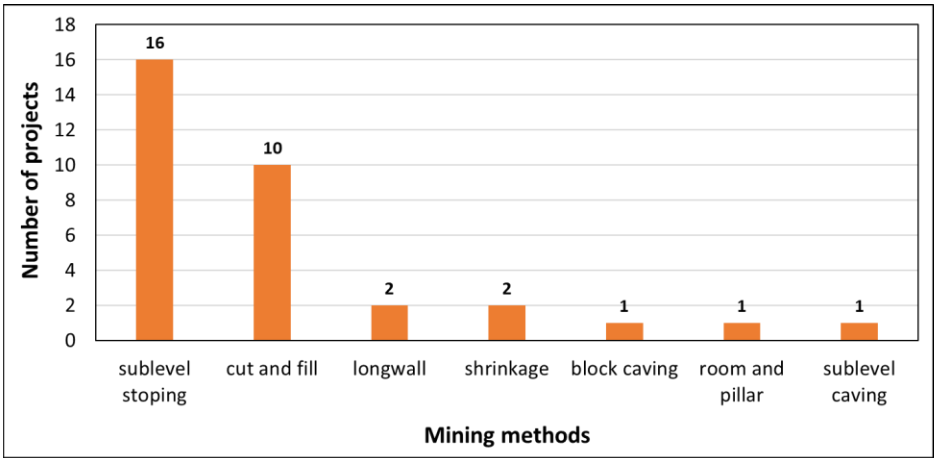


Figure 6.3: Distribution of projects (datapoints) in each mining method (or each class)

To clearly understand the impact of class imbalance datasets, we present the confusion matrix of the model with the best performance, the artificial neural network (ANN) model, in Figure 6.4. The confusion matrix in Figure 6.4 shows each class's (mining methods) Recall or TPR and the True Negative Rate (TNR). We can observe from the TPR that the model is very good at predicting the majority classes, sublevel stoping: 100%

and cut and fill: 60%; however, the model performs poorly at predicting the minority classes (block caving, longwall, room and pillar, and sublevel caving), all with 0% TPR. The low TPR (Recall) on the minority classes will reflect on the average per-class Recall and the F1-score, resulting in a low F1-score, as shown in Table 6.2.

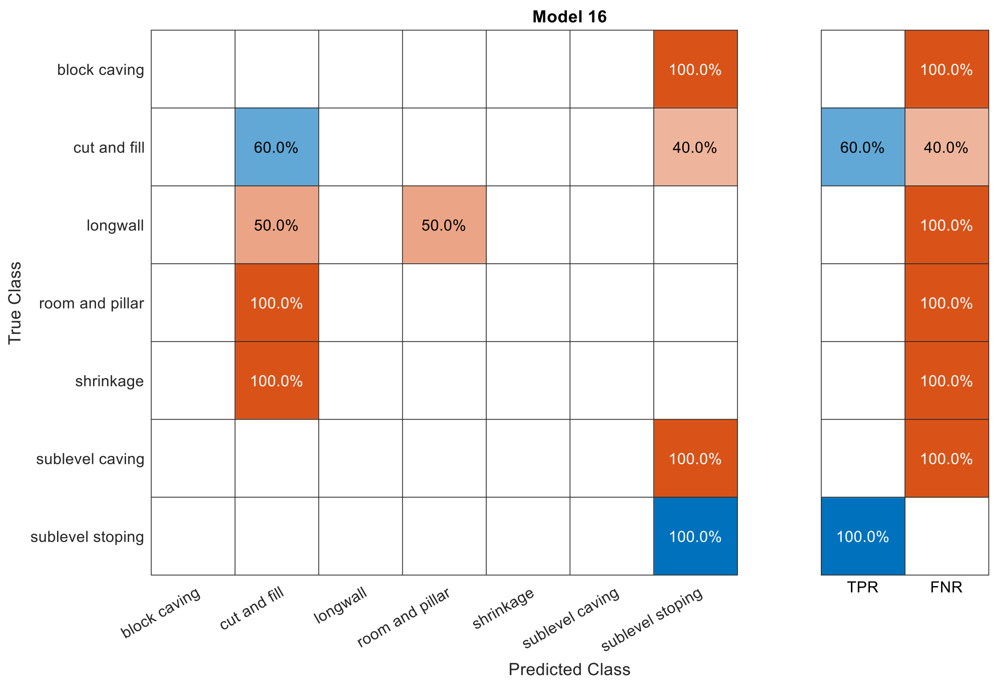


Figure 6.4: Confusion matrix of the artificial neural network (ANN) model showing the per-class Recall or True Positive Rate (TPR) and the True Negative Rate (TNR)

To further understand the effect of the class imbalanced dataset problem, we compare the performance of the five classifiers on imbalanced and balanced class datasets.

6.4.2 Effect of class imbalance/balance datasets

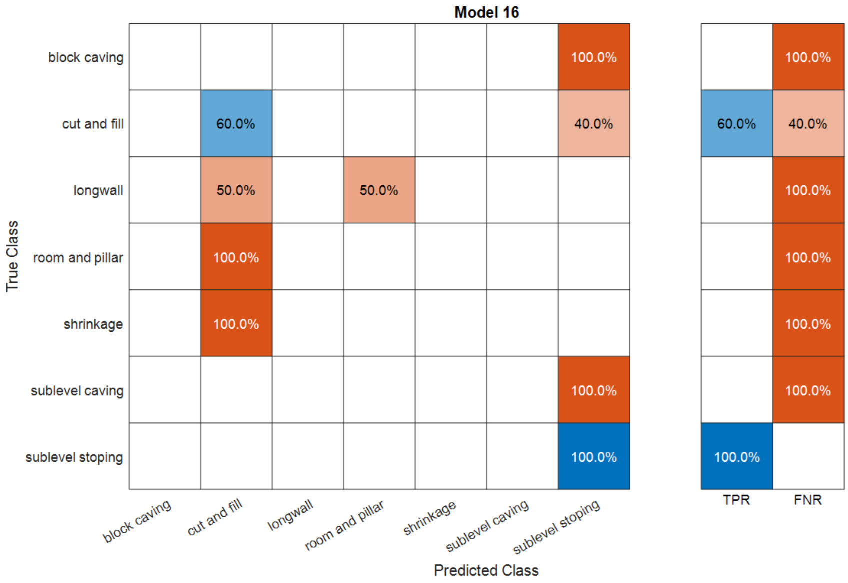
To evaluate the effect of class balance/imbalance on the performance of the classifiers, we train the same five classifiers on a balanced class dataset and then compare the results. As such, we used one of the oversampling methods [136], [170] to deal with the issue of imbalanced datasets by creating the new augmented dataset by generating synthetic datapoints using the conditional tabular generative adversarial network (CTGAN) in the synthetic data vault package. CTGAN is a collection of deep learning-based synthetic data generators for tabular data that can learn from the distribution of real/original data and generate synthetic datapoints [171]. Therefore, to produce new datapoints, the CTGAN model is first trained using a real dataset (described in Figure 6.3), and then use the trained model to produce new synthetic datapoints. The new synthetic datapoints produced using CTGAN are also submitted to the validation process (done in chapter 3) using the UBC tool [4] to avoid having meaningless and inaccurate datapoints in the dataset, which could affect the quality of the models. The new augmented dataset comprises 100 datapoints in each mining method (each class), resulting in a class balanced dataset comprising 700 datapoints. The augmented dataset trains the five classifiers under the same evaluation method (10-fold cross-validation). Table 6.4 compares the performance of the classifiers on the original dataset (imbalanced class dataset with 33 datapoints) and the new augmented dataset (balanced class dataset with 700 datapoints).

Table 6.3: Results comparing the performance of the five ML classifiers on original imbalanced and augmented balanced class datasets

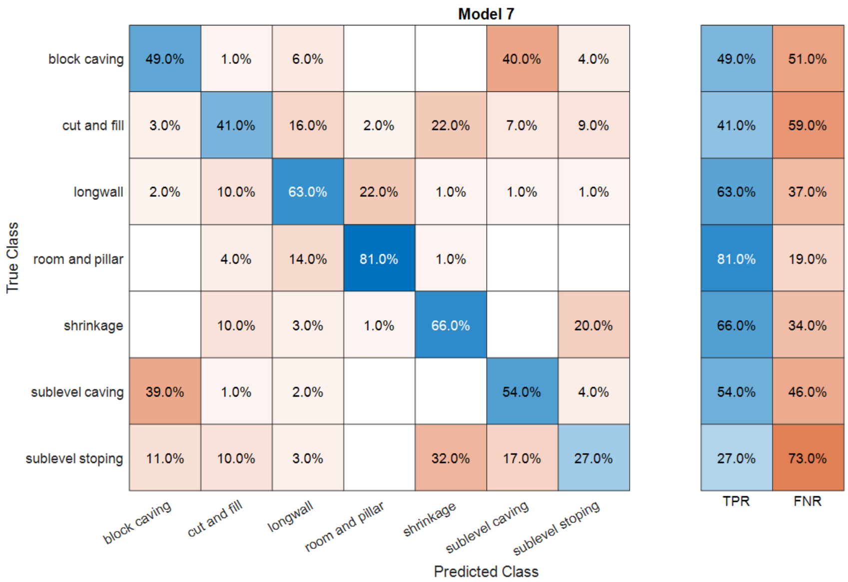
Dataset	ML classifier	GAR (%)	Average per-class Precision (%)	Average per-class Recall (%)	Average per-class F1-score (%)
Original dataset: small and imbalanced class	ANN	66.7	19.0	22.9	20.6
	KNN	63.6	19.3	22.0	20.3
	SVM	63.6	20.5	22.0	21.2
	Decision Tree	57.6	16.6	20.2	18.2
	Kernel	54.5	14.4	18.2	16.0
Augmented dataset: larger and balanced class dataset	ANN	54.4	53.8	54.4	53.6
	SVM	54.1	53.8	54.1	53.3
	Decision Tree	53.4	53.9	53.4	53.4
	KNN	53.0	51.8	53.0	51.9
	Kernel	44.9	44.3	44.9	44.4

From the results in Table 6.4, we can observe the following. Similar to the original dataset (small and imbalance class dataset), artificial neural network (ANN) scored as the best classifier on the augmented dataset (larger and balance class dataset). The support vector machine (SVM) scored as the second-best classifier, followed by the decision tree. The performance of the k-nearest neighbours (KNN) classifier is lower on the augmented dataset than on the original dataset. The GAR of five classifiers is higher on the original dataset (small and imbalanced class). Using the augmented dataset, the performance of the classifiers in terms of GAR drops slightly, as the GAR of the five models is higher on the original dataset than on the augmented dataset. However, we can observe that the values of the GAR are closer to the F1-score, thus reducing the gap between the GAR and the average per-class F1-score on the augmented dataset (larger and balance class).

