Machine Learning Assisted Identification of BCC-FCC Regime in Al-Co-Cr-Fe-Ni

M. Tech. Thesis

By

Shantanu Dilip Gagare

DEPARTMENT OF METALLURGICAL ENGINEERING AND MATERIALS SCIENCE

INDIAN INSTITUTE OF TECHNOLOGY INDORE

MAY 2024

Machine Learning Assisted Identification of BCC-FCC Regime in Al-Co-Cr-Fe-Ni

A THESIS

Submitted in partial fulfillment of the

Requirements for the award of the degree

of

Master of Technology

by

Shantanu Dilip Gagare

DEPARTMENT OF METALLURGICAL ENGINEERING AND MATERIALS SCIENCE

INDIAN INSTITUTE OF TECHNOLOGY INDORE

MAY 2024

INDIAN INSTITUTE OF TECHNOLOGY INDORE

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the thesis entitled **Machine Learning Assisted Identification of BCC-FCC Regime in Al-Co-Cr-Fe-Ni** in the partial fulfillment of the requirements for the award of the degree of **MASTER OF TECHNOLOGY** and submitted in the **DISCIPLINE OF METALLURGICAL ENGINEERING AND MATERIALS SCIENCE, Indian Institute of Technology Indore**, is an authentic record of my own work carried out during the time period from August 2023 to May 2024 under the supervision of Dr. Ram Sajeevan Maurya, Assistant Professor, IIT Indore.

The matter presented in this thesis has not been submitted by me for the award of any other degree of this or any other institute.

31-05-2024

Signature of the student with date
(SHANTANU DILIP GAGARE)

This is to certify that the above statement made by the candidate is correct to the best of my/our knowledge.

31/05/2024

Signature of the Supervisor of

M.Tech. thesis #1 (with date)

(Dr. Ram Sajeevan Maurya)

SHANTANU DILIP GAGARE has successfully given his M.Tech. Oral Examination held on **May 29, 2024**.

Signature(s) of Supervisor(s) of M.Tech. thesis
Date: 31/05/2024

Convener, DPGC
Date:

ACKNOWLEDGEMENTS

I sincerely thank all the individuals whose unswerving support and guidance have been instrumental during this whole MTech journey.

To begin with, I would like to convey my deepest gratitude to my thesis advisor, **Dr. Ram Sajeevan Maurya**, whose encouragement and direction have been foundational to this project. His consistent support has enabled me to confront challenges with determination.

I am particularly grateful to **Mr. Gopi Tallur**i for his invaluable assistance in shaping the problem statement, guiding me through every phase of the project, providing essential databases, and sharing insights that significantly contributed to its timely completion. Without his support, expertise, and completing this thesis would have been quite challenging.

I also thank **Mr. Deepak Kumar Srivastava** for his assistance and all the Amorphous Metals Laboratory (AML) group members, including Gulshan and Shrevesh, for their valuable assistance and engaging discussions.

I extend my gratitude to **Prof. Gandham Phanikumar (IIT Madras)** and **Prof. B.S. Murty (IIT Hyderabad)** for enabling access to the Thermo-Calc software. Additionally, I am thankful to **Dr. Hariharan** (IIT Madras), **Naveen L (IIT Indore)**, and Gokul M Pillai (IIT Indore) for their guidance and valuable insights.

Finally, I am deeply thankful to my family for their constant love, support, and understanding throughout this whole journey. Their encouragement and sacrifices have been a source of motivation and strength.

DEDICATION

Dedicated to

My Mother

Table of Contents

List Of Figures

List of Tables

Abbreviations

HEAs	High-Entropy Alloys
LHS	Latin Hypercube Sampling
VEC	Valence Electron Concentration
KNN	K-Nearest Neighbors
SVM	Support Vector Machine
DT	Decision Tree
ANN	Artificial Neural Network
XRD	X-ray diffraction
DSC	Differential Scanning Calorimetry
MCAs	Multicomponent Alloys
CCAs	Concentrated Alloys
CALPHAD	Calculation of Phase Diagrams
DFT	Density Functional Theory
ML	Machine Learning
AI	Artificial Intelligence
XGBoost	Gradient Boosting
IM	Intermetallic
BCC_SS	Body-Centered Cubic Solid Solution
FCC_SS	Face-Centered Cubic Solid Solution
CNN	Convolutional Neural Network
RFC	Random Forest Classifier
DNN	Deep Neural Network
LR	Logistic Regression
GB	Gradient Boosting
DSS	Duplex Stainless Steel
SVC	Support Vector Classifier
MSE	Mean Squared Error
LHS	Latin Hypercube Sampling
VEC	Valence Electron Concentration
EDA	Exploratory Data Analysis

ABSTRACT

High-Entropy Alloys (HEAs) have gathered significant attention since their first report in 2004 due to the vast compositional space offering diverse functional and mechanical properties. Quickly locating exact compositions with desired properties is essential before experimental characterization. This thesis aims to use a machine learning approach to quickly determine the compositional space of BCC, FCC, and combined BCC plus FCC phases, using the Al-Co-Cr-Fe-Ni system as a case study. A total of 75,000 compositions were generated through Latin Hypercube Sampling (LHS) and analyzed via TC-Python, resulting in 54,056 cleaned data points. Key physicochemical properties, such as mixing enthalpy ($\Delta Hmix$), Valence Electron Concentration (VEC), electronegativity difference ($\Delta\chi$), and atomic size difference (δ) were found to correlate with phase formation. Machine learning models, including Decision Tree Algorithm, Logistic Regression model, Random Forest Algorithm, Support Vector Machine Classifier (SVM), K-Nearest Neighbors Classifier (KNN), and Artificial Neural Networks (ANN), were trained to predict phase formation (BCC, FCC, and BCC+FCC). Among these, the ANN model achieved the highest F1 score of 98.17%, establishing it as the best-performing model, followed by KNN, SVM, and Random Forest. Validation against literature-reported data confirmed the ANN model's accuracy at approximately 90%. For experimental verification of the identified compositional space through the ANN model, an alloy with the specified composition of $Al_{15}Co_5Cr_5Fe_{45}Ni_{30}$ was cast. Results from Differential Scanning Calorimetry (DSC) and X-ray diffraction (XRD) corroborated the identified compositional range, validating the effectiveness of the machine-learning approach.

Chapter 1: Introduction

In the realm of advanced materials, high-entropy alloys have garnered significant interest in material science due to their unconventional compositional space, which confers superior properties compared to conventional alloys. This contradicts traditional principles, as most HEAs generally contain multiple principal elements in equiatomic or near-equiatomic ratios, with a huge increase in configurational entropy. This contradicts traditional principles and the concept proposed by Yeh *et al.* [1] and subsequently by Cantor *et al.*[2] opened the door for the advent of new alloying systems with Outstanding mechanical characteristics, notable thermal stability, and high resistance to wear, corrosion, and oxidation. The concept of HEAs relies on the principle of high mixing entropy, which promotes the formation of solid solution phases instead of intermetallic compounds. These four can be summed up as the high-entropy effect, sluggish diffusion, lattice distortion, and "cocktail" effect, which are the four core effects determining the behavior and characteristics of HEAs. The sum of the effects usually gives the excellent performance of HEAs in different fields, among which aerospace engineering comes under the call of work at a high temperature.

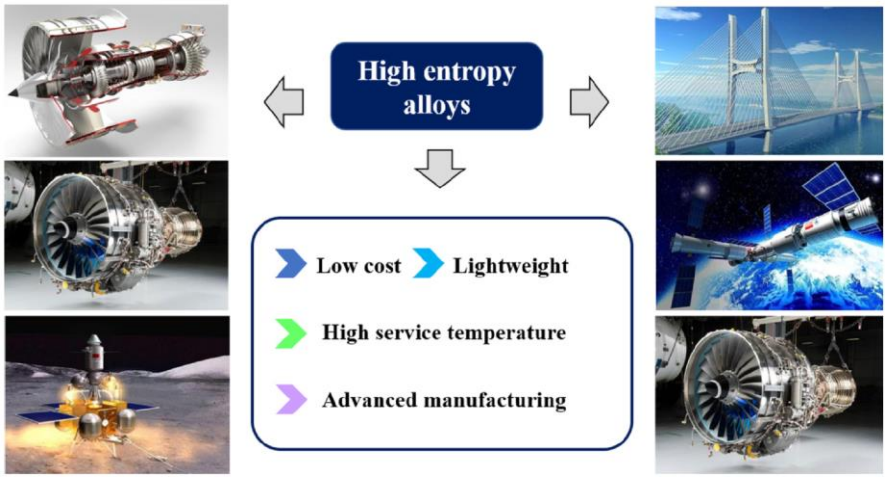

Figure 1: Applications of High Entropy Alloys [3]

High Entropy Alloys are emerging as revolutionary materials in multiple high-performance applications due to their excellent characteristics, including low cost, lightweight, high service temperature, and advanced manufacturing capabilities. These alloys are increasingly being utilized in critical sectors such as aerospace, which are used in jet engines, spacecraft components, and infrastructure projects like bridge construction. HEAs' diverse and exceptional properties make them suitable for applications that demand high strength, durability, and resistance to extreme conditions[3].

Recent advancements in high-throughput methods and machine learning have further accelerated the development of HEAs. By leveraging large datasets and advanced computational techniques, researchers can now predict the properties of new alloys and optimize their compositions more efficiently. This data-driven approach reduces the time and cost associated with experimental alloy development and expands the possibilities for discovering alloys with tailored properties for specific applications.

1.1 Definition

1.1.1 Definition based on composition

Historically, high-entropy alloys (HEAs) were defined based on their composition, typically consisting of five or more principal elements in near to equal molar ratios, where the concentration of each element ranges from 5 to 35 atomic percent (at%)[2]. However, recent studies have shown that HEAs can also exhibit promising properties with non-equimolar or fewer elements[4]. Some literature even classifies quaternary alloys as HEAs due to their multi-component composition [5–7]. This broader definition allows for minor elements and non-equimolar concentrations, thereby expanding the number of potential HEAs. Consequently, terms like multicomponent alloys (MCAs)[8], concentrated alloys (CCAs)[9], and multiple principal elements alloys (MPEAs)[10]have been proposed to describe these materials.

1.1.2 Definition based on Entropy

The concept of high-entropy alloys focuses on maximizing mixing entropy to promote the formation of solid-solution phases. The configurational entropy (*ΔSconf*) of an ideal random solid solution with n components is represented by Eq. (1). [11]:

$$\Delta Sconf = -R \sum_{i=1}^{n} c_i lnc_i \qquad (1)$$

where R is the gas constant, ci is the mole fraction of the i-th element, and n is the number of components. For an equimolar alloy, the configurational entropy is given by [12]:

$$\Delta Sconf = Rlnn \qquad (2)$$

Complex and highly concentrated homogeneous solid solutions have extremely low configurational entropies. They are low-entropy alloys (LEAs) with ΔS<0.69R. In contrast, medium-entropy alloys have a temperature-dependent, higher configurational entropy but with a limit that remains below that of high-entropy alloys. However, the exact values can vary for different MEAs, with some quaternary alloys also easily qualifying as HEAs. Two papers suggest different settings to address this, shown in Figure 1.2. The first study classified HEAs and MEAs based on a boundary of 1.5R. The other results classify low-entropy alloys as those with mixing entropies below 1R.

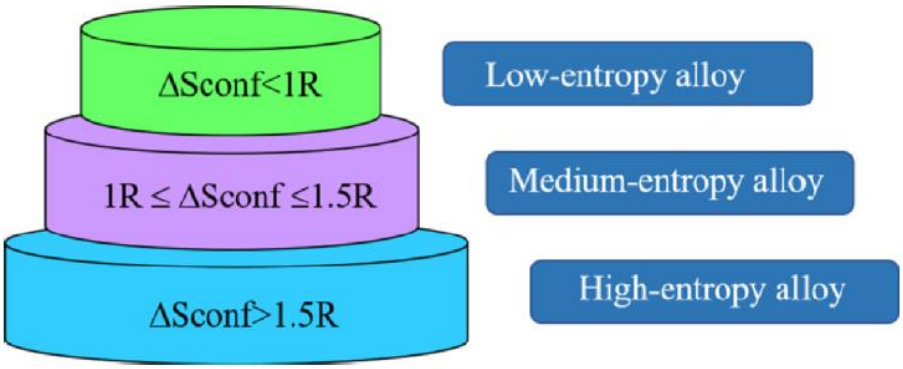

Figure 2: Alloy classification based on configuration entropy[3]

1.2 Unique Properties and Core Effects of High-Entropy Alloys (HEAs)

High-entropy alloys (HEAs) exhibit a range of unique properties due to their complex, multi-element composition, leading to several core effects. These properties make HEAs stand out from traditional alloys and are highly suitable for advanced applications.

1.2.1 High Thermal Stability

HEAs preserve their mechanical properties and structural rigidity at high temperatures, making them suitable for aerospace and power generation industries. Their thermal stability, resulting from a complex multi-element composition that forms stable solid-solution phases rather than intermetallic compounds, is vital for components such as turbine blades and aerospace structures that require maintained properties under extreme conditions for safety and performance.[11].

1.2.2 Exceptional Mechanical Properties

The composition of HEAs, which includes elements of different atomic sizes, leads to lattice distortions in the crystal structure. These distortions hinder the movement of dislocations, thereby enhancing the hardness and strength of the alloys.[13]. Severe lattice distortions in HEAs, documented using techniques like electron microscopy and X-ray diffraction, have been linked to enhanced high-temperature strength, as demonstrated in the refractory HEA NbMoTaW by Zou et al. [14].