Moreover, we can observe a significant improvement in the performance of the classifiers on the augmented dataset in terms of average per-class Precision, Recall and the F1-score. The improvement in the average per-class F1-score suggests a generalized/balanced and less biased performance of the classifiers to predict all seven mining methods (class labels). Figure 6.5 shows the confusion matrices of the artificial neural network (ANN) on the (a) original imbalance class dataset with 33 datapoints and (b) augmented balance class dataset with 700 datapoints. As expected, we can observe significant improvement (and balance) in the Recall or True Positive Rate (TPR) on the (a) augmented dataset. These results suggest that with a balanced dataset, the classifiers learn to map the patterns of the projects (datapoints) in each mining method (class label) evenly. Thus, they may perform significantly better predicting all seven underground mining methods (class labels).



(a) Original imbalance class dataset with 33 datapoints



(b) New augmented balance class dataset with 700 datapoints

Figure 6.5: Comparison of the performance of the artificial neural network (ANN) model showing the per-class Recall or True Positive Rate (TPR) and the True Negative Rate (TNR): (a) confusion matrix on the original imbalanced class dataset and (b) confusion matrix on the augmented balanced class dataset

These results suggest that the problem of an imbalanced class dataset negatively affects the performance of the classifiers, as has been proved in some studies [115], [137], [172]. The classifiers trained on the imbalanced class dataset are biased/overwhelmed by the majority classes (sublevel stoping and cut and fill); thus, they are very good at predicting (classifying) the majority classes but badly predict the minority classes. In these experiments, training the classifiers on the augmented dataset (larger and balance class dataset) did not improve the global accuracy rate (GAR) of the classifiers; however, it improved the average per-class F1-score. In other words, classifiers trained on the balanced dataset are less biased and more generalized because the classifiers learn better to map the patterns of every underground mining method (class label) in the dataset. Lastly, using the global accuracy rate (GAR) alone as a performance evaluation metric is not enough/efficient for supervised machine learning (ML) on imbalanced datasets. Therefore, combining the GAR with the decision support metrics from the confusion matrix is crucial for more insightful evaluations.

6.5 Discussion on the Implementation of Classification Machine Learning Algorithms for Mining Methods Selection

Mining methods selection (MMS) has been a subject of discussion and research for many years, culminating with the development of several systems. Few studies [14]–[17] have investigated the applicability of machine learning (ML) algorithms, specifically artificial neural networks (ANN), in MMS. In this chapter, we investigated the capability of five ML classification algorithms for predicting (classifying) underground mining methods. Here we evaluated the performance of classifiers such as decision trees, support vector machine (SVM), k-nearest neighbours (KNN), ANN and kernel approximation. The classifiers were trained to predict (classify) seven underground mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping) evaluated based on a dataset with thirty projects (thirty datapoints) and five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip). The ANN classifier outperformed the four classifiers with an accuracy of 66.7%, followed by the KNN and SVM with 63.6% accuracy. Decision tree and kernel classifiers had the lowest performance of 57.6% and 54.5% accuracy, respectively.

The result from this study highlights the effectiveness of the ANN in MMS, which has been proven effective in previous studies [14]–[17]. Studies by Lv and Zhihui [14] and Chen and Shixiang[15] developed ANN models for selecting the most optimum mining methods for thick and thin coal seams, respectively. The ANN model by Lv and Zhihui was based on thirty datapoints and ten input variables to predict three variables, including the optimum underground mining method. Chen and Shixiang developed the ANN model for thin coal based on thirty-three datapoints and six input variables to predict two variables, including the underground mining method. Ozyurt and Karadogan [16] further integrated ANN with game theory for selecting the most optimum underground mining methods for different ore types. Their study was based on a mixture of six different ANN models to evaluate several conditions in underground mining methods; the last ANN outputs technical scores of eleven mining methods ranked based on safety. After that, they use the ultimatum game theory to select the most optimum mining method, i.e., players (decision-makers) have a task to select the mining method (s) that satisfies safety and economy. Their ANN models were trained based on synthetic data and tested using real data, requiring about nineteen input variables in total for the evaluation and ultimately selecting the most feasible mining method. The study by Ozyurt and Karadogan further proved the effectiveness of ANN in solving the complexity of MMS. In this study, we implement classification ANN entirely evaluated based on mining projects' data, whilst, in the previous studies, the ANN [14]–[17] is implemented for regression problems. Furthermore, our study demonstrates that the ANN classifier is powerful for MMS using minimum required input variables (only five input variables) despite using limited datapoints (small dataset).

Our results further highlight the effectiveness of the KNN algorithm, which was proven effective for predicting and recommending top-3 underground mining methods with good accuracy, as was observed from the results in Chapter 4 of this study. The results in this Chapter further reveal the capability of the SVM to solve the complexity of MMS.

In this study, it was also demonstrated that training the classifiers on class imbalanced class datasets can negatively affect the performance of the classifiers, a fact that has been stated/shown in other studies [113], [115], [136], [138], [172], [173]. The results in this chapter demonstrated that classifiers based on the imbalanced class dataset are biased to have better performance on predicting (classifying) majority classes and low performance on minority classes, resulting in more biased and less generalized classifiers. On the other hand, building the classifiers using the balanced class dataset will produce less biased and more generalized, able to learn/map the patterns of each mining method (class label) evenly.

Given that the five classifiers were originally trained on a limited dataset (small and imbalanced class) for common ML algorithms, the accuracy of the classifiers (ANN: 66.7%, KNN: 63.6% and SVM: 63.6%) is considered reasonable and realistic. Therefore, we can state that the ANN, KNN and SVM are powerful algorithms for underground mining methods selection and must be considered for further optimization and implementation in the proposed AI-based mining methods recommendation system (AI-MMRS).

6.6 Summary of Implementing Classification Machine Learning Algorithms for Mining Methods Selection

In this chapter, we evaluated the capability of machine learning classification algorithms for predicting (classifying) underground mining methods for developing the proposed AI-based mining methods recommendation system (AI-MMRS). As such, we evaluated the performance of five classifiers (classification algorithms), namely, artificial neural network (ANN), k-nearest neighbours (KNN), support vector machines (SVM), decision trees and kernel approximation (kernel). The classifiers were originally evaluated based on a dataset comprising thirty-three projects, five input variables (ore strength, host-rock strength, orebody thickness, shape, and dip) and seven mining methods (block caving, cut and fill, longwall, room and pillar, shrinkage, sublevel caving and sublevel stoping). We evaluate the five classifiers' capabilities to predict (classify) the seven underground mining methods accurately. The results reveal that the ANN, KNN and SVM are the three most effective classifiers for scoring the best global accuracy rates of 66.7%, 63.6% and 63.6%, respectively. Given that the five classifiers were originally trained on a limited dataset (small and imbalance class) for common ML algorithms, the accuracy of the models is considered reasonable and realistic.

In this chapter, we further evaluated the effect of class imbalanced/balanced datasets by comparing the performance of the five classifiers on the imbalanced class dataset (the original dataset) and a balanced class dataset (an artificially augmented dataset). The results demonstrated that training the classifiers on a balanced class dataset would improve the performance of the classifiers, thus, building less biased and more generalized classifiers that effectively learn to predict (classify) better every mining method in the dataset. Therefore, suggesting the need to improve the quality of the datasets toward creating larger and more balanced class datasets, which can be achieved by continuous data collection.

Finally, it can be concluded that the machine learning (ML) classification algorithms, specifically the ANN, KNN and SVM, are powerful algorithms for mining methods selection (MMS). Therefore, the algorithms can be implemented in the proposed AI-MMRS, which can be customized and optimized as the size and quality of the dataset improves.

7 Study’s Conclusion, Significance of the Proposed AI-MMRS and Study’s Contribution

7.1 Study’s Conclusion

Undoubtedly, the success of a mining project heavily depends on the feasibility of the adopted mining method (s) to recover the orebody deposit from the earth safely and efficiently. Thereby making the mining methods selection (MMS) one of the most critical decision-making tasks in mine planning. This study introduced the application of recommendation system technologies in the MMS process by incorporating artificial intelligence (AI) to explore available mining projects database. The general aim of the study was to develop an AI-based mining methods recommendation system (AI-MMRS) by filtering (extracting) useful information from mining projects' historical data. As such, this study evaluated the applicability of the collaborative filtering (CF) recommendation system approach in underground MMS.

Chapter 1 provided an overview of MMS and a discussion of the existing MMS systems, including the current trend of AI and machine learning (ML) applications to solve the complexity of the MMS process. This chapter also provides an overview of recommendation systems and their common types and applications. The purpose and concept of this study are also described in this chapter.

Chapter 2 aimed at objectively determining the weights and the most relevant factors in MMS without the direct involvement of decision-makers (mining engineering professionals). In this chapter, the Entropy method was applied to assess the relative importance of twenty factors influencing the MMS process by determining their objective weights. The results suggested that ore strength, host-rock strength, orebody thickness, orebody shape, orebody dip, ore uniformity, mining costs, and the dilution of the mining methods are the most relevant factors in MMS. These findings are strongly supported by literature and technological and scientific advancement in selecting and implementing mining methods. Using the Entropy method to estimate the objective weights of the factors eliminate the need for decision-makers' opinions and judgement, thereby avoiding a certain level of bias associated with their subjective (customized) opinions and making the results from this chapter applicable to any case study. The results from this chapter were used as a foundation to create the input datasets in Chapter 3 and incorporated into the pre-processing data stages of Chapters 4, 5 and 6.

Chapter 3 aimed to provide an overview of the study's data explaining the procedures from data collection to creating the input datasets used to evaluate the models for the proposed AI-MMRS. The main data source is the *SEDAR* database (literature review as a secondary source), from which thousands of mining projects' technical reports were collected. Two input datasets were created by filtering information in the technical reports about the five input variables selected in Chapter 2: ore strength, host-rock strength, orebody thickness, shape, and dip. It was observed and stated that this study's main limitation is the “data sparsity problem” caused by a lack of information about the required input variables in some technical reports. The “data sparsity problem” forced the reduction of the quality of the input datasets resulting in small datasets (and imbalanced) with thirty and thirty-three projects. The first dataset comprises thirty projects with five underground mining methods (block caving, cut and fill, room and pillar, shrinkage, and sublevel stoping). The second comprises thirty-three, with all seven most common underground mining methods (sublevel stoping, shrinkage, room and pillar, cut and fill, longwall, block caving, and sublevel caving). The two datasets were used to evaluate the models in Chapters 4, 5 and 6.

Chapter 3 also reviewed the trend in the most preferred underground mining methods and the possible change in orebody conditions-mining methods in the late 2000s based on the database from *SEDAR*. Sublevel stoping is the most preferred mining method, followed by the cut and fill, room and pillar, shrinkage, and longwall. Block and sublevel caving seem the least preferred lately, probably due to the associated negative environmental impacts (destroying the surface). Square set stoping and stull stoping methods seem to be in extinction because they were not observed in the database. In terms of the change of orebody conditions of the preferred mining methods, most mining methods are selected for orebody conditions following the existing literature. However, there is a noticeable change in some conditions for some methods, which can be attributed to technological advancement, enabling easier and more flexible mechanisation and better support systems, thus improving the versatility of most mining methods. These results update the literature on the MMS discipline.

The aim of **Chapter 4** was to investigate the applicability of the memory-based collaborative filtering (CF) approach to predicting and recommending top-N most relevant underground mining methods. The memory-based approach was implemented using the KNN-cosine similarity algorithm combined with the weighted sum method. The KNN-cosine similarity was used to compute similarities among the projects and find a set of k-nearest projects (projects with similar input variables). Then the set of k-nearest projects was used for predictions and recommendations using the weighted sum method.

With the aid of the UBC-MMS tool, it was possible to design a diversified model to recommend not just one but the top-3 most relevant mining methods for a target project (one primary and two secondary methods). Therefore, it was possible to evaluate the capability model to predict and recommend the primary mining methods accurately and evaluate the quality of the top-3 recommended mining methods (one primary and two secondary methods) generated by the model. The results showed that the model accurately predicts the projects' primary mining methods with an accuracy of 63.8%. Furthermore, the model performs even better in predicting and recommending the top-3 relevant mining methods, with an accuracy of 87.9%. These results revealed the effectiveness of the memory-based approach for selecting and recommending top-N underground mining methods (by providing only five input variables), thus, applicable in developing the AI-MMRS.

Chapter 5 aimed to assess the capability of the nonnegative matrix factorization (NMF) algorithm to address the "data sparsity problem", which is one of the study's limitations, as stated in Chapter 3. The NMF was proposed to address the data sparsity problem to enable the augmentation of the input datasets to improve the quality and size of the datasets, thus, improving the quality and performance of the models. As such, Chapter 5 investigated the capability of the WNMF to predict missing values of mining methods. The results reveal that the WNMF model can accurately predict missing values with a moderate accuracy of 67.5%. Further findings suggest that the WNMF model is sensitive to class imbalance datasets which negatively affects the model's performance. The accuracy of 67.5% is considered reasonable and realistic, reflecting the sensibility of the model and the size and quality of the dataset (small and imbalanced dataset).

With further improvement and optimization (continuous data collection and model optimization), the WNMF model can be implemented to address the "data sparsity problem" (one of the limitations of this study), thus enabling data augmentation (improving the quality of the input datasets) and the practical implementation of the proposed AI-MMRS system in situations when information about some input variables is not available or accessible.

Chapter 6 investigated the capability of machine learning (ML) classification algorithms to predict (classify) underground mining methods. As such, five classifiers (classification algorithms) were evaluated, namely, artificial neural network (ANN), k-nearest neighbours (KNN), support vector machines (SVM), decision trees and kernel approximation (kernel). This Chapter evaluated the five classifiers' capabilities to predict (classify) accurately seven underground mining methods. The results reveal that the ANN, KNN and SVM are the three most effective classifiers for scoring the best global accuracy rates of 66.7%, 63.6% and 63.6%, respectively. Given that the five classifiers were originally trained on a limited dataset (small and imbalance class) for common ML algorithms, the accuracy of the models is considered reasonable and realistic.

Chapter 6 further evaluated the effect of class imbalanced/balanced datasets by comparing the performance of the five classifiers on the imbalanced class dataset (the original dataset) and a balanced class dataset (an artificially augmented dataset). The results demonstrated that training the classifiers on a balanced class dataset would improve the performance of the classifiers, thus, building less biased and more generalized classifiers that effectively learn to predict (classify) better every mining method in the dataset. Therefore, suggesting the need to improve the quality of the datasets toward creating larger and more balanced class datasets, which can be achieved by continuous data collection.

In summary, this study introduced the recommendation systems approach in the mining methods selection (MMS) discipline by implementing the collaborative filtering approach to develop a system that recommends the most appropriate underground mining methods by learning from previous mining projects' procedures. The introduction of the recommendation systems approach in MMS possesses benefits associated with efficiency and the potential to learn from past experiences. This study proposed and evaluated different machine learning (ML) models for developing the AI-MMRS. Most evaluated models can effectively predict underground mining methods with moderate accuracy, which is considered realistic given the limitation associated quality of the input datasets (small size and imbalanced datasets). Despite the limitations, the findings from this study demonstrated that the proposed AI-MMRS can be viable and practical for MMS. Continuous data collection and model optimization are required to improve the recommendations, thus building a robust system.

7.2 Significance of the Proposed AI-MMRS and Study's Contribution

This study investigated the possibility of incorporating artificial intelligence (AI) to explore available mining projects database to develop a system that can aid in decision-making in mining project development (mine planning). As such, the study introduced the application of the recommendation systems approach in the mining methods selection (MMS) discipline to propose an AI-based mining methods recommendation system (AI-MMRS) by filtering useful information from mining projects' historical data. The results from this study demonstrated that it is possible to incorporate AI for filtering (extracting) useful information from mining projects database for developing an effective MMS system. Furthermore, this study integrated different strategies attempting to address the challenges/gaps in the previous MMS systems and, in doing so, made important contributions to mining in general, especially to the MMS discipline.

The outcomes from this study generate important knowledge that contributes to the AI and ML community and, most importantly, the literature and scientific advancement in the MMS discipline, thus suggesting new directions for MMS. Furthermore, contributes to the integration of MMS in Mining informatics (Smart mining or Mining 4.0), a new discipline in Mining Engineering integrating ICT (information communication technologies) in mining to create new technologies to optimize efficiency, safety, and productivity. Through this study, the following achievements were possible:

- This study proposed, designed (and developed) an effective methodology for incorporating AI to explore available mining projects database for developing an AI-MMRS to aid in the mine planning decision-making process. Following the stages/steps in this study, it is possible to implement the same approach for different scenarios (different datasets or case studies). This study is among the first to introduce and implement the recommendation system approach in the MMS discipline, making a significant contribution to the literature and scientific advancements, thus, updating the literature and suggesting new directions for the MMS discipline.

- This study assessed and determined the relative importance of the factors influencing MMS without the direct involvement of decision-makers (i.e., subjective opinions or judgement from mining engineering professionals). Therefore, avoiding a certain level of bias associated with decision-makers' subjective (customized) opinions and conflicting views about the influencing factors and thus producing generalized results. The absence of subjective (customized) judgement makes the outcomes from this study practically applicable to any case study assisting on practical mining project decision-making and scientific studies (and research) on MMS when the opinion (or judgement) from decision-makers is unavailable (partially or totally) or even not required.
- This study designed and demonstrated an effective methodology to implement a memory-based collaborative filtering recommendation system approach in MMS to recommend a set of top-N relevant mining methods by measuring the similarity among mining projects in terms of orebody deposit characteristics. We also proved the effectiveness of ML classification algorithms in addressing the complexity of MMS tasks. Using ML classifiers, we can build a robust system that can be optimized based on specific datasets to make future predictions about underground mining methods. The proposed system only requires five input variables (ore and host rock strength, thickness, shape, and dip) which are less complex and easily accessible, especially during the early stages of mining project development (mine planning). This outcome will advance scientific studies (research) in the MMS discipline, thereby suggesting new directions and opening the doors for further future studies on the application of AI and ML to solve the complexity of MMS problems.
- Acknowledging the limitations of getting detailed information about the required input variables during the early stages of mining project development causing “data sparsity problem”. This study proposed a methodology to address this issue to predict possible missing values about required input variables for MMS. The proposed methodology can be implemented in different ML problems as a pre-

processing data method to solve data sparsity problems. This strategy can be practically used to predict information on the orebody characteristics for mining projects that do not have access to detailed information about the orebody during the early stages of mining project development (mine planning). Therefore, enabling the practical implementation of the proposed system even if there is a lack of some data about the input variables.