1.2.3 Slower Diffusion

The sluggish diffusion effect in HEAs, characterized by slower atomic diffusion than conventional alloys, enhances their high-temperature strength and stability by inhibiting the development of unwanted phases. This phenomenon has been studied by Tsai et al. in the Co–Cr–Fe–Mn–Ni alloy system, where the increased variety of atoms at

each lattice site raises the normalized activation energy, contributing to this effect[15].

1.2.4 Cocktail Effect

The cocktail effect, introduced by Ranganathan in 2003 [16], refers to the synergistic interactions among multiple elements in HEAs that enhance their properties beyond those of individual elements or simpler alloys. This effect enables the customization of materials for specific applications. For example, adding light elements like Al can reduce the density, while refractory elements like Nb can improve high-temperature properties. The result is a material with unique properties not found in traditional alloys, making HEAs ideal for advanced applications in extreme conditions due to their high thermal stability, superior mechanical properties, sluggish diffusion, and the cocktail effect.

1.3 Applications of High-Entropy Alloys (HEAs)

Due to their outstanding properties, HEAs have found applications in several critical fields:

1.3.1 Aerospace

The high strength-to-weight ratio and exceptional high-temperature performance of HEAs make them ideal for use in aircraft and spacecraft components, where both lightweight and durability are crucial. HEAs provide the necessary mechanical strength and thermal stability required for aerospace applications [3].

1.3.2 Automotive

HEAs' wear resistance and mechanical properties are beneficial for manufacturing durable and lightweight automotive parts, potentially improving fuel efficiency and performance. Their ability to withstand harsh environments and mechanical stresses makes them suitable for various automotive applications [11].

1.3.3 Energy

The excellent corrosion and oxidation resistance of HEAs makes them suitable for components in energy generation and chemical processing industries, where materials are often exposed to harsh environments. HEAs' stability and durability under extreme conditions contribute to their effectiveness in these applications[11].

1.3.4 Biomedical

HEAs' biocompatibility and corrosion resistance also show promise for biomedical applications such as implants and prosthetics. The combination of mechanical strength, wear resistance, and biocompatibility makes HEAs ideal for medical devices that require long-term stability and reliability [17].

1. 4 Design Strategies for High-Entropy Alloys (HEAs)

High-Entropy Alloys (HEAs) design involves various strategies to predict and optimize their properties. These strategies leverage both experimental and computational approaches to explore the vast compositional space and identify alloys with desired properties.

1.4.1 Design Strategies

Design strategies in the context of HEAs refer to systematic approaches used to develop alloys with specific properties by manipulating their composition and microstructure. Given the complex multi-element nature of HEAs, traditional trial-and-error methods are insufficient. Instead, these strategies use theoretical models, computational simulations, and machine learning to predict the behavior of different alloy compositions and guide the experimental synthesis of new materials.

1.4.2 Need for Design Strategies

The necessity for design strategies arises from the vast compositional space of HEAs. With five or more principal elements, the number of possible combinations is enormous, making exploring all potential alloys experimentally impractical. Design strategies enable researchers to narrow down the compositional space to the most promising candidates, saving time and resources. These strategies help predict phase stability, mechanical properties, and other critical characteristics, leading to the efficient discovery of new HEAs with tailored properties for specific applications.

1.4.3 Widely Used Design Strategies

1. **CALPHAD Approach**

   The CALPHAD (Calculation of Phase Diagrams) method is thermodynamic-based for predicting phase stability and thereby guiding alloy design through an understanding of the stable phase at a specified composition and temperature for alloy design with desired properties. by phase diagram calculations [18]. This will be a very important point in understanding the phase stabilization and destabilization in HEAs[18].

2. **Density Functional Theory (DFT)**

   DFT calculations help understand the electronic structure of HEAs and predict their mechanical properties at the atomic level. This quantum mechanical method allows researchers to explore the fundamental interactions between atoms and design alloys with tailored properties [19].

3. **Machine Learning**

   Machine learning (ML) approaches have revolutionised materials discovery by predicting properties and guiding experimental efforts through high-throughput screening. ML algorithms can

analyse large datasets to identify patterns and correlations between composition and properties, significantly accelerating the discovery of new materials. This data-driven approach has been used to predict phase formation, mechanical properties, and other critical characteristics of HEAs [3].

4. **CALPHAD + Machine Learning**

Machine learning will then build upon CALPHAD, as it can be used with these databases to extract the phase stability of high-entropy alloys constructed by CALPHAD. CALPHAD can generate large databases through high-throughput thermodynamic calculations and can then be used to train ML models. Using this methodology, it is possible to predict the phase stability with high accuracy and guide the design of new HEAs. For example, CALPHAD has been coupled with advanced machine learning techniques such as eXtreme Gradient Boosting, most commonly abbreviated as XGBoost, by Zeng et al. in the development of high-fidelity phase selection rules for HEAs. The accuracy and efficiency of a design have been improved greatly [20].

1.5 Introduction to Machine Learning Approach in High-Entropy Alloys (HEAs)

Machine learning (ML) has become an important tool in materials science, particularly in designing and optimizing High-Entropy Alloys (HEAs). ML techniques enable the efficient exploration of vast compositional spaces and predict material properties, significantly accelerating the discovery of new alloys with tailored properties. In this research, machine learning is used to predict the phases that may form in a particular composition of HEA using physicochemical descriptors. This application of ML falls under supervised learning, specifically in the task of classification, where the goal is to classify the phases based on given input features.

This section briefly overviews various types of machine learning and the specific models employed in our research to address the classification problem in HEAs.

1.5.1 Types of Machine Learning

1. **Supervised Learning:**

   This is a machine learning technique where the learning model learns from labeled data, and it is in the form of vectors, and the output is a desired result, like a label or a category. The models learn the mapping of the input data to the output data. It may later be used in the performance of the most common tasks: classification and regression. [21,22].

   - **Classification:** Predicting a discrete class label (e.g., identifying the phase of an alloy).
   - **Regression:** Predicting a continuous value (e.g., estimating the hardness of an alloy).

2. **Unsupervised Learning**:

   It deals with unlabelled data, aiming to identify the natural structure within a set of data points. Common tasks include clustering and dimensionality reduction [23,24].

   - **Clustering:** Group formation of the similar data.
   - **Dimensionality Reduction:** It reduces the number of features or descriptors.

3. **Reinforcement Learning**:

   Reinforcement learning entails teaching an agent to make a series of decisions through interactions with an environment. The agent learns to reach a goal by gaining rewards or incurring penalties depending on its actions.[25].

1.5.2 Supervised Learning Models Used in This Research

Our problem statement involves classifying the phase of HEAs based on their composition. We have employed several supervised learning models for this classification task:

1. **Logistic Regression Algorithm:**

   This linear machine learning model is exclusively used for the classification type of problem statements. Logistic regression, being a binary classification, or a multiclass classification, predicts the probability of the default class, but it is flexible and can be coerced into multi-class classification with practices like one-vs-rest [26].

2. **Decision Trees Algorithm**:

   This is a non-linear type of model that divides data into subsets based on feature values. They are straightforward to understand and can manage both categorical as well as numerical data. [27].

3. **Ensemble Techniques**:

   Ensemble techniques combine multiple similar types of models to generate a mean result, which eventually improves prediction accuracy. For example:

   - **Random Forest:** An ensemble of decision trees that reduces overfitting by averaging multiple trees trained on different subsets of the data [28].
   - **Gradient Boosting:** Builds trees sequentially, where each tree corrects the errors of the previous ones [29].

4. **Support Vector Machines (SVM):**

   This machine learning algorithm identifies the pattern in more than four dimensions called hyperplanes that optimally separate classes within the feature space. They are particularly effective in high-dimensional spaces and situations where the number of dimensions exceeds the number of samples. [30].

5. **K-Nearest Neighbours (KNN):**

   This straightforward, example-based learning algorithm classifies a data point by determining the most frequent class among its k-nearest neighbors in the feature space [31].

6. **Artificial Neural Networks (ANNs):**

   These network models are inspired by the human brain and are composed of layers of interconnected mathematical nodes called neurons. They can capture complex non-linear relationships within the data [32].

Chapter 2: Literature Review, Research Methodology and Objectives

2.1 Machine Learning for Phase Prediction

The ability to tailor alloy properties accordingly to meet their intended application makes high entropy alloys an area of day-by-day emerging research in phase prediction. Several machine learning models successfully applied in the prediction of intermetallic (IM) phases, body-centered cubic solid solution (BCC_SS), face-centered cubic solid solution (FCC_SS), and mixed-phases (BCC+FCC, BCC+B2, etc.) in the HEA have reduced the dependence on the trial-and-error approach. Several works already show the variation of performances across different machine learning algorithms and sets of features. For instance, the accuracy was found to be 86% using a random forest classifier in the works by Joshi et al. to predict phases. [33]. Vazquez et al. (2023) developed a deep neural network (DNN) model, achieving a coefficient of determination greater than 0.96 [34]. Guo et al. (2022) simplified the prediction process using a convolutional neural network (CNN), reaching over 98% accuracy for solid solutions and amorphous phases [35]. Moreover, Almomani et al. (2023) utilized artificial neural networks (ANNs) and other models to predict tensile properties, showing ANN's superiority in predictions [36]. These studies and others reviewed highlight the significant advancements and varied approaches in using ML for phase prediction in HEAs, showcasing the potential for accelerated materials discovery and optimization.

Table 1: Literature Data for Phase Prediction Analysis

SN	Objective	Alloy System	Methodology	Machine Learning Models	Ref.
1	Predict phases and develop new HEA	Ni-Cu-Fe-Co-Al	Developed RFC model to predict phases, reporting new HEA with FCC phase.	Random Forest Classifier (RFC)	[33]
2	Estimate phase constitution	Al-Co-Cr-Fe-Mn-Nb-Ni	Developed DNN model with $r^2 > 0.96$, demonstrating effectiveness in predicting phase constitution.	Deep Neural Network (DNN)	[34]
3	Predict phase formation	AlxFeCrNi (x=0, 0.5, 1.0)	Achieved 98% accuracy for solid solutions and amorphous phases using CNN.	Convolutional Neural Network (CNN)	[35]
4	Identify corrosion-resistant compositions	AlCrFeCoNi	Used RFC to identify corrosion-resistant compositions, integrating ML inter-atomic potentials.	Random Forest Classifier (RFC)	[37]
5	Predict tensile properties	FeNiCrCoCu	ANN models showed high prediction accuracy for	ANN, SVM, GPR	[36]

			tensile properties, highlighting strain rate sensitivity.		
6	Predict phase formation in refractory HEAs	Refractory HEAs	Used decision tree classifier, achieving 90% accuracy for phase prediction in refractory HEAs.	Decision Tree Classifier	[38]
7	Predict phases and visualize the design process	Generalized	Voting and Stacking methods achieved over 92% accuracy in phase prediction.	Voting, Stacking, XGBoost	[39]
8	Explain the prediction accuracy of phase selection	Amorphous alloys, SS alloys, IM alloys	Analyzed prediction accuracies using SVM, highlighting the impact of parameter combinations.	Support Vector Machine (SVM)	[40]
9	Predict phase formation	Generalized	Developed an SVM model achieving high prediction accuracy for various phases in HEAs.	Support Vector Machine (SVM)	[41]

10	Design and selection of HEAs for hardmetal matrix applications	Al, Co, Cr, Fe, Ni	Developed new HEAs using experimental and ML approaches, confirming phase predictions with high accuracy	SVM, RF, DT	[42]
11	Predict the structure of HEAs	Al, Co, Cr, Fe, Ni	Employed self-organizing maps (SOM) and feature engineering to predict HEA structures with 87% accuracy	KNN, SVM, ANN, KNN	[43]
12	Develop single-phase BCC RHEAs	Mo, Nb, Ti, V, W	Used Latin hyper-cube sampling and multiple ML algorithms to predict and validate RHEA phases; achieved 93.88% accuracy with RF	LR, SVM, DT, ANN,	[44]
13	Predict phases of HEAs using ML	Al, Co, Cr, Fe, Ni	Analyzed phase prediction accuracy and feature importance using ML models; SVM achieved 90% accuracy	SVM, DT, RF, GNB	[45]

14	Design new RHEAs and explain ML model predictions	Hf, Nb, Ta, Ti, Zr	Used SHAP and LIME to explain ML model predictions for phase formation in new RHEAs; RF achieved 92% accuracy	DT, RF, SVM, KNN, ANN	[46]

2.2 The Choice of Alloy System

The AlCoCrFeNi alloy system is chosen for this machine learning study because of its superior mechanical properties and phase stability, ideal for demanding applications in marine and aerospace environments. This system stands out among high-entropy alloys (HEAs) due to its remarkable mechanical strength, wear resistance, and corrosion resistance. It includes various phases like face-centered cubic (FCC), body-centered cubic (BCC), and a dual-phase BCC+FCC, each affecting its mechanical properties distinctly. For instance, the BCC phase offers high hardness and strength, while the FCC phase is more ductile and tough. The dual-phase BCC+FCC combines these benefits, providing both strength and ductility, making it particularly valuable for applications that face high mechanical stress. [47,48].

AlCoCrFeNi alloys, with their enhanced properties through the synergistic effects of constituent elements like Al, offer significantly higher hardness and strength compared to conventional alloys like 2205 duplex stainless steel (DSS) and Incoloy 825. Their exceptional wear and corrosion resistance make them highly suited for long-term use in harsh, corrosive environments such as marine settings.[47].