- Based on the study's database and the searching system in the document management software, our integrated system can be used as a practical tool to aid in decision-making on mining project investment by providing additional information (in the technical reports) about local and regional adjacent mining projects regarding common mining methods, geological and geotechnical conditions, local labour cost, environmental concerns, available infrastructure, and mineral processing procedures, among others.
- Since the system is developed based on data from mining projects dating from the 2000s, the system aids the benefit of providing up-to-date solutions following the change in factors-mining methods classification and scientific and technological advancement.
- This study also reviewed the most (and least) preferred underground mining methods and the possible change in orebody conditions-mining methods in the late 2000s based on the study's database (from *SEDAR*). This way, reviewing the impact of the scientific and technological advancement associated with the change of orebody conditions in which underground mining methods are implemented. Thus, updating the literature on the MMS discipline.

Lastly, the proposed AI-MMRS can be practically implemented in academia as a teaching or learning tool in mining engineering education. Furthermore, with the cooperation with mining projects (mining industry), the proposed system can be practically validated (tested) and implemented in mine planning to recommend a set of underground mining methods to be submitted for further evaluations (economic, technological, environmental, political). The system holds the benefit of not requiring mining engineering experts to be used (as long as there is information about the required variable) and requiring minimum information about the orebody characteristics.

REFERENCES

[1] H. L. Hartman and J. M. Mutmansky, *Introductory Mining Engineering*, Secod edit. New Jersey: John Wiley & Sons, Inc., 2002.

[2] B. H. G. Brady and E. T. Brown, *Rock mechanics for Underground Mining*, Third., vol. 24, no. C. USA: Springer Science + Business Media, Inc. Kluwer Academic Publishers, 2004. doi: 10.1016/B978-0-444-99789-0.50014-5.

[3] P. G. Carter, "Selection Process for Hard-Rock Mining," in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining Engineering, Metallurgy and Exploration, Inc., 2011, pp. 357–376.

[4] L. Miller-Tait, R. Pakalnis, and R. Poulin, "UBC Mining Method Selection," *Mine Planning and Equipment Selection (MPES)*, pp. 163–168, 1995.

[5] D. E. Nicholas, "Method Selection- A numerical Approach," in *Design and Operation of Caving and Sublevel Stoping Mines*, Illustrated., D. R. Stewart, Ed. New York: Society of Mining Engineers of the American Institute of Mining, Metallurgical and Petroleum Engineers (SME-AIME), 1981, pp. 39–53. Accessed: Jan. 16, 2023. [Online]. Available: https://www.cnitucson.com/publications.html

[6] M. Ataei, H. Shahsavany, and R. Mikaeil, "Monte Carlo Analytic Hierarchy Process (MAHP) approach to selection of optimum mining method," *Int J Min Sci Technol*, vol. 23, no. 4, pp. 573–578, Jul. 2013, doi: 10.1016/J.IJMST.2013.07.017.

[7] M. K. Bitarafan and M. Ataei, "Mining method selection by multiple criteria decision making tools," *Journal of The South African Institute of Mining and Metallurgy*, vol. 104, no. 9, pp. 493–498, 2004.

[8] A. Azadeh, M. Osanloo, and M. Ataei, "A new approach to mining method selection based on modifying the Nicholas technique," *Applied Soft Computing Journal*, vol. 10, no. 4, pp. 1040–1061, 2010, doi: 10.1016/j.asoc.2009.09.002.

[9] M. Ataei, M. Jamshidi, F. Sereshki, and S. Jalali, "Mining method selection by AHP approach Some applications of AHP in mining Engineering area," *The Journal of The Southern African Institute of Mining and Metallurgy*, vol. 108, no. April 2007, pp. 741–749, 2008.

[10] S. Alpay and M. Yavuz, "Underground mining method selection by decision making tools," *Tunnelling and Underground Space Technology*, vol. 24, no. 2, pp. 173–184, 2009, doi: 10.1016/j.tust.2008.07.003.

[11] D. Bogdanovic, D. Nikolic, and I. Ilic, "Mining method selection by integrated AHP and PROMETHEE method," *An Acad Bras Cienc*, vol. 84, no. 1, pp. 219–233, Mar. 2012, doi: 10.1590/S0001-37652012005000013.

[12] S. Shariati, A. Yazdani-chamzini, and B. P. Bashari, "Mining method selection by using an integrated model," *International Research Journal of Applied and Basic Sciences*, vol. 6, no. 2, pp. 199–214, 2013.

[13] F. S. Namin, K. Shahriar, M. Ataee-pour, and H. Dehghani, “A new model for mining method selection of mineral deposit based on fuzzy decision making,” *SIAMM - Journal of The South African Institute of Mining and Metallurgy*, vol. 108, no. 7, pp. 385–395, 2008.
[14] W. Y. Lv and Z. H. Zhang, “Application of Thick Coal Seam Mining Method Prediction Model Based on Artificial Neural Network,” *Adv Mat Res*, vol. 962–965, pp. 242–246, Jun. 2014, doi: 10.4028/www.scientific.net/AMR.962-965.242.
[15] T. Shixiang and W. Chen, “Evolving neural network using genetic algorithm for mining method evaluation in thin coal seam working face,” *Int J Min Miner Eng*, vol. 9, no. 3, p. 228, Nov. 2018, doi: 10.1504/IJMME.2018.10017388.
[16] M. C. Özyurt and A. Karadogan, “A New Model Based on Artificial Neural Networks and Game Theory for the Selection of Underground Mining Method,” *Journal of Mining Science*, vol. 56, no. 1, pp. 66–78, Jan. 2020, doi: 10.1134/S1062739120016491.
[17] A. M. A. Shohda, M. A. M. Ali, G. Ren, J. G. Kim, and M. A. E. H. Mohamed, “Application of Cascade Forward Backpropagation Neural Networks for Selecting Mining Methods,” *Sustainability (Switzerland)*, vol. 14, no. 2, Jan. 2022, doi: 10.3390/su14020635.
[18] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds., *Recommender Systems Handbook*, First Edition. Boston, MA: Springer US, 2011. doi: 10.1007/978-0-387-85820-3.
[19] C. C. Aggarwal, *Recommender Systems: The Textbook*. Cham: Springer International Publishing, 2016. doi: 10.1007/978-3-319-29659-3.
[20] M. D. Ekstrand, “Collaborative Filtering Recommender Systems,” *Foundations and Trends® in Human–Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2011, doi: 10.1561/1100000009.
[21] F. Ricci, L. Rokach, and B. Shapira, “Introduction to Recommender Systems Handbook,” in *Recommender Systems Handbook*, First Edition., F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. New York: Springer Science+Business Media, LLC 2011, 2011, pp. 1–35.
[22] E. P. A. Manjate, Y. Ohtomo, T. Arima, T. Adachi, Y. Kawamra, and B. M. Bene, “Applying Nonnegative Matrix Factorization for Underground Mining Method Selection Based on Mining Projects Historical Data,” *International Journal of the Society of Materials Engineering for Resources*, vol. 26, no. 1/2, Mar. 2024.
[23] R. L. Bullock, “Room-and-Pillar Mining in Hard Rock,” in *SME Mining Engineering Handbook*, Third Edition., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1327–1338.
[24] J. Haptonstall, “Shrinkage Stoping,” in *SME Mining Engineering Handbook*, Third edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1347–1353.

[25] R. T. Pakalnis and P. B. Hughes, “Sublevel Stoping,” in *SME Mining Engineering Handbook*, Third edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1355–1363.

[26] G. Stephan, “Cut-and-Fill Mining,” in *SME Mining Engineering Handbook*, Third edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1365–1373.

[27] S. L. Bessinger, “Longwall Mining,” in *SME Mining Engineering Handbook*, Third edit., Darling Peter, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1399–1415.

[28] G. Dunstan and G. Power, “Sublevel Caving,” in *SME Mining Engineering Handbook*, Third edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1347–1353.

[29] C. A. Brannon, G. K. Carlson, and T. P. Casten, “Block Caving,” in *SME Mining Engineering Handbook*, Third edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 1437–1451.

[30] S. Okubo and J. Yamatomi, “Underground Mining Methods and Equipment,” in *Civil Engineering*, Volume II., vol. II, K. Horikawa and Q. Guo, Eds. Oxford: Encyclopedia of Life Support Systems (EOLSS), 2009, pp. 171–194. [Online]. Available: https://books.google.co.jp/books?id=7h9bCwAAQBAJ&lpg=PA171&ots=HJK7gWnd0x&dq=Encyclopedia of Life Support Systems (EOLSS) UNDERGROUND MINING METHODS AND EQUIPMENTS.&pg=PA171#v=onepage&q&f=false

[31] E. P. A. Manjate, M. Saadat, H. Toriya, F. Inagaki, and Y. Kawamura, “Application of Entropy Method for Estimating Factor Weights in Mining-Method Selection for Development of Novel Mining-Method Selection System,” *Journal of Sustainable Mining*, vol. 20, no. 4, p. 296, Jan. 2022, doi: 10.46873/2300-3960.1328.

[32] M. Kubat, *An Introduction to Machine Learning*, Second. Cham, Switzerland: Springer International Publishing AG, 2017. doi: 10.1007/978-3-319-63913-0.

[33] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Second. New York, NY: Springer New York, 2009. doi: 10.1007/b94608.

[34] C. C. Aggarwal, *Neural Networks and Deep Learning: The Textbook*. Cham, Switzerland: Springer International Publishing AG, 2018. doi: 10.1007/978-3-319-94463-0.

[35] C. Lucchese, C. I. Muntean, R. Perego, F. Silvestri, H. Vahabi, and R. Venturini, “Recommender systems,” *Mining User Generated Content*, pp. 287–318, 2014, doi: 10.4018/ijeei.2013100103.

[36] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, “Recommender systems survey,” *Knowl Based Syst*, vol. 46, pp. 109–132, Jul. 2013, doi: 10.1016/j.knosys.2013.03.012.

[37] L. Li, R. Hsu, and F. Lee, “Review of Recommender Systems and Their Applications,” *International Journal of Advanced Information Technologies*, pp. 63–87, 2012.

[38] B. Smith and G. Linden, “Two Decades of Recommender Systems at Amazon.com,” *IEEE Internet Comput*, vol. 21, no. 3, pp. 12–18, May 2017, doi: 10.1109/MIC.2017.72.

[39] G. Linden, B. Smith, and J. York, “Amazon.com recommendations: item-to-item collaborative filtering,” *IEEE Internet Comput*, vol. 7, no. 1, pp. 76–80, Jan. 2003, doi: 10.1109/MIC.2003.1167344.

[40] J. Davidson *et al.*, “The YouTube video recommendation system,” in *Proceedings of the fourth ACM conference on Recommender systems - RecSys ’10*, 2010, p. 293. doi: 10.1145/1864708.1864770.

[41] C. A. Gomez-Uribe and N. Hunt, “The Netflix Recommender System,” *ACM Trans Manag Inf Syst*, vol. 6, no. 4, pp. 1–19, Jan. 2016, doi: 10.1145/2843948.

[42] F. O. Isinkaye, Y. O. Folajimi, and B. A. Ojokoh, “Recommendation systems: Principles, methods and evaluation,” *Egyptian Informatics Journal*, vol. 16, no. 3, pp. 261–273, Nov. 2015, doi: 10.1016/j.eij.2015.06.005.

[43] C. C. Aggarwal, “An Introduction to Recommender Systems,” in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed. Cham, Switzerland: Springer International Publishing AG Switzerland, 2016.

[44] F. McCarey, M. Cinneide, and N. Kushmerick, “A Recommender Agent for Software Libraries: An Evaluation of Memory-Based and Model-Based Collaborative Filtering,” in *2006 IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, 2006, pp. 154–162. doi: 10.1109/IAT.2006.23.

[45] Kai Yu, A. Schwaighofer, V. Tresp, Xiaowei Xu, and H. Kriegel, “Probabilistic memory-based collaborative filtering,” *IEEE Trans Knowl Data Eng*, vol. 16, no. 1, pp. 56–69, Jan. 2004, doi: 10.1109/TKDE.2004.1264822.

[46] G. Chandrashekar and F. Sahin, “A survey on feature selection methods,” *Computers and Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014, doi: 10.1016/j.compeleceng.2013.11.024.

[47] R. C. Chen, C. Dewi, S. W. Huang, and R. E. Caraka, “Selecting critical features for data classification based on machine learning methods,” *J Big Data*, vol. 7, no. 1, 2020, doi: 10.1186/s40537-020-00327-4.

[48] V. Mabin and M. Beattie, “Mulit-Criteria Decision Analysis - Workbook Companion to VISA,” no. May, 2006.

[49] C. E. Shannon, “A Mathematical Theory of Communication,” *Bell System Technical Journal*, vol. 27, no. 4, pp. 623–656, 1948, doi: 10.1002/j.1538-7305.1948.tb00917.x.

[50] F. H. Lotfi and R. Fallahnejad, “Imprecise shannon’s entropy and multi attribute decision making,” *Entropy*, vol. 12, no. 1, pp. 53–62, 2010, doi: 10.3390/e12010053.

[51] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*, First Edit. Urbana: The University of Illinois Press, 1964.

[52] M. Şahin, “A comprehensive analysis of weighting and multicriteria methods in the context of sustainable energy,” *International Journal of Environmental Science and Technology*, vol. 18, no. 6, pp. 1591–1616, Jun. 2021, doi: 10.1007/s13762-020-02922-7.

[53] R. Al-Aomar, “A combined ahp-entropy method for deriving subjective and objective criteria weights,” *International Journal of Industrial Engineering: Theory Applications and Practice*, vol. 17, no. 1, pp. 12–24, 2010.

[54] J. I. R. Gelvez and F. A. C. Aldana, “Mining Method Selection Methodology by Multiple Criteria Decision Analysis - Case Study in Colombian Coal Mining,” pp. 1–11, 2014, doi: 10.13033/isahp.y2014.123.

[55] M. Vujicic, M. Papic, and M. Blagojevic, “Comparative analysis of objective techniques for criteria weighing in two MCDM methods on example of an air conditioner selection,” *Tehnika*, vol. 72, no. 3, pp. 422–429, 2017, doi: 10.5937/tehnika1703422v.

[56] B. C. Balusa and J. Singam, “Underground Mining Method Selection Using WPM and PROMETHEE,” *Journal of The Institution of Engineers (India): Series D*, vol. 99, no. 1, pp. 165–171, 2018, doi: 10.1007/s40033-017-0137-0.

[57] A. Karadogan, A. Kahriman, and U. Ozer, “Application of fuzzy set theory in the selection of underground mining method,” *J South Afr Inst Min Metall*, vol. 108, no. 2, pp. 73–79, 2008.

[58] R. L. Bullock, “Comparison of Underground Mining Methods,” in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 384–403.

[59] E. Bohnet, “Comparison of Surface Mining Methods,” in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 404–413.

[60] A. Nieto, “Selection Process for Soft-Rock Mining,” in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 376–384.

[61] L. Alder and S. D. Thompson, “Mining Methods Classification Systems,” in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 349–357.

[62] M. G. Nelson, “Evaluation of Mining Methods and Systems,” in *SME Mining Engineering Handbook*, Third Edit., P. Darling, Ed. New York: Society for Mining, Metallurgy, and Exploration, Inc., 2011, pp. 341–348.

[63] C. C. Aggarwal, “Knowledge-Based Recommender Systems,” in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed. Cham, Switzerland: Springer International Publishing AG Switzerland, 2016.

[64] F. Bengtsson and K. Lindblad, “Methods for handling missing values: A simulation study comparing imputation methods for missing values on a Poisson distributed explanatory variable,” Uppsala University, Uppsala, Sweden, 2020.

[65] M. Soley-Bori, “Dealing with missing data: Key assumptions and methods for applied analysis,” Boston, May 2013.

[66] H. Kang, “The prevention and handling of the missing data,” *Korean J Anesthesiol*, vol. 64, no. 5, p. 402, 2013, doi: 10.4097/kjae.2013.64.5.402.

[67] I. Chahrour and J. Wells, “Comparing machine learning and interpolation methods for loop-level calculations,” *SciPost Physics*, vol. 12, no. 6, p. 187, Jun. 2022, doi: 10.21468/SciPostPhys.12.6.187.

[68] B. Saini, V. Singh, and S. Kumar, “Information retrieval models and searching methodologies: Survey,” *Inf Retr Boston*, vol. 1, no. 2, p. 20, 2014.

[69] G. J. Kowalski, *Information Retrieval Systems: Theory and Implementation*, Third Printing., vol. Volume 1. United States of America: Kluwer Academic Publishers, 1999.

[70] R. E. Neapolitan and X. Jiang, “Collaborative Filtering,” *Probabilistic Methods for Financial and Marketing Informatics*, pp. 373–385, 2007, doi: 10.1016/B978-012370477-1.50028-1.

[71] M. D. Ekstrand, “Collaborative Filtering Recommender Systems,” *Foundations and Trends® in Human–Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2011, doi: 10.1561/1100000009.

[72] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, “Item-based collaborative filtering recommendation algorithms,” in *Proceedings of the tenth international conference on World Wide Web - WWW ’01*, Apr. 2001, pp. 285–295. doi: 10.1145/371920.372071.

[73] M. Deshpande and G. Karypis, “Item-based top- N recommendation algorithms,” *ACM Trans Inf Syst*, vol. 22, no. 1, pp. 143–177, Jan. 2004, doi: 10.1145/963770.963776.

[74] G. Karypis, “Evaluation of Item-Based Top-N Recommendation Algorithms,” in *Proceedings of the tenth international conference on Information and knowledge management-CIKM’01*, 2001, p. 247. doi: 10.1145/502585.502627.

[75] C. C. Aggarwal, “Model-Based Collaborative Filtering,” in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed. Cham, Switzerland: Springer International Publishing AG Switzerland, 2016.

[76] S. Gong, H. Ye, and H. Tan, “Combining Memory-Based and Model-Based Collaborative Filtering in Recommender System,” in *2009 Pacific-Asia Conference on Circuits, Communications and Systems*, May 2009, pp. 690–693. doi: 10.1109/PACCS.2009.66.

[77] P. H. Aditya, I. Budi, and Q. Munajat, “A comparative analysis of memory-based and model-based collaborative filtering on the implementation of recommender system for E-commerce in Indonesia: A case study PT X,” in *2016 International Conference on Advanced Computer Science and Information Systems (ICACSIS)*, Oct. 2016, pp. 303–308. doi: 10.1109/ICACSIS.2016.7872755.

[78] C. C. Aggarwal, “Neighborhood-Based Collaborative Filtering,” in *Recommender Systems: The Texbook*, C. C. Aggarwal, Ed. Cham, Switzerland: Springer International Publishing AG Switzerland, 2016.

[79] C. Desrosiers and G. Karypis, “A Comprehensive Survey of Neighborhood-based Recommendation Methods,” in *Recommender Systems Handbook*, First Edition., F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. New York: Springer Science+Business Media, LLC 2011, 2011, pp. 107–144.

[80] X. Ning, C. Desrosiers, and G. Karypis, “A Comprehensive Survey of Neighborhood-Based Recommendation Methods,” in *Recommender Systems Handbook*, Second Edition., F. Ricci, L. Rokach, and B. Shapira, Eds. New York: Springer Science+Business Media LLC, 2015, pp. 37–76. doi: 10.1007/978-1-4899-7637-6_2.