2.3 Research Gap

Despite significant progress in using machine learning (ML) for phase prediction in high entropy alloys (HEAs), there are critical gaps that need addressing to enhance model effectiveness and reliability:

2.3.1 Dataset Quality and Balance

- **Issue:** Many current studies use imbalanced datasets, which can lead to biased predictions.
- **Solution:** Ensure datasets are well-balanced and contain a robust number of data points, representing all relevant phases and compositions to develop more accurate and reliable models.

2.3.2 Data Point Availability

- **Issue:** There is often insufficient data points for training complex models, such as artificial neural networks (ANNs), which can lead to overfitting and limited generalizability.
- **Solution:** Increase the collection and validation of experimental data to expand and enhance the reliability of datasets.

2.3.3 Model Specificity

- **Issue:** Generalized models may fail to capture the specific characteristics of different HEA systems, resulting in less precise predictions.
- **Solution:** Develop targeted models that focus on specific types of HEAs or particular phase predictions to improve accuracy and applicability.

2.3.4 Feature Selection and Model Explainability

- **Issue:** The selection of features for model training often lacks correlation with the phase formation processes, affecting the model's performance and explainability.
- **Solution:** Implement effective feature engineering that aligns closely with the physicochemical processes of phase formation and ensures that model predictions are interpretable and scientifically valid.

2.4 Objectives

The objectives of this study are structured as follows:

2.4.1 Develop a High-Accuracy Machine Learning Model:

- **Purpose:** To predict the phases formed in the AlCoCrFeNi alloy system, with a specific focus on BCC, FCC, and the dual-phase BCC+FCC.
- **Highlight:** The primary emphasis is on predicting the BCC+FCC dual-phase composition, known for its superior mechanical properties compared to single-phase systems.

2.4.2 User-Friendly Interface Development:

- **Implementation:** A web-based interface was designed to facilitate ease of use.
- **Functionality:** The interface accepts an Excel file containing alloy compositions and outputs phase predictions, simplifying the user interaction.

2.4.3 Model Validation:

- **Verification Method:** To ensure reliability, the model's predictive accuracy was assessed using previously reported data.
- **Experimental Verification:** An alloy was cast to verify the model experimentally, focusing on achieving specific properties:
    - Yield Strength: 600 MPa
    - Tensile Strength: 1000 MPa
    - Uniform Elongation: 15%
    - Density: Less than 8 grams per cubic centimeter

2.5 Research Methodology

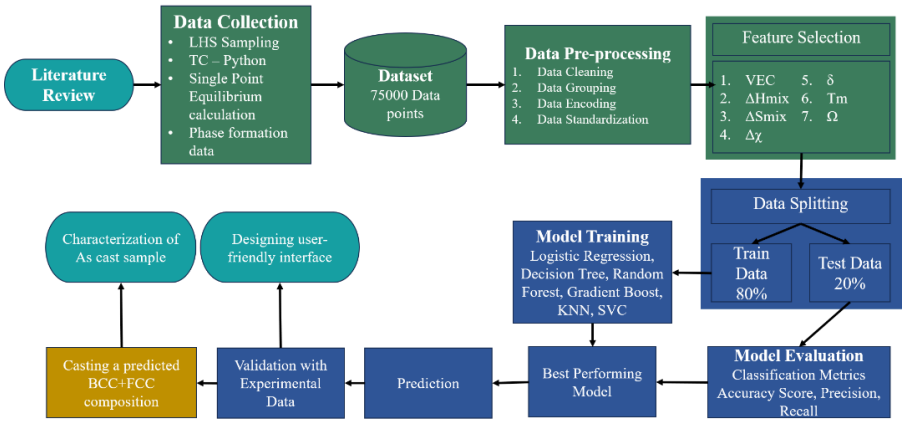

Figure 3: Research Outline for Phase Prediction Machine Learning Model

Research methodology begins with a sound literature review to understand the background and current state of research in the concerned field. After the literature review is conducted, data collection is collected through LHS Sampling and TC-Python for the single-point equilibrium calculations, resulting in a dataset of 75,000 data points. Accordingly, the data pre-processing activity cleans the data from the inconsistencies and groups, encodes the data to be compatible with machine learning, and standardizes the data to bring uniformity. Selected important features include entropy of mixing (ΔSmix), enthalpy of mixing (ΔHmix), Valence Electron Concentration (VEC), lattice distortion (δ), electronegativity difference (Δχ), melting temperature (Tm), and the Omega parameter (Ω). Eighty percent of the dataset is set aside for training, and the remaining twenty percent is set aside for testing. This data is used to train a variety of machine learning models, including Decision Trees, Support Vector Classifiers (SVC), Random Forests, K-nearest neighbors (KNN), and Logistic Regression. Accuracy, precision, recall, and F1 scores are among the classification measures used to evaluate the models' performance. These measures are used to identify which model is the most effective.

This best-performing model is then used to make predictions on new data. The predictions are validated with experimental data to ensure accuracy and reliability. To further validate the model, a sample with a

predicted BCC+FCC composition is cast and characterized to confirm its properties and structure. Finally, a user-friendly interface is designed to facilitate easy use of the model for future predictions.

Chapter 3: Machine Learning

Machine Learning (ML), a branch of Artificial Intelligence (AI), focuses on developing algorithms that analyse and learn from data, enabling the creation of models for predicting outcomes, classifying data, or performing tasks autonomously. The recent surge in ML's popularity is attributed to enhanced computational capabilities and the development of advanced algorithms.

ML is especially valuable for predicting material properties and tackling complex challenges. This chapter details how ML can be applied to predict material properties, highlighting its usefulness in scenarios involving large datasets, complex problem spaces, and unpredictable data patterns.

The typical ML workflow includes several critical stages:

1. **Problem Definition:** Identifying the challenge and setting clear objectives that guide the choice of ML algorithms (classification, regression, dimensionality reduction).
2. **Data Collection:** Acquiring diverse and high-quality data essential for model performance, categorized into supervised, unsupervised, or reinforcement learning approaches based on the data nature.
3. **Data Cleaning and Homogenization:** Addressing data inconsistencies, missing values, and malformed inputs, often using data visualization to extract insights and prepare data for training.
4. **Feature Extraction:** Analyzing feature interactions to derive additional learning features, focusing on those that are significant for the problem at hand.
5. **Data Splitting and Pre-processing:** Dividing the dataset into training and testing sets, applying normalization and stratification to maintain uniformity and enhance algorithm performance.

6. **Training:** Selecting suitable algorithms and adjusting model parameters (or optimizing actions in reinforcement learning) to improve prediction or classification accuracy.
7. **Evaluation:** Assessing the model's effectiveness using unseen data and refining through hyperparameter tuning and retraining, if necessary, until the model meets set goals.

Similar approaches can be found in the literature in general contexts [49] and specifically for material science applications [50].

3.1 Types of Machine Learning

ML algorithms are categorized into Supervised, Unsupervised, and Reinforcement Learning.

3.1.1 Supervised Learning

Supervised learning refers to learning a function which maps an input to an output modeled with example input-output pairs. There is an inferred function describing a set of training examples. Thus, it is labeled and works with the kind of problems where the desired outputs or classes for particular inputs have to be found. The most common approaches to supervised learning include classification and regression [23].

Classification:

This task involves categorizing input data into predefined classes or labels. Classification can be binary, where the output is either yes or no (e.g., determining if a substance forms a compound), or multiclass, where the outputs are multiple categories (e.g., recognizing a handwritten digit from an image).

Regression:

In regression, the goal is to predict a continuous variable based on inputs. This involves establishing a relationship between predictor

variables and a continuous response variable, enabling the model to forecast outcomes accurately.

3.1.2 Unsupervised Learning

Unsupervised learning is a data-driven technique that analyses unlabelled datasets without human intervention. Two popular approaches in this field are clustering and dimensionality reduction [22].

Clustering: Clustering groups data points based on similarities and differences without predefined labels, helping to identify patterns and structures within the data.

Dimensionality Reduction: This method minimizes the quantity of arbitrary factors by establishing a collection of primary factors. It streamlines models, diminishes computational requirements, and addresses the challenge of high-dimensional data, which can impede the effectiveness of machine learning algorithms.

3.1.3 Reinforcement Learning

In reinforcement learning, an agent engages with its surroundings to acquire knowledge by employing a reward system. This reward system assigns a value to the actions the agent executes, which is influenced by the environmental state. Over successive interactions, the agent grasps how to execute actions that yield greater rewards, thereby adjusting its behavior to conform with the reward system. Despite variances, this form of learning can be likened to supervised learning since the researcher furnishes the reward system, delineating a form of "ground truth" for desired behavior [25].

3.2 Algorithms and Applications

With specific examples, the upcoming section will explore standard machine learning methods and their applications in metallurgy and material science research.

3.2.1 Artificial Neural Networks (ANN)

It is a deep learning algorithm which mimics the human brain activity. An artificial neuron's primary structure resembles that of biological neurons, with several inputs and a single output. The term ANN comes from the fact that various networks of neurons can be created by connecting neurons in specific ways. ANN is used for both supervised and reinforcement learning applications.

- **The Perceptron**

The Perceptron model simulates a brain neuron deciding whether to activate based on inputs. Mathematically:

- Input features form a vector xi.
- These inputs are weighted by a vector w.
- The dot product $zi=xi \cdot w$ influences neuron activation.
- The activation function, a step function, determines the output:
  [32]:

Perceptron Activation Function:

$$\boldsymbol{y_i = f(z_i) = \begin{cases} +1 \ \ if \ \ z_i \geq t \\ -1 \ \ if \ \ z_i < t \end{cases}}$$

By including a bias unit $w_0=t$ and $x_0=1$, the function simplifies to:

$$\boldsymbol{y_i = f(z_i) = \begin{cases} +1 \ \ if \ \ z_i \geq 0 \\ -1 \ \ if \ \ z_i < 0 \end{cases}}$$

The Perceptron algorithm iteratively updates the weights based on the predicted and actual output differences.

1. Compute the output value $\boldsymbol{y_i=f'(x_i \cdot w)}$ for the $\boldsymbol{i_{th}}$ entry.
2. Calculate the desired change in the $\boldsymbol{j_{th}}$ weight for the $\boldsymbol{i_{th}}$ entry:

$\Delta w_j = \eta(y_i - \hat{y}_i)\, x_{ji}$

3. Update the weight: $w_j = w_j + \Delta w_j$

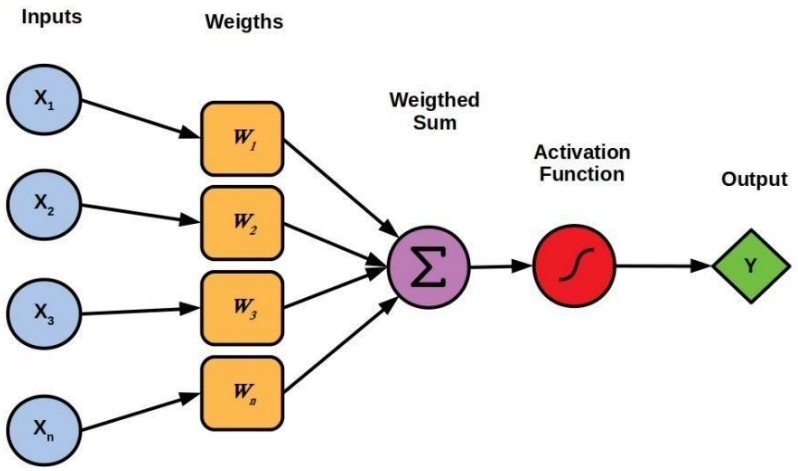

Figure 4: Perceptron - Primary Neuron

Non-Linear Activation Functions

In the Perceptron model we previously examined, the activation function was a step function. More advanced neuron models incorporate activation functions with more than two states, allowing for greater flexibility and complexity [32]. Various activation functions serve different purposes, each with distinct advantages. Here are some of the most common activation functions and their properties:

Common Activation Functions

Sigmoid Function: $f(x) = \sigma(x) = \frac{1}{1+e^{-x}}$

The sigmoid function is a monotonic, non-linear function with a finite range from 0 to 1, and it is continuously differentiable. It is widely used in ANNs for models predicting probabilities. However, it suffers from the vanishing gradient problem, where the gradient approaches zero at extreme values.

Hyperbolic Tangent (tanh) Function:

$$tanh(x) = \frac{e^{x} - e^{-x}}{e^{x} + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

The *tanh* function, with its symmetry and range from -1 to 1, provides a larger gradient than the sigmoid function. Despite this advantage, it also experiences the vanishing gradient problem. The tanh function is essentially a scaled and shifted version of the sigmoid function:

$$tanh(x) = 2\sigma(2x) - 1$$

Rectified Linear Unit (ReLU) Function:

$$ReLU(x) = max(0, x)$$

The *ReLU* function is non-linear and has become the default activation function for many types of neural networks due to its simplicity and effectiveness. It does not suffer from the vanishing gradient problem, allowing for faster and more efficient training. However, it can encounter issues with "dying *ReLU*s," where neurons become inactive and stop learning.

Neural Networks - Layers of Neurons

Neurons in an ANN are organized into layers:

- **Input Layer:** Number of neurons equals the number of input features.
- **Output Layer:** Number of neurons equals the number of predicted features or classes [32].

Gradient Descent and Backpropagation

Gradient Descent adjusts ANN weights to minimize an error function, such as Mean Squared Error (MSE):

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

where yi represents the actual values, and $y\hat{}i$ represents the predicted values.

Backpropagation calculates the gradient of the loss function with respect to each weight, propagating errors backward through the network to update weights effectively.[32].