[81] A. Yamashita, H. Kawamura, and K. Suzuki, “Adaptive Fusion Method for User-based and Item-based Collaborative Filtering,” *Adv Complex Syst*, vol. 14, no. 02, pp. 133–149, Apr. 2011, doi: 10.1142/S0219525911003001.

[82] M. Hasan, S. Ahmed, Md. A. I. Malik, and S. Ahmed, “A comprehensive approach towards user-based collaborative filtering recommender system,” in *2016 International Workshop on Computational Intelligence (IWCI)*, Dec. 2016, pp. 159–164. doi: 10.1109/IWCI.2016.7860358.

[83] F. Fkih, “Similarity measures for Collaborative Filtering-based Recommender Systems: Review and experimental comparison,” *Journal of King Saud University - Computer and Information Sciences*, vol. 34, no. 9, pp. 7645–7669, Oct. 2022, doi: 10.1016/j.jksuci.2021.09.014.

[84] A. Gunawardana and G. Shani, “A Survey of Accuracy Evaluation Metrics of Recommendation Tasks,” *Journal of Machine Learning Research*, vol. 10, no. 100, pp. 2935–2962, 2009, [Online]. Available: http://jmlr.org/papers/v10/gunawardana09a.html

[85] P. Cremonesi, Y. Koren, and R. Turrin, “Performance of recommender algorithms on top-n recommendation tasks,” in *Proceedings of the fourth ACM conference on Recommender systems - RecSys ’10*, 2010, p. 39. doi: 10.1145/1864708.1864721.

[86] F. Xue, X. He, X. Wang, J. Xu, K. Liu, and R. Hong, “Deep Item-based Collaborative Filtering for Top-N Recommendation,” *ACM Trans Inf Syst*, vol. 37, no. 3, pp. 1–25, Jul. 2019, doi: 10.1145/3314578.

[87] T. Kaya and C. Kaleli, “A novel top-n recommendation method for multi-criteria collaborative filtering,” *Expert Syst Appl*, vol. 198, p. 116695, Jul. 2022, doi: 10.1016/J.ESWA.2022.116695.

[88] V. W. Anelli, A. Bellogín, T. di Noia, D. Jannach, and C. Pomo, “Top-N Recommendation Algorithms: A Quest for the State-of-the-Art,” in *Proceedings of the 30th ACM Conference on User Modeling, Adaptation and Personalization*, Jul. 2022, pp. 121–131. doi: 10.1145/3503252.3531292.

[89] I. Benouaret and S. Amer-Yahia, “A Comparative Evaluation of Top-N Recommendation Algorithms: Case Study with Total Customers,” in *2020 IEEE International Conference on Big Data (Big Data)*, Dec. 2020, pp. 4499–4508. doi: 10.1109/BigData50022.2020.9378404.

[90] A. S. Tewari, J. P. Singh, and A. G. Barman, “Generating Top-N Items Recommendation Set Using Collaborative, Content Based Filtering and Rating Variance,” *Procedia Comput Sci*, vol. 132, pp. 1678–1684, 2018, doi: 10.1016/j.procs.2018.05.139.

[91] M.-L. Zhang and Z.-H. Zhou, “ML-KNN: A lazy learning approach to multi-label learning,” *Pattern Recognit*, vol. 40, no. 7, pp. 2038–2048, Jul. 2007, doi: 10.1016/j.patcog.2006.12.019.

[92] M. Kubat, “Similarities: Nearest-Neighbor Classifiers,” in *An Introduction to Machine Learning*, Second Edition., Cham: Springer International Publishing, 2017, pp. 43–64. doi: 10.1007/978-3-319-63913-0_3.

[93] T. Anwar, V. Uma, Md. I. Hussain, and M. Pantula, “Collaborative filtering and kNN based recommendation to overcome cold start and sparsity issues: A comparative analysis,” *Multimed Tools Appl*, vol. 81, no. 25, pp. 35693–35711, Oct. 2022, doi: 10.1007/s11042-021-11883-z.

[94] M. Bahadorpour, B. S. Neysiani, and M. H. N. Shahraki, “Determining Optimal Number of Neighbors in Item-based kNN Collaborative Filtering Algorithm for Learning Preferences of New Users,” *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 9, pp. 163–167, 2017.

[95] S. Airen and J. Agrawal, “Movie Recommender System Using K-Nearest Neighbors Variants,” *National Academy Science Letters*, vol. 45, no. 1, pp. 75–82, Feb. 2022, doi: 10.1007/s40009-021-01051-0.

[96] “Collaborative Filter Based Product Recommendation System using Machine Learning KNN and Cosine Similarity over Twitter by Big Data vide Ontology,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 5, pp. 7235–7240, Oct. 2020, doi: 10.30534/ijatcse/2020/51952020.

[97] D. Parra and S. Sahebi, “Recommender Systems: Sources of Knowledge and Evaluation Metrics,” in *Advanced Techniques in Web Intelligence-2*, J. D. Velásquez, V. Palade, and L. C. Jain, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, p. 149.

[98] K. Zuva, “Evaluation of Information Retrieval Systems,” *International Journal of Computer Science and Information Technology*, vol. 4, no. 3, pp. 35–43, Jun. 2012, doi: 10.5121/ijcsit.2012.4304.

[99] A. Gunawardana and G. Shani, “Evaluating Recommender Systems,” in *Recommender Systems Handbook*, Boston, MA: Springer US, 2015, pp. 265–308. doi: 10.1007/978-1-4899-7637-6_8.

[100] G. Schröder, “Setting Goals and Choosing Metrics for Recommender System Evaluations,” in *UCERSTI 2 Workshop at the 5th ACM Conference on Recommender Systems*, Oct. 2011.

[101] G. Shani and A. Gunawardana, “Evaluating Recommendation Systems,” in *Recommender Systems Handbook*, First Edition., F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. New York: Springer Science+Business Media, LLC 2011, 2011, pp. 257–298.

[102] J. T. Hancock and T. M. Khoshgoftaar, “Survey on categorical data for neural networks,” *J Big Data*, vol. 7, no. 1, p. 28, Dec. 2020, doi: 10.1186/s40537-020-00305-w.

[103] I. Ul Haq, I. Gondal, P. Vamplew, and S. Brown, “Categorical Features Transformation with Compact One-Hot Encoder for Fraud Detection in Distributed Environment,” in *Data Mining*, 2019, pp. 69–80. doi: 10.1007/978-981-13-6661-1_6.

[104] G. Hackeling, *Mastering Machine Learning with scikit-learn*. Birmingham, United Kingdom: Packt Publishing Ltd., 2014.

[105] C. C. Aggarwal, “Evaluating Recommender Systems,” in *Recommender Systems: The Textbook*, C. C. Aggarwal, Ed. Cham, Switzerland: Springer International Publishing AG Switzerland, 2016.

[106] G. J. Kowalski, “Information System Evaluation,” in *Information Retrieval Systems: Theory and Implementation*, Third Printing., vol. Volume 1, United States of Americs: Kluwer Academic Publishers, 1999.

[107] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, “Evaluating collaborative filtering recommender systems,” *ACM Trans Inf Syst*, vol. 22, no. 1, pp. 5–53, Jan. 2004, doi: 10.1145/963770.963772.

[108] N. Craswell, “Mean Reciprocal Rank,” in *Encyclopedia of Database Systems*, M. T. LIU LING and ÖZSU, Ed. Boston, MA: Springer US, 2009, pp. 1703–1703. doi: 10.1007/978-0-387-39940-9_488.

[109] J. Tiedemann, “Comparing Document Segmentation Strategies for Passage Retrieval in Question Answering,” Dec. 2022.

[110] B. Sarwar, G. Karypis, J. Konstan, and J. Riedl, “Analysis of recommendation algorithms for e-commerce,” in *Proceedings of the 2nd ACM conference on Electronic commerce - EC ’00*, 2000, pp. 158–167. doi: 10.1145/352871.352887.

[111] I. Paryudi, “What Affects K Value Selection In K-Nearest Neighbor,” *International Journal of Scientific & Technology Research*, vol. 8, pp. 86–92, 2019.

[112] Y. Zhongguo, L. Hongqi, Z. Liping, L. Qiang, and S. Ali, “A case based method to predict optimal k value for k-NN algorithm,” *Journal of Intelligent & Fuzzy Systems*, vol. 33, no. 1, pp. 55–65, Jun. 2017, doi: 10.3233/JIFS-161062.

[113] X. Guo, Y. Yin, C. Dong, G. Yang, and G. Zhou, "On the Class Imbalance Problem," in *2008 Fourth International Conference on Natural Computation*, 2008, pp. 192–201. doi: 10.1109/ICNC.2008.871.

[114] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of Imbalanced Data:Review of Methods and Applications," *IOP Conf Ser Mater Sci Eng*, vol. 1099, no. 1, p. 012077, Mar. 2021, doi: 10.1088/1757-899X/1099/1/012077.

[115] M. Lango, K. Napierala, and J. Stefanowski, "Evaluating Difficulty of Multi-class Imbalanced Data," in *Foundations of Intelligent Systems. ISMIS 2017*, Jun. 2017, pp. 312–322. doi: 10.1007/978-3-319-60438-1_31.

[116] Y. Koren, R. Bell, and C. Volinsky, "Matrix Factorization Techniques for Recommender Systems," *Computer (Long Beach Calif)*, vol. 42, no. 8, pp. 30–37, Aug. 2009, doi: 10.1109/MC.2009.263.

[117] S. A. P. Parambath, "Matrix Factorization Methods for Recommender Systems," p. 44, 2013, [Online]. Available: http://www.diva-portal.org/smash/record.jsf?pid=diva2:633561

[118] J. L. Herlocker, J. A. Konstan, and J. Riedl, "Explaining collaborative filtering recommendations," in *Proceedings of the 2000 ACM conference on Computer supported cooperative work -CSCW '00*, 2000, pp. 241–250. doi: 10.1145/358916.358995.

[119] D. D. Lee and H. S. Seung, "Algorithms for Non-negative Matrix Factorization," no. 1, 2000.

[120] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999, doi: 10.1038/44565.

[121] M. H. Aghdam, M. Analoui, and P. Kabiri, "Application of nonnegative matrix factorization in recommender systems," in *6th International Symposium on Telecommunications (IST)*, Nov. 2012, no. November, pp. 873–876. doi: 10.1109/ISTEL.2012.6483108.

[122] B. Ren, L. Pueyo, G. ben Zhu, J. Debes, and G. Duchêne, "Non-negative Matrix Factorization: Robust Extraction of Extended Structures," *Astrophys J*, vol. 852, no. 2, p. 104, 2018, doi: 10.3847/1538-4357/aaa1f2.

[123] J. Park, J. Shin, and K. Lee, "Separation of Instrument Sounds using Non-negative Matrix Factorization with Spectral Envelope Constraints," pp. 1–9, 2018, [Online]. Available: http://arxiv.org/abs/1801.04081

[124] A. Ozerov, C. Févotte, and E. Vincent, "An introduction to multichannel NMF for audio source separation," *Signals and Communication Technology*, pp. 73–94, 2018, doi: 10.1007/978-3-319-73031-8_4.

[125] C. Févotte, E. Vincent, and A. Ozerov, "Single-channel audio source separation with NMF: Divergences, constraints and algorithms," *Signals and Communication Technology*, pp. 1–24, 2018, doi: 10.1007/978-3-319-73031-8_1.

[126] Z.-Y. Zhang, "Nonnegative Matrix Factorization: Models, Algorithms and Applications," pp. 99–134, 2012, doi: 10.1007/978-3-642-23241-1_6.

[127] M. W. Berry, M. Browne, A. N. Langville, V. P. Pauca, and R. J. Plemmons, “Algorithms and applications for approximate nonnegative matrix factorization,” *Comput Stat Data Anal*, vol. 52, no. 1, pp. 155–173, 2007, doi: 10.1016/j.csda.2006.11.006.

[128] Z. Yang, H. Zhang, Z. Yuan, and E. Oja, “Kullback-Leibler divergence for nonnegative matrix factorization,” *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 6791 LNCS, no. PART 1, pp. 250–257, 2011, doi: 10.1007/978-3-642-21735-7_31.

[129] Z. Zheng, J. Yang, and Y. Zhu, “Initialization enhancer for non-negative matrix factorization,” *Eng Appl Artif Intell*, vol. 20, no. 1, pp. 101–110, 2007, doi: 10.1016/j.engappai.2006.03.001.

[130] D. Guillamet, M. Bressan, and J. Vitria, “A weighted non-negative matrix factorization for local representations,” in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, pp. I-942-I–947. doi: 10.1109/CVPR.2001.990629.

[131] Y.-D. Kim and S. Choi, “Weighted nonnegative matrix factorization,” in *2009 IEEE International Conference on Acoustics, Speech and Signal Processing*, Apr. 2009, pp. 1541–1544. doi: 10.1109/ICASSP.2009.4959890.

[132] S. Thakallapalli, S. Gangashetty, and N. Madhu, “A new weighted NMF algorithm for missing data interpolation and its application to speech enhancement,” *European Signal Processing Conference*, vol. 2019-Septe, 2019, doi: 10.23919/EUSIPCO.2019.8903029.

[133] N. Ho, P. van Dooren, and V. Blondel, “Weighted nonnegative matrix factorization and face feature extraction,” *Image Vis Comput*, no. March 2008, pp. 1–17, 2008, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.550.2833&rep=rep1&type=pdf%0Ahttp://www.researchgate.net/publication/253588948_Weighted_Nonnegative_Matrix_Factorization_and_Face_Feature_Extraction/file/50463529d128a02565.pdf

[134] S. G. K. Patro and K. K. sahu, “Normalization: A Preprocessing Stage,” *Iarjset*, pp. 20–22, 2015, doi: 10.17148/iarjset.2015.2305.

[135] an Langville, C. Meyer, R. Albright, and J. Cox, “Initializations for the nonnegative matrix factorization,” *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 1–8, 2006, [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.4302&rep=rep1&type=pdf

[136] S. Kotsiantis, D. Kanellopoulos, and P. Pintelas, “Handling imbalanced datasets: A review,” *GESTS International Transactions on Computer Science and Engineering*, vol. 30, pp. 25–36, Dec. 2005.

[137] B. Krawczyk, “Learning from imbalanced data: open challenges and future directions,” *Progress in Artificial Intelligence*, vol. 5, no. 4, pp. 221–232, Nov. 2016, doi: 10.1007/s13748-016-0094-0.
[138] J. M. Johnson and T. M. Khoshgoftaar, “Survey on deep learning with class imbalance,” *J Big Data*, vol. 6, no. 1, p. 27, Dec. 2019, doi: 10.1186/s40537-019-0192-5.
[139] M. Sahare and H. Gupta, “A Review of Multi-Class Classification for Imbalanced Data,” *International Journal of Advanced Computer Research*, vol. 2, pp. 160–164, Dec. 2012.
[140] A. Fernández, V. López, M. Galar, M. J. del Jesus, and F. Herrera, “Analysing the classification of imbalanced data-sets with multiple classes: Binarization techniques and ad-hoc approaches,” *Knowl Based Syst*, vol. 42, pp. 97–110, Apr. 2013, doi: 10.1016/j.knosys.2013.01.018.
[141] N. Gillis, “The Why and How of Nonnegative Matrix Factorization,” 2014.
[142] A. N. Langville, C. D. Meyer, R. Albright, J. Cox, and D. Duling, “Algorithms, Initializations, and Convergence for the Nonnegative Matrix Factorization,” no. July 2015, 2014, [Online]. Available: http://arxiv.org/abs/1407.7299
[143] L. Muzzarelli, S. Weis, S. B. Eickhoff, and K. R. Patil, “Rank Selection in Non-negative Matrix Factorization: systematic comparison and a new MAD metric,” in *2019 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2019, pp. 1–8. doi: 10.1109/IJCNN.2019.8852146.
[144] S. Lee, “Estimating the Rank of a Nonnegative Matrix Factorization Model for Automatic Music Transcription Based on Stein’s Unbiased Risk Estimator,” *Applied Sciences*, vol. 10, no. 8, p. 2911, Apr. 2020, doi: 10.3390/app10082911.
[145] S. M. Atif, S. Qazi, and N. Gillis, “Improved SVD-based initialization for nonnegative matrix factorization using low-rank correction,” *Pattern Recognit Lett*, vol. 122, pp. 53–59, 2019, doi: 10.1016/j.patrec.2019.02.018.
[146] H. Qiao, “New SVD based initialization strategy for non-negative matrix factorization,” *Pattern Recognit Lett*, vol. 63, pp. 71–77, 2015, doi: 10.1016/j.patrec.2015.05.019.
[147] C. Boutsidis and E. Gallopoulos, “SVD based initialization: A head start for nonnegative matrix factorization,” *Pattern Recognit*, vol. 41, no. 4, pp. 1350–1362, 2008, doi: 10.1016/j.patcog.2007.09.010.
[148] J. Frederic, “Examination of Initialization Techniques for Nonnegative Matrix Factorization,” *Mathematics Theses*, p. 63, 2008, [Online]. Available: http://digitalarchive.gsu.edu/cgi/viewcontent.cgi?article=1062&context=math_theses
[149] D. K. Sokendai, “Efficient initialization for NMF based on nonnegative ICA,” 2016.
[150] Y. D. Kim and S. Choi, “A method of initialization for nonnegative matrix factorization,” *ICASSP, IEEE International Conference on Acoustics, Speech and*

Signal Processing - Proceedings, vol. 2, no. May 2007, 2007, doi: 10.1109/ICASSP.2007.366291.
[151] F. Esposito, "A review on initialization methods for nonnegative matrix factorization: Towards omics data experiments," *Mathematics*, vol. 9, no. 9, 2021, doi: 10.3390/math9091006.
[152] Z. Liu and V. Y. F. Tan, "Rank-One NMF-Based Initialization for NMF and Relative Error Bounds under a Geometric Assumption," *IEEE Transactions on Signal Processing*, vol. 65, no. 18, pp. 4717–4731, 2017, doi: 10.1109/TSP.2017.2713761.
[153] X. Amatriain, A. Jaimes, N. Oliver, and J. M. Pujol, "Data Mining Methods for Recommender Systems," in *Recommender Systems Handbook*, First Edition., F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. New York: Springer Science+Business Media, LLC 2011, 2011, pp. 39–71.
[154] I. H. Sarker, "Machine Learning: Algorithms, Real-World Applications and Research Directions," *SN Comput Sci*, vol. 2, no. 3, p. 160, May 2021, doi: 10.1007/s42979-021-00592-x.
[155] Aized Amin Soofi and Arshad Awan, "Classification Techniques in Machine Learning: Applications and Issues," *Journal of Basic & Applied Sciences*, vol. 13, pp. 459–465, Jan. 2017, doi: 10.6000/1927-5129.2017.13.76.
[156] X. Amatriain and J. M. Pujol, "Data Mining Methods for Recommender Systems," in *Recommender Systems Handbook*, Second Edition., F. Ricchi, L. Rokach, and B. Shapira, Eds. Boston, MA: Springer US, 2015, pp. 227–262. doi: 10.1007/978-1-4899-7637-6_7.
[157] X. Amatriain, A. Jaimes*, N. Oliver, and J. M. Pujol, "Data Mining Methods for Recommender Systems," in *Recommender Systems Handbook*, First Edition., F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Boston, MA: Springer US, 2011, pp. 39–71. doi: 10.1007/978-0-387-85820-3_2.
[158] V. Nasteski, "An overview of the supervised machine learning methods," *HORIZONS.B*, vol. 4, pp. 51–62, Dec. 2017, doi: 10.20544/HORIZONS.B.04.1.17.P05.
[159] S. M. Mirmazloumi *et al.*, "Supervised Machine Learning Algorithms for Ground Motion Time Series Classification from InSAR Data," *Remote Sens (Basel)*, vol. 14, no. 15, p. 3821, Aug. 2022, doi: 10.3390/rs14153821.
[160] J. Han, M. Kamber, and J. Pei, "Classification: Basic Concepts," in *Data Mining: Concepts and Techniques*, Third., USA: Data Mining: Concepts and Techniques, 2012.
[161] C. C. Aggarwal, "Data Classification," in *Data Mining: The Textbook*, Cham: Springer International Publishing, 2015. doi: 10.1007/978-3-319-14142-8.
[162] H. Tan, "Machine Learning Algorithm for Classification," *J Phys Conf Ser*, vol. 1994, no. 1, p. 012016, Aug. 2021, doi: 10.1088/1742-6596/1994/1/012016.