3.2.2 Linear Regression and Regularization:

Linear Regression is a cornerstone machine learning algorithm aiming to delineate a linear connection between input data and a target variable. Its operation entails fine-tuning the slope of each feature to minimize a predetermined cost function. Linear Regression adheres to a parallel principle as the Perceptron model, utilizing gradient descent on the squared errors cost function to iterate on the weights. The cost function, symbolized as $J(w)$, is computed as the summation of squared errors:

$$J(w) = \frac{1}{n}\sum_{J=1}^{1}(yi - \hat{y}i)^2$$

Where yi signifies the actual target values, $y\hat{}i$ represents the predicted values, and n stands for the number of samples. While Mean Squared Error (MSE) serves as a prevalent metric for assessing the model's efficacy, another vital measure is the coefficient of determination ($R2$).

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

where SSE represents the sum of squared errors, SST is the sum of squares total, y^- is the mean of the target variable, and $\hat{y}i$ is the predicted value.

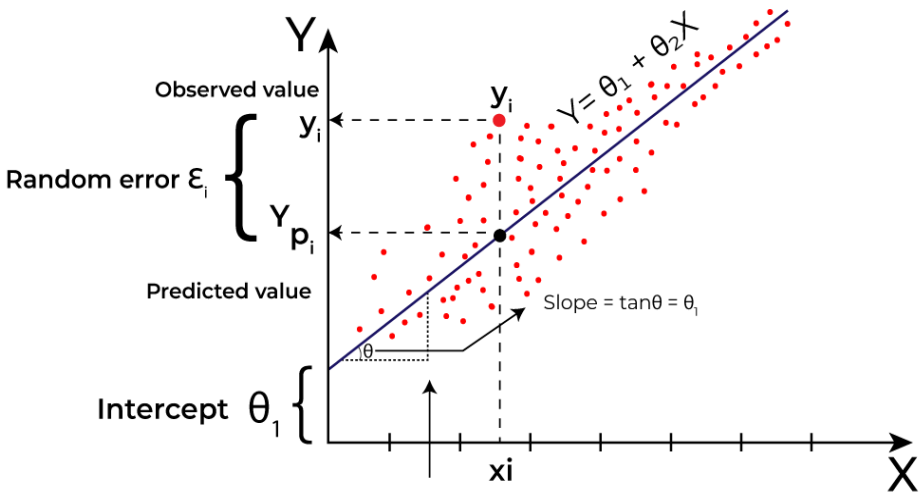

Figure 5: Linear Regression Geometric Intuition

When applied to the training dataset, $R2$ ranges between 0 and 1, also known as the goodness of fit. It indicates the model's ability to capture patterns within the independent features. A higher $R2$ value signifies that the model captures these patterns more effectively, leading to more accurate predictions. However, when applied to the testing dataset, it can be negative, suggesting that the model performs worse than a simple horizontal line. Notably, a perfect fit with a mean squared error of zero yields a coefficient of determination of 1 [22,23].

Regularization Techniques: L1 and L2

Regularization methods like L1 (Lasso Regression) and L2 (Ridge Regression) are used in Linear Regression to prevent overfitting.

1. **L1 Regularization**:

    - Adds the absolute value of coefficients to the loss function.

    - Can lead to sparse models by zeroing out less important features.

2. **L2 Regularization**:

    - Adds the squared magnitude of coefficients to the loss function.
    - Shrinks coefficients towards zero without making them exactly zero, helping to reduce multicollinearity.

L2 regularization tends to shrink the coefficients but usually does not lead to sparse models [22,23].

Gradient Descent Optimization

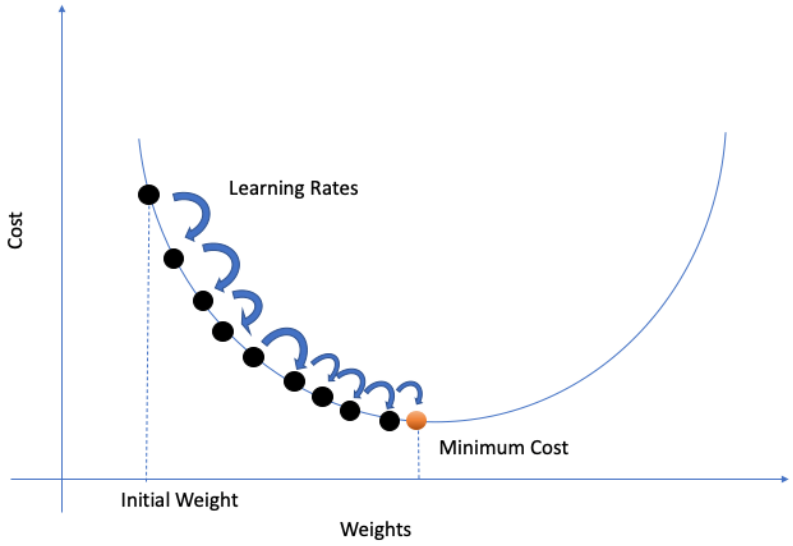

Figure 6: Stochastic Gradient Descent

Gradient Descent is an optimization technique employed to minimize the cost function within Linear Regression. Its principle lies in commencing with an initial set of coefficients and systematically refining them in a manner that maximally diminishes the cost function. This iterative adjustment process relies on the gradients, or partial derivatives, of the cost function concerning the coefficients.

The update rule for the coefficients is:

$$w_j = w_j - \alpha \frac{\partial J(w)}{\partial w_j}$$

Where:

- wj is the coefficient.
- α is the learning rate, e.g. 0.01.
- $\frac{\partial w_j}{\partial J(w)}$ is the partial derivative of the cost function with respect to the coefficient wj.

The gradient descent process continues until the changes in the cost function are smaller than a predefined threshold, indicating that the algorithm has converged to a minimum [22].

3.2.2 Logistic Regression

It is a straightforward machine learning model primarily utilized for binary classification tasks, where the response variable has one of two possible outcomes. It can also be extended to handle multiclass classification problems. Linear Regression predicts continuous data, whereas Logistic Regression estimates the probability that a given input pertains to a specific class. This is accomplished by utilizing the logistic function, also recognized as the sigmoid function, which transforms any real-valued input into the interval [0, 1].[21]

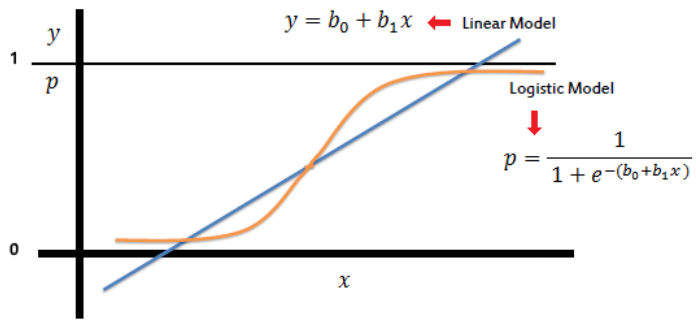

Figure 7: Logistic Regression Geometric Intuition

The logistic function is defined as:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

Where:

- $\sigma(z)$ is the logistic function.
- z is the linear combination of the input features,

$$z = w_0 + \sum_{j=1}^{p} w_j x_j$$

The probability that the response variable y equals 1 is given by:

$$P(y = 1|x) = \sigma(z) = \frac{1}{1 + e^{-\left(w_0 + \sum_{j=1}^{p} w_j x_j\right)}}$$

The probability that y equals 0 is:

$$P(y = 0|x) = 1 - \sigma(z) = \frac{1}{1 + e^{-\left(w_0 + \sum_{j=1}^{p} w_j x_j\right)}} \cdot e^{-\left(w_0 + \sum_{j=1}^{p} w_j x_j\right)}$$

The **cost function** for Logistic Regression is derived from the likelihood function, which measures how well the model predicts the observed data. The goal is to maximize this likelihood function or minimize the negative log-likelihood equivalently. The cost function $J(w)$ is given by:

$$J(w) = -\frac{1}{n}\sum_{i=1}^{n}[y_i \log(\widehat{y_i}) + (1 - y_i)\log(1 - \widehat{y_i})]$$

Where:

- $y\text{^}i$ is the predicted probability that yi=1.
- yi is the actual binary outcome (0 or 1).

3.2.3 Support Vector Machine (SVM)

It is a supervised machine learning algorithm suitable for both classification as well as regression endeavors. Its core objective entails identifying a hyperplane within the data space that maximizes the minimum separation distance between samples associated with distinct classes. This specific hyperplane is termed the maximum margin hyperplane.[51].

SVMs build upon the simple Perceptron model discussed earlier. While the Perceptron model uses the Mean Squared Error (MSE) for misclassification, SVMs maximize the distance between the separating hyperplane and the nearest samples from each class, known as "support vectors."

The equations of the two hyperplanes, in vector form, are:

$$w \cdot x_{1i} + b = 1$$

$$w \cdot x_{2i} + b = -1$$

where w is the weight vector, x_{1i}, and x_{2i} are the sample vectors, and b is the bias term.

SVMs can struggle with datasets that are not linearly separable. To address this issue, the kernel trick is used. This approach involves applying a non-linear function to the input features, mapping them into a higher-dimensional space where they become linearly separable. In this new space, the standard SVM algorithm can be applied effectively.

The optimization objective of SVM can be represented as:

$$min\ (\frac{1}{2} \mid\mid w \mid\mid^2 + c \sum_i \xi i)$$

Here, c is a model hyperparameter that controls the trade-off between achieving a large margin and minimizing classification errors. Small values of c allow more classification errors but result in a larger

margin, while large values of *c* reduce classification errors but produce a smaller margin.

Example of SVM in Classification: Consider a dataset where we must classify two types of materials based on their properties. The goal is to find the optimal hyperplane that separates these two classes with the maximum margin.

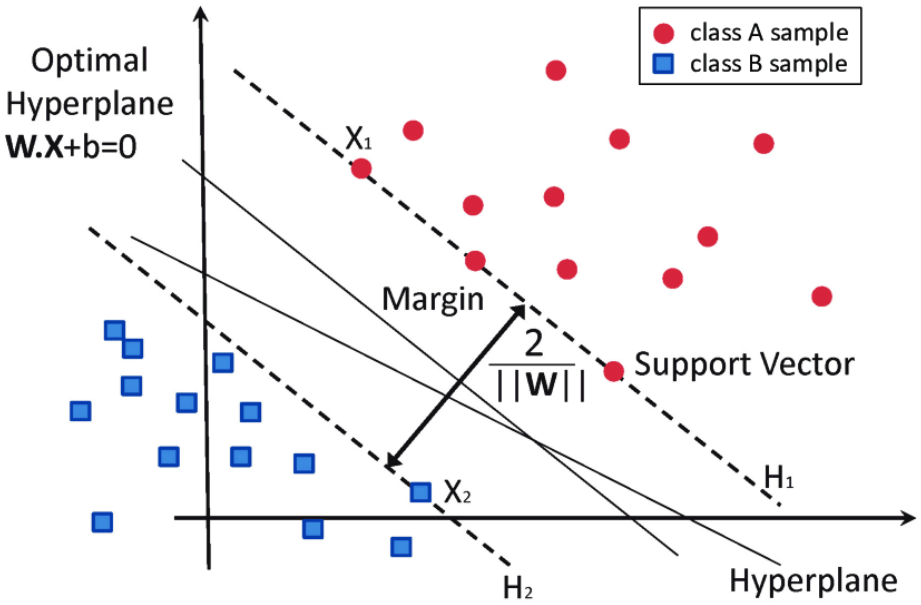

Figure 8: SVM Classification with Linear Kernel

In this figure, the blue and red points represent two different classes, and the black line is the maximum margin hyperplane that separates them. The support vectors are the points closest to the hyperplane.

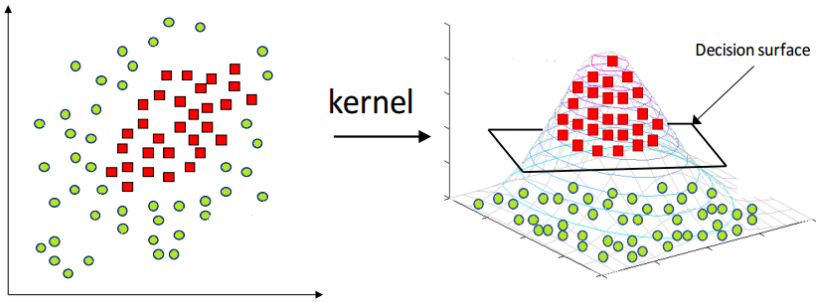

Figure 9: SVM Classification with Non-linear Kernel

This figure uses a non-linear kernel to map the data into a higher-dimensional space, making it linearly separable. The black curve represents the separating hyperplane in this transformed space. Using SVM with appropriate kernels can efficiently handle complex classification tasks, providing robust and accurate models.

3.2.4 Decision Trees (DT) and Random Forests (RF)

Decision Trees (DT) represent a transparent and intuitive classification method that organizes data points into various categories based on a series of questions. Each decision tree consists of nodes and leaves. Nodes are decision points that split the data based on the values of specific features, while leaves represent the classification outcomes. For categorical features, the questions might ask whether a feature equals a specific value or falls within a range of values. For continuous features, comparison operators are used to make decisions. The process of creating these questions and splits is recursive, continuing until all data points within a node are of the same class or a predefined limit on tree depth or node size is reached, typically restricting splits to binary outcomes for simplicity [52].