[163] J. Han, M. Kamber, and J. Pei, "Data Preprocessing," in *Data Mining: Concepts and Techniques*, Third., USA: Elsevier Inc., 2012.

[164] D. Berrar, "Cross-Validation," in *Encyclopedia of Bioinformatics and Computational Biology*, Elsevier, 2019, pp. 542–545. doi: 10.1016/B978-0-12-809633-8.20349-X.

[165] H. Wang and H. Zheng, "Model Validation, Machine Learning," in *Encyclopedia of Systems Biology*, W. Dubitzky, O. Wolkenhauer, and KH. Y. H. Cho, Eds. New York, NY: Springer New York, 2013, pp. 1406–1407. doi: 10.1007/978-1-4419-9863-7_233.

[166] T. Hastie, R. Tibshirani, and J. Friedman, "Model Assessment and Selection," in *The Elements of Statistical Learning*, Second., New York, NY: Springer New York, 2009.

[167] M. A. el Mrabet, K. el Makkaoui, and A. Faize, "Supervised Machine Learning: A Survey," in *2021 4th International Conference on Advanced Communication Technologies and Networking (CommNet)*, Dec. 2021, pp. 1–10. doi: 10.1109/CommNet52204.2021.9641998.

[168] S. Raschka, "Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning," *CoRR*, vol. abs/1811.12808, 2018, [Online]. Available: http://arxiv.org/abs/1811.12808

[169] M. Bramer, *Principles of Data Mining*. London: Springer London, 2016. doi: 10.1007/978-1-4471-7307-6.

[170] G. Lemaitre, F. Nogueira, and C. K. Aridas, "Imbalanced-Learn: A Python Toolbox to Tackle the Curse of Imbalanced Datasets in Machine Learning," *J. Mach. Learn. Res.*, vol. 18, no. 1, pp. 559–563, Jan. 2017.

[171] L. Xu, M. Skoularidou, A. Cuesta-Infante, and K. Veeramachaneni, "Modeling Tabular data using Conditional GAN," in *Advances in Neural Information Processing Systems*, 2019, vol. 32. [Online]. Available: https://proceedings.neurips.cc/paper/2019/file/254ed7d2de3b23ab10936522dd547b78-Paper.pdf

[172] M. Buda, "A systematic study of the class imbalance problem in convolutional neural networks," Master in Computer Science, KTH Royal Institute of Technology, Stockholm, Sweden, 2017.

[173] P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of Imbalanced Data:Review of Methods and Applications," *IOP Conf Ser Mater Sci Eng*, vol. 1099, no. 1, p. 012077, Mar. 2021, doi: 10.1088/1757-899X/1099/1/012077.

APPENDICES

Table A.1: Dataset used for experiments in Chapters 4 and 6

Project	Ore strength	Host-rock strength	Thickness	Shape	Dip	Mining method
PJ001	Strong	Moderate-strong	Intermediate-thick	Tabular	Steep	Sublevel stoping
PJ002	Moderate	Moderate-strong	Very narrow-narrow	Tabular	Intermediate	Longwall
PJ003	Moderate	Strong	Narrow	Tabular	Steep	Cut and fill
PJ004	Strong	Moderate-strong	Thick	Irregular	Intermediate	Cut and fill
PJ005	Weak	Moderate	Very narrow	Irregular	Steep	Cut and fill
PJ006	Moderate	Weak-moderate	Very narrow	Tabular	Flat	Room and pillar
PJ007	Strong	Moderate-strong	Intermediate	Irregular	Intermediate	Cut and fill
PJ008	Weak	Weak	Very narrow	Tabular	Flat	Longwall
PJ009	Strong	Moderate	Thick	Tabular	Steep	Sublevel stoping
PJ010	Moderate	Strong	Narrow	Irregular-tabular	Steep	Shrinkage
PJ011	Strong	Strong	Intermediate	Tabular	Flat	Cut and fill
PJ012	Moderate-strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ013	Weak	Weak	Narrow	Irregular	Intermediate	Cut and fill
PJ014	Strong	Strong	Narrow	Tabular	Intermediate	Sublevel stoping
PJ015	Very strong	Very strong	Intermediate	Tabular	Flat	Sublevel stoping
PJ016	Moderate	Moderate	Narrow	Tabular	Steep	Sublevel stoping
PJ017	Weak-moderate	Weak-moderate	Thick	Tabular	Intermediate	Sublevel stoping
PJ018	Moderate	Moderate	Narrow	Tabular	Steep	Sublevel stoping
PJ019	Moderate	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ020	Moderate	Moderate	Narrow	Tabular	Steep	Sublevel stoping
PJ021	Strong	Moderate-strong	Narrow	Tabular	Steep	Sublevel stoping
PJ022	Moderate-strong	Moderate	Thick	Massive	Flat	Block caving
PJ023	Weak	Moderate	Very narrow-narrow	Tabular	Steep	Shrinkage
PJ024	Strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ025	Moderate	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ026	Moderate-strong	Strong	Very narrow	Tabular	Steep	Cut and fill
PJ027	Moderate	Moderate	Narrow	Tabular	Intermediate	Cut and fill
PJ028	Weak	Strong	Narrow	Tabular	Flat-intermediate	Cut and fill
PJ029	Moderate-strong	Moderate-strong	Intermediate	Massive	Steep	Sublevel stoping
PJ030	Strong-very strong	Strong-very strong	Narrow	Tabular	Intermediate-steep	Sublevel stoping
PJ031	Strong	Moderate-strong	Intermediate	Tabular	Intermediate-steep	Sublevel stoping
PJ032	Moderate	Moderate-strong	Narrow	Tabular	Intermediate	Cut and fill
PJ033	Strong	Strong	Intermediate-thick	Tabular	Steep	Sublevel caving

Table A.2: Dataset used for experiments in Chapter 5

Project	**Ore strength**	**Host-rock strength**	**Thickness**	**Shape**	**Dip**	**Mining method**
PJ001	Strong	Strong	Narrow	Tabular	Intermediate	Room and pillar
PJ002	Moderate	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ003	Moderate	Strong	Narrow	Irregular-tabular	Steep	Shrinkage
PJ004	Strong	Strong	Intermediate	Tabular	Flat	Cut and fill
PJ005	Weak	Weak-moderate	Narrow	Tabular	Flat	Cut and fill
PJ006	Moderate-strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ007	Weak	Weak	Narrow	Irregular	Intermediate	Cut and fill
PJ008	Moderate-strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ009	Very strong	Very strong	Intermediate	Tabular	Intermediate	Sublevel stoping
PJ010	Weak	Weak-moderate	Narrow	Tabular	Flat	Room and pillar
PJ011	Strong-very strong	Very weak-weak	Narrow	Tabular	Intermediate	Cut and fill
PJ012	Strong	Strong	Narrow	Tabular	Steep	Sublevel stoping
PJ013	Moderate-strong	Moderate-strong	Narrow	Tabular	Intermediate	Sublevel stoping
PJ014	Moderate	Weak	Narrow	Tabular	Steep	Cut and fill
PJ015	Moderate	Weak-moderate	Narrow	Tabular	Steep	Sublevel stoping
PJ016	Weak-moderate	Weak-moderate	Thick	Tabular	Intermediate	Sublevel stoping
PJ017	Very weak-weak	Weak	Narrow	Tabular	Steep	Cut and fill
PJ018	Very weak-weak	Weak	Narrow	Tabular	Intermediate	Cut and fill
PJ019	Moderate	Moderate-strong	Narrow	Massive	Steep	Sublevel stoping
PJ020	Very weak-weak	Weak-moderate	Narrow	Tabular	Flat	Room and pillar
PJ021	Moderate	Moderate	Intermediate	Tabular	Steep	Sublevel stoping
PJ022	Strong	Strong	Narrow	Tabular	Steep	Shrinkage
PJ023	Strong	Moderate-strong	Narrow	Tabular	Steep	Sublevel stoping
PJ024	Moderate-strong	Moderate-strong	Narrow	Tabular	Steep	Sublevel stoping
PJ025	Moderate	Moderate	Narrow	Tabular	Intermediate	Cut and fill
PJ026	Weak	Strong-very strong	Narrow	Tabular	Steep	Shrinkage
PJ027	Weak	Very weak-weak	Narrow	Tabular	Steep	Cut and fill
PJ028	Strong	Moderate	Narrow	Tabular	Steep	Sublevel stoping
PJ029	Moderate-strong	Moderate	Thick	Massive	Flat	Block caving
PJ030	Weak	Moderate	Narrow	Tabular	Steep	Shrinkage