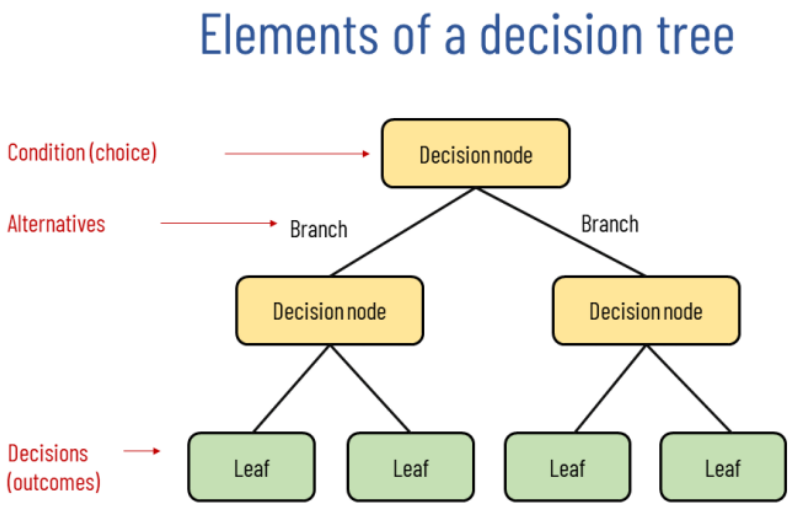

Figure 10: Elements of Decision Tree Classifier

This learning process entails constructing a tree-like model where the optimal sequence of questions maximizes the information gained from each split. The metric used to measure this gain is known as the Information Gain (IG), illustrated in Figure 3.7, which depicts a decision tree with its root and leaf nodes. The formula for IG is expressed as:

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^{m} \frac{N_p}{N_j} I(D_j)$$

where f represents the feature used for splitting, I is a function that measures impurity (such as Gini impurity or entropy), Dp is the dataset at the parent node, Dj are the datasets at the jth child node, and Np, Nj are the numbers of elements at the parent and jth child node, respectively.

Gini impurity and entropy are commonly used measures of impurity. Gini impurity is calculated as:

$$IG = 1 - \sum_{i=1}^{c} p(i|j)^2$$

Entropy is another measure calculated using the following:

$$IH = -\sum_{i=1}^{c} p(i|j) \log_2 p\,(i|j)$$

where $p(i|j)$ is the proportion of class i samples at node j. These measures help ensure that each split maximizes the homogeneity of classes in the child nodes.

A significant challenge in decision tree models is their tendency to overfit the training data, impairing their generalization to new data. This overfitting is often due to excessively complex trees that too closely model the training data's nuances. A robust solution to this problem is to use Random Forests (RF), an ensemble method that combines multiple decision trees to enhance predictive performance and robustness. Each

tree in a Random Forest is constructed using a different "bootstrap sample" from the dataset, ensuring diversity among the trees. A subset of features is randomly selected at each node, guiding the split decisions to decorate the trees further and boost the ensemble's accuracy.

This ensemble approach mitigates overfitting by averaging the predictions from multiple trees, effectively using a majority voting system among the trees to classify new samples, as illustrated in Figure 3.7. This figure should represent multiple decision trees forming a forest, with arrows showing how individual tree predictions are aggregated to produce the final ensemble prediction.

3.2.5 K-Nearest Neighbors (k-NN) Algorithm

The k-Nearest Neighbors (k-NN) algorithm represents a simple and extensively employed non-parametric technique within classification and regression undertakings. Its operation involves forecasting the label of a novel data point by considering the majority label or the average outcome of its 'k' nearest neighbors from the training dataset. [52,53].

Steps of the k-NN Algorithm

1. **Selection of 'k'**: Choose the number of neighbors 'k'. The choice of 'k' influences the prediction, with smaller values potentially leading to overfitting.

2. **Distance Calculation**: Compute the distance between the test instance and each instance in the training set. The standard metric used is Euclidean distance.

3. **Identify Neighbors**: Determine the 'k' nearest neighbors to the test instance based on the calculated distances.

4. **Aggregate Labels**: Assign the most common label among the neighbors to the test instance for classification. For regression, calculate the mean of the neighbors' values.

5. **Prediction Output**: The result from the aggregation step is the predicted label or value for the test instance.

The equation for Euclidean Distance Calculation:

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

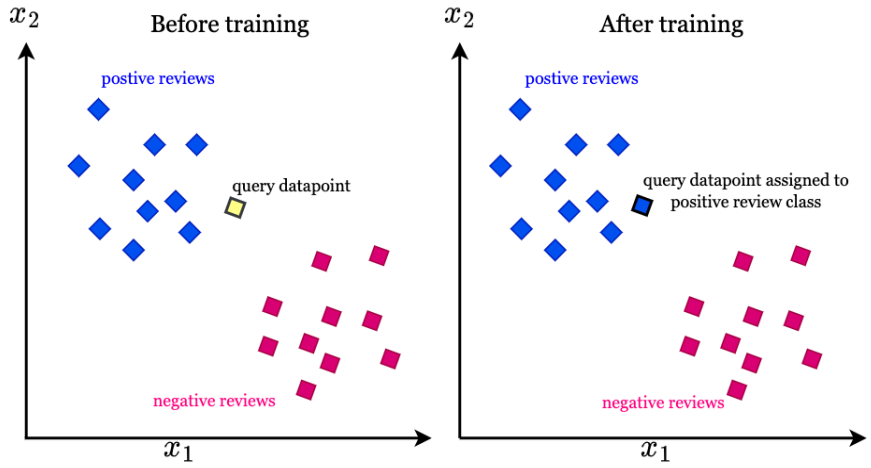

Figure 11: KNN algorithm working

Figure 3.8 illustrates the selection of k-nearest neighbors in a two-dimensional space, showing how different values of 'k' affect classification boundaries and decision-making. This visual helps us understand the influence of neighbor selection on the algorithm's predictions.

Chapter 4: Data Collection, Preprocessing and Modelling

This chapter outlines the crucial steps needed to build a machine-learning model for identifying the BCC-FCC regime in Al-Co-Cr-Fe-Ni alloys. It begins with explaining Latin Hypercube Sampling (LHS) and its importance in covering the entire range of alloy compositions. The method for gathering accurate thermodynamic data using Thermo-Calc is also detailed, highlighting its essential role in the machine learning model. The chapter further explains the necessary data cleaning and preprocessing processes to ensure data quality and readiness for modeling. Lastly, the approach to modeling, which facilitates the analysis of phase transitions in these alloys, is described.

4.1 Latin Hypercubic Sampling (LHS)

Understanding the Quasi-Random sampling is essential before diving into LHS.

4.1.1 Quasi-Random Sampling

Quasi-random sampling methods, also recognized as low-discrepancy or deterministic sampling methods, comprise a set of sampling techniques intended to yield samples that exhibit greater uniformity compared to random samples. While random sampling selects points randomly and independently, quasi-random sampling methods aim to achieve more balanced coverage across the parameter space by minimizing the discrepancy between the generated points and the desired distribution.

Latin Hypercube Sampling (LHS) is an example of a quasi-random sampling method, an efficient statistical approach engineered to produce random samples of parameter values from a multidimensional distribution. This method is grounded in the principle of the Latin square design, which guarantees that each sample occupies a unique position within each axis-aligned hyperplane of the space. In systems with

multiple dimensions, such as alloy compositional space, this arrangement extends into a "hypercube" configuration. [54].

LHS is especially advantageous when dealing with the Al-Co-Cr-Fe-Ni system, where each of the five elements can vary between 5% and 35%. It divides the range of possible concentrations for each element into equal probable segments. For instance, if generating 75,000 samples, the concentration range for each element is divided into 75,000 intervals. From each interval, a single concentration is randomly selected, ensuring that each possible value has an equal chance of being chosen. This strategy prevents the clustering of values and covers the entire compositional range thoroughly and uniformly.

Discrepancy measures the deviation between the actual distribution of sample points and a perfectly uniform distribution across the sampled space. It quantitatively assesses how evenly sample points are spread throughout the parameter or compositional space. The lower the discrepancy, the closer the sample distribution is to being uniform.

Discrepancy is crucial in model-building scenarios because:

- **Uniformity in Sampling**:
  A lower discrepancy ensures that all regions of the parameter space are adequately represented, reducing the risk of bias in the model's predictions or simulations.
- **Efficiency of Sampling**:
  Methods with lower discrepancies require fewer samples to achieve a representative coverage of the space, which can be more computationally efficient and cost-effective, especially in high-dimensional settings.

In the specific case of materials science and alloy development, where compositional variations can lead to dramatically different material properties, ensuring a low discrepancy in sampling allows for a more thorough and accurate exploration of potential material behaviours under various compositional adjustments.

4.1.2 Detailed Comparison of Sampling Methods

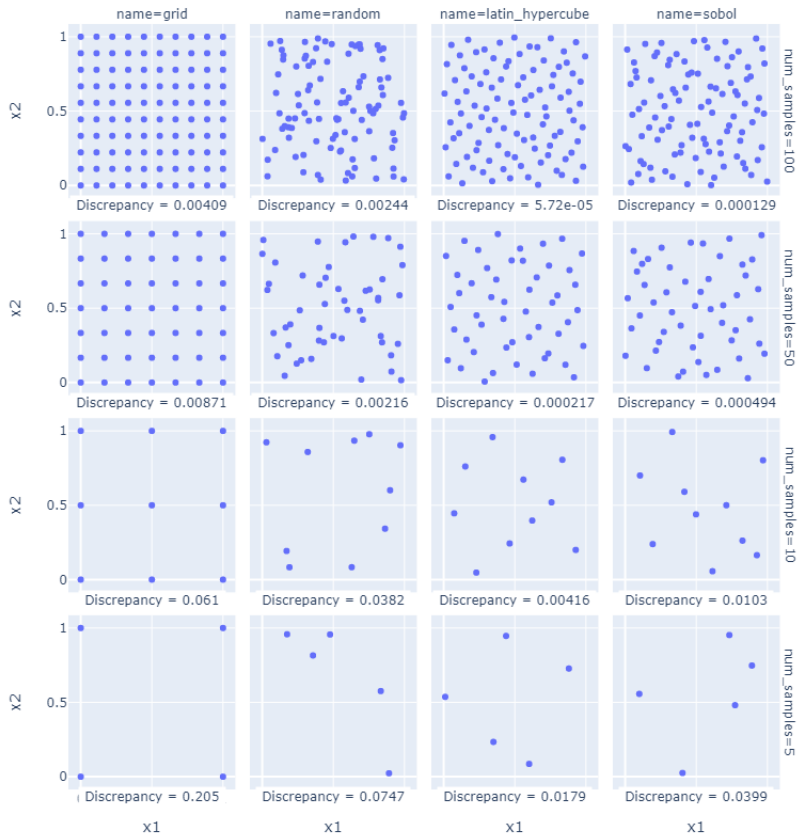

Figure 12: Discrepancy plot within different sampling techniques

1. **Grid Sampling**:

   This method, though structured, shows significant gaps in higher dimensions. For example, with 10 dimensions, the discrepancy reaches an overwhelming 53.396889, indicating very poor space coverage.

2. **Random Sampling**:

   While slightly better than Grid sampling, Random sampling still suffers from occasional clustering and gaps. Its discrepancy increases substantially with the number of dimensions, reaching 11.771547 in 20 dimensions.

3. **Latin Hypercube (LHS)**:

   LHS consistently outperforms both Grid and Random sampling in terms of discrepancy. At 20 dimensions, the discrepancy recorded for LHS is 6.131461, significantly lower than the others, demonstrating its superior space-filling capability.

4. **Sobol Sequences**:

   Similar to LHS, Sobol sequences are another form of quasi-random sampling that achieves low discrepancies. At 20 dimensions, Sobol's discrepancy is 7.369506, slightly higher than LHS but still far more efficient than Grid or Random sampling.

4.1.3 Performance in Higher Dimensions

Table 2: Discrepancy comparison at higher dimensions

SN	Number of Dimensions	Discrepancy	Sampling Method
1	2	0.000064	Latin_hypercube
2	2	0.000134	Sobol
3	2	0.004093	Grid
4	2	0.006478	Random
5	3	0.00021	Latin_hypercube
6	3	0.000324	Sobol
7	3	0.010809	Random
8	3	0.053356	Grid
9	10	0.018654	Latin_hypercube
10	10	0.022224	Sobol
11	10	0.068428	Random
12	10	53.396889	Grid
13	20	0.454235	Latin_hypercube

14	20	0.571777	Sobol
15	20	0.840589	Random
16	20	3309.123807	Grid

As dimensionality increases, the efficiency of quasi-random methods like LHS and Sobol becomes more apparent. The discrepancy values for LHS and Sobol in a 20-dimensional space are markedly lower compared to Grid and Random sampling. This is crucial in fields like materials science, where high-dimensional compositional spaces are common.

The superiority of LHS in creating compositional spaces for high entropy alloys is evident from the significantly lower discrepancy values compared to traditional sampling methods. This uniform sampling across all dimensions ensures that no part of the compositional space is overlooked, making LHS particularly suitable for generating reliable data for machine learning models. As shown in the discrepancy data, LHS provides a methodical and efficient approach to sampling in complex, multi-dimensional spaces, making it an invaluable tool in the computational exploration of new materials.

The objective was to create 75,000 samples for Thermo-Calc calculations to explore the compositional space of the Al-Co-Cr-Fe-Ni high entropy alloy system. A Python code was designed to perform Latin Hypercube Sampling (LHS) to achieve this, ensuring a comprehensive and statistically robust dataset.

The process begins by initializing a random seed to guarantee reproducibility, allowing for consistent results across different code executions. A total of 75,000 unique datasets were aimed to be generated, each representing a different potential composition of the five elements: Aluminum (Al), Cobalt (Co), Chromium (Cr), Iron (Fe), and Nickel (Ni). The concentrations of each element were allowed to vary between 5% and 35%. To maintain realistic alloy compositions, the sum of the

concentrations of all five elements in each dataset was constrained to equal 100% (or 1.0000 in fractional terms). This constraint is crucial as it mirrors the real-world scenario where the total composition of the alloy must sum to 100%.

The dataset generation involves selecting random values for Al, Co, Cr, and Fe concentrations within their designated ranges. The concentration of Ni, the fifth element, is calculated by subtracting the sum of the concentrations of the other four elements from 100%. This step ensures the total composition remains consistent at 100%. Given the potential for rounding errors during calculations, the concentration of Ni is recalculated after rounding the concentrations of the other elements to four decimal places. This recalibration corrects any discrepancies caused by rounding, ensuring that the sum of the concentrations still equals 100%.

Each combination of element concentrations is tracked to prevent duplication in the dataset. If the recalculated concentration of Ni falls within its allowable range and the combination has not been previously recorded, it is added to the dataset. Once all 75,000 unique datasets are generated, they are compiled into a DataFrame—a tabular structure provided by the Pandas library. This data frame is subsequently saved to a CSV file, ensuring that the data is readily accessible for further analysis, such as for input into Thermo-Calc for thermodynamic modeling or other computational analyses.

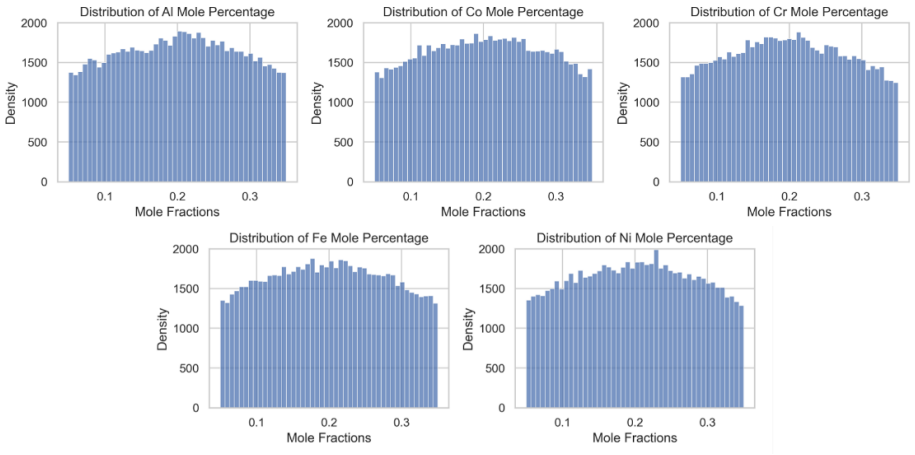

Figure 13: Uniform distribution of all elements within the 5% to 35% range.

This systematic approach guarantees extensive coverage across the possible compositional space and adheres to practical constraints typical in alloy research and development. The resulting dataset serves as a foundational resource for investigating the effects of compositional variations on the properties of the Al-Co-Cr-Fe-Ni high entropy alloy system.

4.2 Data Collection using ThermoCalc

Thermo-Calc is a sophisticated computational software specifically designed to analyze thermodynamic properties in multi-component systems. It plays a pivotal role in the field of materials science, enabling researchers and engineers to simulate complex chemical interactions and phase transformations. This capability is essential for predicting the properties and behaviors of new alloys under various conditions, thereby aiding in materials design and optimization.

Integral to enhancing Thermo-Calc's functionality is its TC Python API, an interface that allows for the automation of Thermo-Calc through Python scripts. This integration streamlines complex calculations and facilitates the seamless incorporation of thermodynamic data into machine-learning models. By utilizing the TC Python API, researchers can efficiently generate, manipulate, and analyze extensive

datasets on material properties, significantly accelerating the cycle of computational materials engineering and discovery.

4.2.1 Data Retrieval Using TC-Python API

TC Python API was utilized to conduct detailed thermodynamic calculations for the Al-Co-Cr-Fe-Ni high entropy alloy system, using 75,000 compositions that were generated using Latin Hypercube Sampling (LHS). These calculations were essential for determining stable phases under various compositional conditions.

The computational work was orchestrated through a script utilizing the TC Python API, which facilitated direct interaction with Thermo-Calc's thermodynamic databases and computational tools. The script employed the "TCHEA5" database, a comprehensive source containing essential data for high entropy alloys, including the elements Al, Co, Cr, Fe, and Ni. This choice ensured that the calculations were based on accurate and relevant material data.

4.2.2 Methodology for Thermodynamic Calculations

The script was designed to automatically process a dataset containing various alloy compositions. For each composition, the script executed a series of steps:

1. **Initialization and Setup**:
   A session with Thermo-Calc was initiated, setting up a specific cache folder to optimize performance and data handling. The script then configured the computational environment to select the appropriate database and elements and to prepare the system for single equilibrium calculations, a method that focuses on calculating the equilibrium state for a set temperature without searching for the global minimum energy state.
2. **Phase Analysis**:
   The script identified the stable phases at temperatures just below the solidus. This was critical for understanding the

microstructural characteristics of the alloy at different cooling stages. A list of stable phases was taken out and recorded into the data frame.

4.2.3 Data Recording and Output

A list of stable phases for each composition was systematically recorded alongside the original compositional data in the dataset. The goal is to create a labeled dataset that was not available earlier. This dataset was then saved to a CSV file. Further, the design parameters were calculated to complete the labeled dataset.

4.3 Design Parameters

High Entropy Alloys (HEAs) leverage elevated configurational entropy to stabilize solid solution phases rather than intermetallic compounds. This stabilization arises from the intricate interplay of several physicochemical attributes, such as Valence Electron Concentration (VEC), mixing enthalpy ($\Delta Hmix$), mixing entropy ($\Delta Smix$), atomic size variation (δ), electronegativity variance ($\Delta\chi$), the parameter Ω, and melting temperature (Tm). Grasping these characteristics proves crucial in forecasting phase emergence and sustainability within HEAs, particularly in the formation of Body-Centered Cubic (BCC) and Face-Centered Cubic (FCC) structures.

4.3.1 Key Physicochemical Properties

1. **Valence Electron Concentration (VEC)**:

   The VEC is a critical parameter in determining the crystal structure of HEAs. Alloys with a VEC around 8 tend to favor the formation of FCC structures, while those with a VEC around 6.87 or lower are more likely to form BCC structures. This relationship helps predict phase stability and guide alloy design [55].

$$\text{VEC} = \sum_{i=1}^{n} c_i V_i$$

2. **Mixing Enthalpy ($\Delta Hmix$)**:

Mixing enthalpy ($\Delta Hmix$) represents the enthalpy change during the mixing of elements. For HEAs, a $\Delta Hmix$ range between -15 and 5 kJ/mol is conducive to forming solid solutions. Values outside this range typically promote the formation of intermetallic compounds, making $\Delta Hmix$ a key factor in phase prediction [55].

$$H_{\text{mix}} = \sum_{i=1, i\neq j}^{n} \Omega_{ij} c_i c_j$$

The interaction parameter for a regular solution between the ith and jth elements is defined as $\Omega_{ij} = 4\frac{H_{\text{mix}}}{D_{AB}}$, where ci or cj represents the atomic percentage of the ith or jth component and H_{mix}^{AB} is the enthalpy of mixing for binary liquid alloys. The values for the enthalpy of mixing ($H\text{mix}AB$) can be calculated using the Miedema macroscopic model for binary liquid alloys, as detailed in referenced literature [56].

3. **Mixing Entropy ($\Delta Smix$)**:

The entropy of mixing ($\Delta Smix$) is always positive and increases with the number of elements in the alloy. High $\Delta Smix$ values enhance the stability of solid solutions by increasing configurational entropy, which counters the enthalpy effects that favour compound formation [55].

$$S_{\text{mix}} = -R \sum_{i=1}^{n} (c_i \ln c_i)$$

where c_i is the mole per cent of the component, $\sum_{i=1}^{n} c_i = 1$, and R (=8.314 JK^{-1} mol^{-1}) is gas constant.

4. **Atomic Size Difference (δ)**:

The Atomic size difference (δ) is characterized as the root-mean-square deviation of atomic sizes among the constituent elements. A diminutive δ (usually ≤ 6.6%) supports the establishment of stable solid solutions by curtailing lattice distortions. Conversely, larger δ values can induce substantial lattice strain and instability, thereby promoting the creation of intermetallic compounds or amorphous phases.

$$\delta = \sqrt{\sum_{i=1}^{n} c_i \left(1 - \frac{r_i}{\bar{r}}\right)^2}$$

where *ci* is the atomic percentage of the *i*th component,

$\bar{r} = \sum_{i=1}^{n} c_i r_i$ is the average atomic radius, and *ri* is the atomic radius [57]

5. **Electronegativity Difference ($\Delta\chi$)**:

The electronegativity difference ($\Delta\chi$) among constituent elements affects the type of bonding and phase formation. Large $\Delta\chi$ values promote the formation of compounds with strong, directional bonds, whereas smaller differences favour random solid solution phases.$\Delta\chi = \sqrt{\sum_{i=1}^{n} c_i(\chi_i - \bar{\chi})^2}$, $\bar{\chi} = \sum_{i=1}^{n} c_i\chi_i$, where χ_i is the Pauling electronegativity for the *i*th component [55].

6. **Parameter Ω**:

The parameter $\Omega = \frac{T_m \Delta S_{\text{mix}}}{|\Delta H_{\text{mix}}|}$ is defined, where Tm is the melting temperature. This parameter helps balance the effects of entropy and enthalpy in predicting solid solution

formation. An Ω value greater than 1.1 indicates a high likelihood of forming solid solution phase [57].

7. Melting Temperature (Tm):

The melting temperature (Tm) of an alloy is used in conjunction with $\Delta Smix$ and $\Delta Hmix$ to calculate Ω. Higher melting temperatures generally favour the stability of solid solution phases at elevated temperatures.

$$T_m = \sum_{i=1}^{n} c_i (T_m)_i$$

Here, $(Tm)i$ is the melting point of the *i*th component of an alloy [57].

4.3.2 Python Code for Design Parameters

Python code processes data from an Excel file containing element properties and calculates various material science parameters. It defines a class named DesignParameters that handles data processing and calculations. The class is initialized with an Excel file read into a pandas DataFrame. An element data dictionary stores properties for elements such as atomic number, weight, density, melting point, electronegativity, radius, and valence electron count. The code includes methods for preprocessing data, calculating the sum of element percentages and the valence electron count, computing entropy and enthalpy changes, determining electronegativity differences, and calculating atomic size differences. Additionally, it calculates the melting temperature, omega parameter indicating alloy stability, and alloy density. The process_dataframe method applies all these calculations to the DataFrame, updating it with new columns for each parameter. An example usage block demonstrates how to initialize the class with an Excel file, process the DataFrame, and save the results to a new Excel file. This code provides a comprehensive method for analyzing and calculating important material properties from elemental composition data.

4.4 Data Cleaning

Data cleaning is essential in machine learning because it ensures the quality and reliability of the data used to train models. Clean data helps improve machine learning algorithms' accuracy, performance, and generalization ability. It involves removing or correcting errors, handling missing values, normalizing data, and eliminating duplicates. Models may learn from noise and inaccuracies without proper data cleaning, leading to poor predictions and unreliable outcomes. Thus, data cleaning is a critical step in the data preprocessing pipeline, contributing to developing robust and effective machine learning models.

The data cleaning process involved removing data points with more than two phases. This study focused only on single phases (BCC and FCC) and dual phases (BCC + FCC) for model training. Any other phases were excluded. Additionally, BCC + B2 dual phases were not considered due to the ambiguity between true BCC and B2, as Thermo-Calc typically cannot distinguish between these two. After this cleaning process, the final dataset contained only three classes: BCC, FCC, and BCC + FCC dual phase, with a total of 54,056 data points. Data cleaning also involves removing data columns that are not useful or necessary, such as serial numbers, compositions, and the number of phases. These columns were deemed unnecessary for the analysis and were removed to streamline the dataset. Further steps included data encoding and standardization.

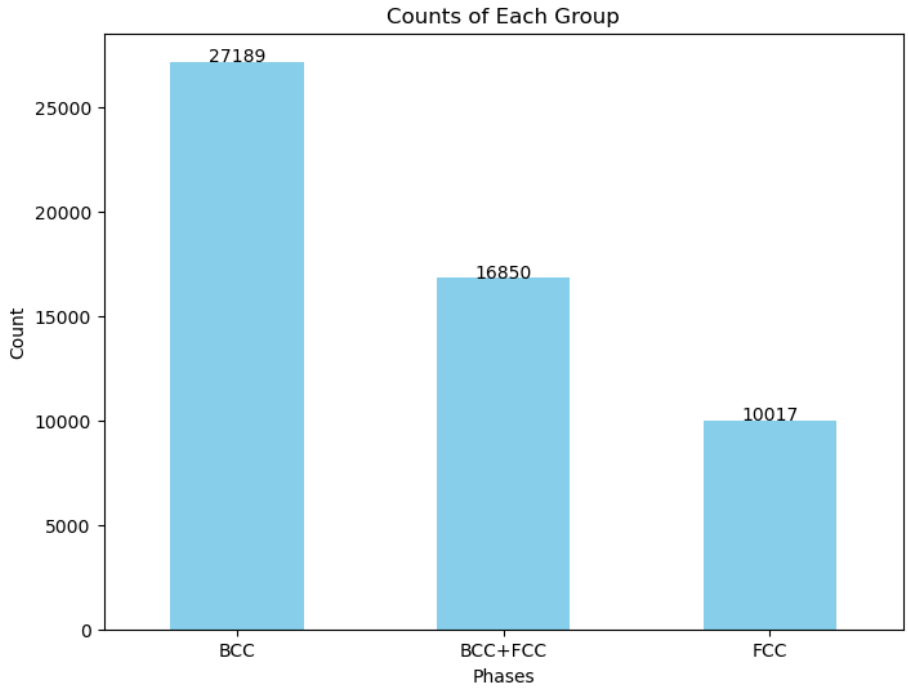

Figure 14: Data distribution in each class

4.5 Data Preprocessing

The first step in data preprocessing is converting categorical columns into numerical format, known as data encoding. In this study, the 'phases' column is the target column and is categorical. It was converted into a numerical column using encoding: 0 was assigned to the BCC class, 1 to the BCC + FCC class, and 2 to the FCC class. This type of encoding is called ordinal encoding.

After data encoding, the next step is data standardization. Standardization is important because it scales the data to a common range, improving many machine learning algorithms' performance and convergence speed. It ensures that each feature contributes equally to the model. Standardization is done using the z-score formula:

$$[z = \frac{(X - \mu)}{\sigma}]$$

Following standardization, the dataset is segregated into X and y variables, where X signifies the independent features and y symbolizes

the dependent (target) variable. The final dataset is divided into two parts: one for training the machine learning model and the other for testing it. The training set comprises 80% of the data, while the test set contains the remaining 20%.

4.6 Model Training

The training data was utilized to train several machine-learning models. Initial models included basic ones like logistic regression and tree-based models such as decision trees and random forests. Random forests tend to exhibit strong performance in these types of problem scenarios. SVM and KNN were also evaluated. Finally, an artificial neural network classifier was developed. All these models were trained using Python and the scikit-learn (sklearn) library. Scikit-learn, an open-source library, enjoys widespread adoption in machine learning due to its extensive array of tools and user-friendly nature.

Chapter 5: Results and Discussion

5.1 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a critical process in data science and machine learning that involves summarising and visualizing the main characteristics of a dataset. EDA helps understand the underlying structure of the data, detect anomalies, test hypotheses, and check assumptions with the help of statistical graphics and plots. Before applying any machine learning algorithms, it is an essential step, as it provides valuable insights and helps make informed decisions about data preprocessing and model selection.

In this study, EDA was performed on the cleaned dataset. Initially, the dataset consisted of 75,000 data points. The phase information was calculated using Thermo-Calc, and design parameters were computed using Python code. After data cleaning, which reduced the dataset size to 54,056, EDA was conducted on this cleaned dataset to identify patterns and insights. This analysis was crucial for understanding the distribution of phases and the relationships between different design parameters, ensuring the data was well-prepared for subsequent machine learning modeling.

A Pearson correlation plot was generated to show how different features of the data are correlated with each other. Pearson correlation coefficient measures the linear correlation between two variables, providing a value between -1 and 1. The formula for the Pearson correlation coefficient (r) is:

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2 \sum(y_i - \bar{y})^2}}$$

Displayed below is the Pearson correlation plot, which illustrates the correlation coefficients among the different features within the dataset. Each cell within the plot denotes the correlation between two

features, where values near 1 signify a robust positive correlation, values near -1 indicate a strong negative correlation, and values approximately 0 suggest no linear correlation.

Figure 15: Pearson's correlation coefficient

The plot displays the correlations among eight features: *ΔSmix*, VEC, *ΔHmix*, *δ*, Δχ, Tm, Ω, and Class. Our target feature is 'Class'. Notably, the 'Class' feature has strong correlations with several other features, such as VEC (0.84), *δ*(-0.81), *ΔHmix* (0.70), and Ω (0.70). This indicates that changes in these features are closely associated with changes in the 'Class' feature.

5.1.1 Correlation with Target Feature ('Class')

- **VEC** shows a strong positive correlation with 'Class' (0.84), indicating that the phase is likely to change from BCC to FCC correspondingly as the Valence Electron Concentration increases.

- **Atomic size difference (δ)** shows a strong negative correlation with ‘Class’, which means as δ increases, the probability of BCC phases increases.

- **ΔHmix** (enthalpy of mixing) also has a significant positive correlation with 'Class' (0.70), suggesting a strong relationship between the enthalpy of mixing and the class of the material.

- **Ω (omega parameter)** positively correlates with 'Class' (0.70), showing that this parameter is also a significant predictor for the class.

- Interestingly, ΔSmix did not show a strong correlation with ‘Class’, so it will not impact the model much.

5.1.2 Correlation Between the Features

- A strong positive correlation between $\Delta Hmix$ and Ω (0.97), indicating that these two features increase together. This also suggests that dropping Ω will not affect the model's performance due to a strong correlation with $\Delta Hmix$.

- A strong negative correlation between $\Delta Hmix$ and δ (-0.94), suggesting an inverse relationship between the enthalpy of mixing and atomic size difference.

5.1.3 Pattern Analysis in Features

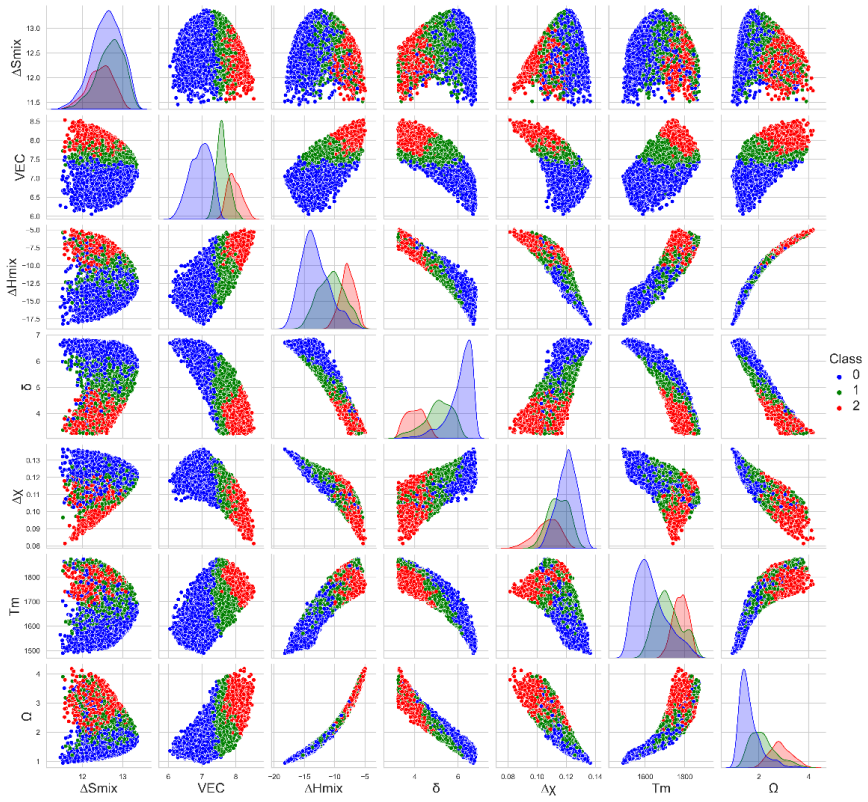

Figure 16: Scatterplot of input features for three phases. BCC:0, BCC+FCC: 1 and FCC: 2

This scatterplot matrix provides a detailed visual representation of the relationships between different features used for model training, specifically focusing on the phases of the materials (BCC, BCC + FCC, and FCC). Here are the detailed observations for the important features: VEC, $\Delta Hmix$, δ, and $\Delta\chi$.

Valence Electron Concentration (VEC)

- **VEC Distribution:** The histogram for VEC shows that Class 0 (BCC) predominantly occupies the lower range of VEC values, Class 2 (FCC) occupies the higher range, and Class 1 (BCC + FCC) is distributed in the middle range.

- **VEC vs. $\Delta Hmix$:** There is a clear trend where higher VEC values are associated with higher $\Delta Hmix$ values. The scatterplot shows

that Class 0 (BCC) is mostly found at lower VEC and $\Delta Hmix$ values, while Class 2 (FCC) is found at higher values, with Class 1 (BCC + FCC) in between. It is the most significant pattern for the separation of the classes.

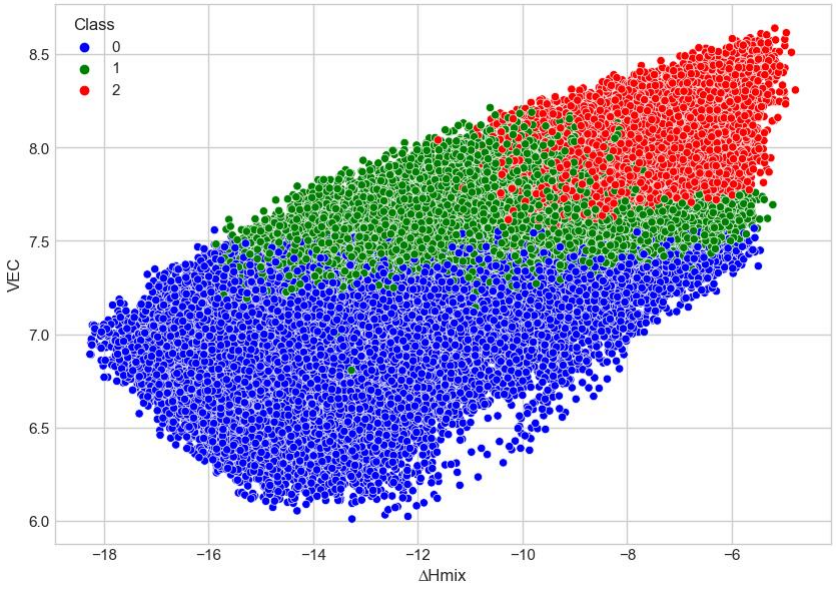

Figure 17: Scatterplot of ΔHmix vs VEC

- **VEC vs. δ:** The scatterplot indicates that lower VEC values are associated with higher δ values for Class 0 (BCC), while higher VEC values correspond to lower δ values for Class 2 (FCC). Class 1 (BCC + FCC) falls in between. A good separation between the three classes leads to a good model.

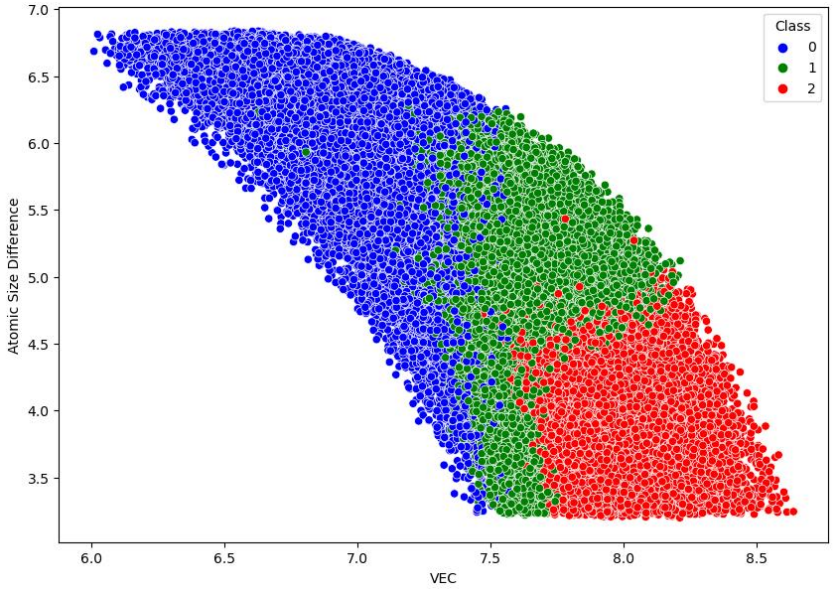

Figure 18: Scatterplot of VEC vs Atomic Size Difference (δ)

Enthalpy of Mixing (*ΔHmix*)

- ***ΔHmix* Distribution:** The histogram for *ΔHmix* shows that Class 0 (BCC) is primarily associated with lower *ΔHmix* values, Class 2 (FCC) with higher values, and Class 1 (BCC + FCC) in the middle range.

- ***ΔHmix* vs. *δ*:** The scatterplot demonstrates that Class 0 (BCC) tends to have higher *δ* values at lower *ΔHmix*, while Class 2 (FCC) has lower *δ* values at higher *ΔHmix*. Class 1 (BCC + FCC) is positioned in the middle.

- ***ΔHmix* vs. Δχ:** The scatterplot shows that Class 0 (BCC) is clustered at higher Δχ values with lower *ΔHmix*, whereas Class 2 (FCC) is found at lower Δχ values with higher *ΔHmix*. Class 1 (BCC + FCC) lies in between these ranges.

Atomic Size Difference (*δ*)

- ***δ* Distribution:** The histogram for *δ* reveals that Class 0 (BCC) is associated with higher *δ* values, Class 2 (FCC) with lower *δ*

values, and Class 1 (BCC + FCC) is spread across the middle range.

- **δ vs. $\Delta\chi$:** The scatterplot indicates that higher δ values correspond to higher $\Delta\chi$ values for Class 0 (BCC), while lower δ values correspond to lower $\Delta\chi$ values for Class 2 (FCC). Class 1 (BCC + FCC) again falls in the intermediate range.

Electronegativity Difference ($\Delta\chi$)

- **$\Delta\chi$ Distribution:** The histogram for $\Delta\chi$ shows that Class 0 (BCC) is generally associated with higher $\Delta\chi$ values, Class 2 (FCC) with lower values, and Class 1 (BCC + FCC) is in between.

- **$\Delta\chi$ vs. Tm:** The scatterplot shows a pattern where Class 0 (BCC) has higher $\Delta\chi$ and a wider range of Tm values, whereas Class 2 (FCC) shows lower $\Delta\chi$ with a narrower range of Tm values. Class 1 (BCC + FCC) is distributed between these ranges.

The scatterplot matrix reveals distinct patterns for each class based on the features VEC, $\Delta Hmix$, δ, and $\Delta\chi$:

- Class 0 (BCC) tends to have lower VEC and $\Delta Hmix$ values, higher δ values, and higher $\Delta\chi$ values.

- Class 2 (FCC) generally exhibits higher VEC and $\Delta Hmix$ values, lower δ values, and lower $\Delta\chi$ values.

- Class 1 (BCC + FCC) falls between Classes 0 and 2 regarding these feature values.

5.2 Feature Importance and Selection

Feature importance is a crucial aspect of machine learning that involves identifying which features (or variables) in a dataset have the most significant impact on the output or prediction of a model. Understanding feature importance helps in improving the model's performance, making it more interpretable, and providing insights into the underlying data.

5.2.1 Feature importance is used for several reasons

- **Model Interpretation:** It helps in understanding the model by highlighting the most influential features.
- **Dimensionality Reduction:** By identifying and retaining only the important features, we can reduce the complexity of the model, leading to faster training times and potentially better performance.
- **Improving Model Performance:** By focusing on the most relevant features, the model can make more accurate predictions.
- **Insight Generation:** It provides insights into the data, helping in understanding the relationships between different features and the target variable.

5.2.2 Methods to Calculate Feature Importance

There are various methods to calculate feature importance, including:

- **Permutation Feature Importance**

1. This technique entails rearranging the values of each feature and observing the resultant drop in the model's performance. A notable drop signifies that the feature holds importance.

2. Permutation feature importance is model-agnostic, meaning it can be applied to any machine learning model.

- **Tree-Based Feature Importance**

    1. Tree-based models, including Decision Trees, Random Forests, and Gradient Boosting, determine feature importance scores by assessing the degree to which each feature aids in diminishing impurity within the tree nodes. This impurity reduction is typically measured using metrics such as Gini impurity or entropy.

    2. This method is specific to tree-based algorithms and is calculated during the model training process.

- **Coefficients as Feature Importance:**

    1. In linear models, the absolute values of the coefficients can indicate feature importance. Larger coefficients represent more important features.

    2. This method is specific to linear models like Linear Regression and Logistic Regression.

- **SHAP Values**

    1. These values deliver a reliable assessment of feature importance by accounting for the individual contribution of each feature across a spectrum of model predictions.

    2. SHAP values are model-agnostic and offer a high level of interpretability.

5.2.3 Permutation Feature Importance

The Feature importance has been calculated for the dataset. Fig 5.5 shows a bar chart for each feature's importance based on the permutation method.

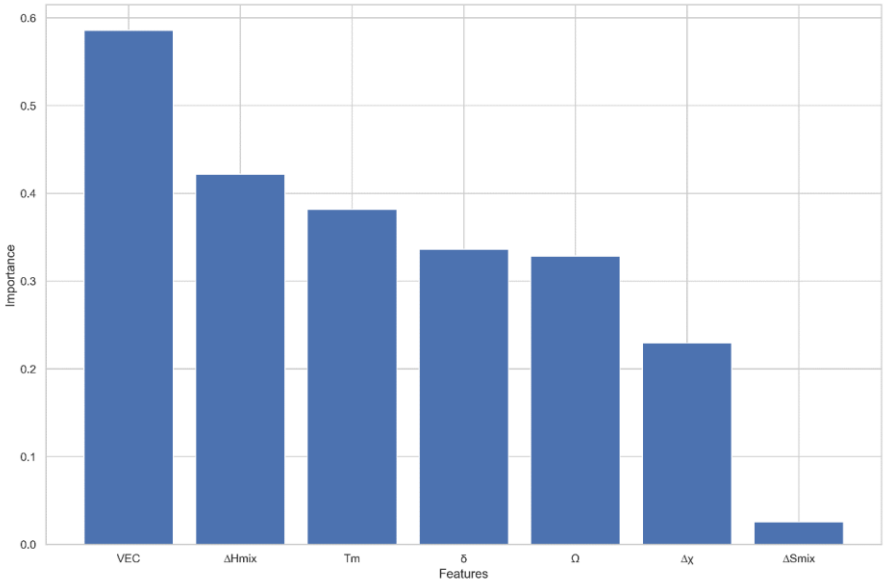

Figure 19: Permutation Feature Importance

- **VEC:** VEC emerges as the most pivotal feature, boasting an importance score nearing 0.6. This underscores the substantial influence of VEC on the model's predictions.
- ***ΔHmix*:** *ΔHmix* is the second most important feature, with an importance score of around 0.45, highlighting its significant role in predicting the target variable.
- **Tm:** Tm is also an important feature, with a score slightly below *ΔHmix*, emphasising its relevance in the model.
- ***δ*:** *δ* shows considerable importance, followed closely by the Omega parameter (Ω), both having scores around 0.35.
- **Δ:** Δχ has a moderate importance score, indicating its influence on the model, although it is less critical than VEC and *ΔHmix*.

- ***ΔSmix*: *ΔSmix*** is found to be the least important feature, with an importance score well below the other features.

5.2.4 Feature Selection

Following the assessment of feature importance, feature selection was conducted to ascertain the optimal number of features for optimal model performance. The Artificial Neural Network (ANN) model was initially trained using all features, yielding an F1 score of 98.144. Subsequently, in each iteration, one feature was omitted, and the F1 score was recalculated to evaluate the influence of each feature on the model's performance.

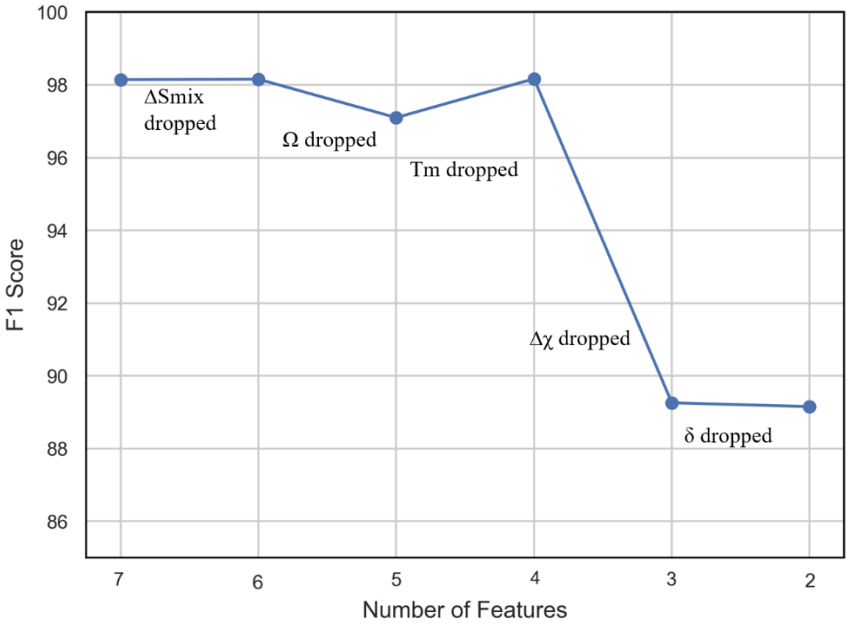

Figure 20: Impact of Sequential Feature Dropping on Model F1 Score

The observations from the feature selection process are as follows: Removing the least important feature, *ΔSmix*, slightly improved the F1 score to 98.16, indicating that *ΔSmix* had minimal impact on the model's performance. After dropping the Omega parameter (Ω), the F1 score decreased to 97.09, suggesting that Ω has a moderate impact. Interestingly, removing the melting temperature (Tm) resulted in a slight increase in the F1 score to 98.17, indicating that Tm may not be crucial for the model's performance. However, eliminating the electronegativity

difference ($\Delta\chi$) caused a significant drop in the F1 score to 89.25, highlighting that $\Delta\chi$ is an important feature for the model's accuracy. Removing the atomic size difference (δ) led to a further drop in the F1 score to 89.1513, confirming that δ is also a key feature.

Feature selection plays a pivotal role in enhancing model performance by eliminating irrelevant or less significant features. This enables the model to concentrate on the most impactful variables, resulting in improved accuracy and performance. Furthermore, feature selection aids in mitigating overfitting since utilizing fewer features reduces the model's complexity, enhancing its ability to generalize to new data. Moreover, a model with fewer features is more interpretable and comprehensible, rendering it more practical for real-world applications. Finally, the reduction in the number of features diminishes the computational burden and training time required for the model.

5.3 Model Evaluation

5.3.1 Evaluation Metrics

In classification problems, evaluating the performance of a model is crucial to understanding how well it predicts the classes of new data. A confusion matrix is an essential tool for this evaluation. It provides a detailed summary of the prediction results by comparing the actual and predicted class labels.

- **Confusion Matrix**

A confusion matrix is a table summarizing a classification algorithm's performance by comparing predicted and actual labels. It provides a detailed breakdown of correct and incorrect classifications and helps calculate various evaluation metrics.

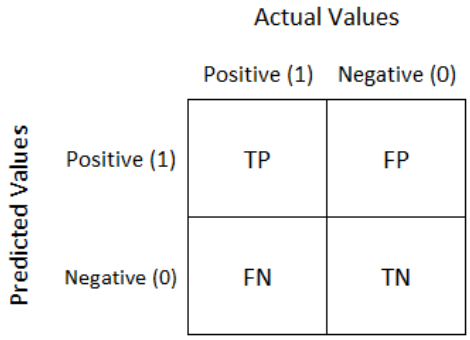

Figure 21: Structure of Confusion matrix

Definitions:

In the context of the confusion matrix structure, four key terminologies are defined. It is important to note that the terms "positive" and "negative" refer to the classes rather than the actual nature of the data. This matrix is typically used for binary classification, and a 3x3 matrix is employed for ternary classification. For simplicity, we use positive = 1 and negative = 0.

1. **True Positive (TP)**: This occurs when the actual and predicted values are 1.
2. **False Positive (FP)**: This occurs when the actual value is 0, but the predicted value is 1.
3. **False Negative (FN)**: This occurs when the actual value is 1, but the predicted value is 0.
4. **True Negative (TN)**: This occurs when both the actual and predicted values are 0.

Accuracy

Accuracy formula:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

While accuracy is easy to compute and understand, it may not be an ideal metric for imbalanced datasets, where one class significantly outweighs the others.

Precision

Precision is calculated as:

$$Precision = \frac{TP}{TP + FP}$$

Recall

Recall is calculated as:

$$Recall = \frac{TP}{TP + FN}$$

F1 Score

The F1 score is calculated as:

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

The F1 score ranges from 0 to 1, 1 indicating perfect precision and recall. It is a robust metric for evaluating models on imbalanced datasets, considering false positives and negatives.

5.3.2 Machine Learning Model Performance

For machine learning modeling, a final dataset comprising ΔHmix, VEC, $\Delta\chi$, and δ was employed to train multiple machine learning models using the scikit-learn library. The utilized models included DTC, SVC, LR, KNN, RFC, and ANN. The F1 scores of these models are depicted in Figure 23.

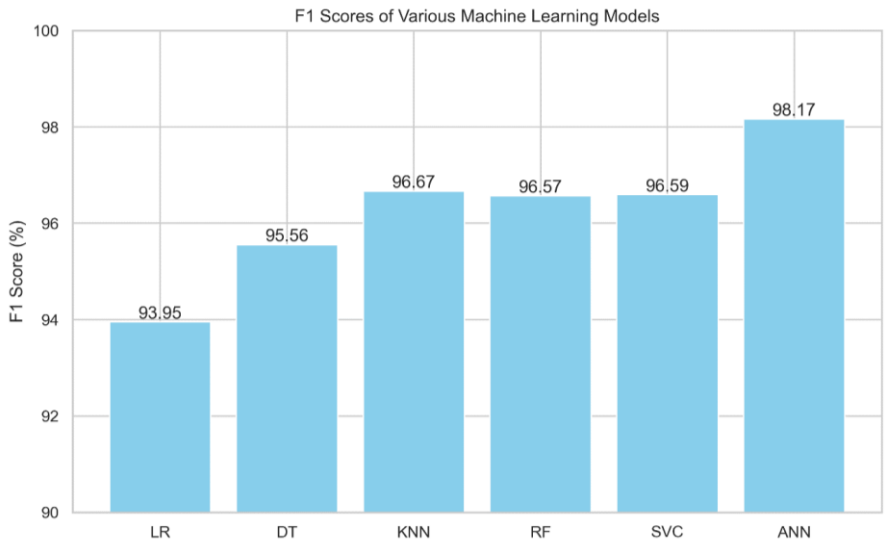

Figure 22: F1 Score for different machine learning models

The ANN model outperformed all the other models, achieving the highest F1 score. The excellent scores of the ANN can be viewed as its ability to capture the complex non-linear pattern in the data. ANN models are highly flexible and can model intricate patterns and interactions between features that simpler models might miss.

5.3.2.1 Hyperparameters for ANN:

- random_state = 1: This parameter determines what seed is used to create a random value, ensuring reproducibility of results in random processes.
- max_iter = 300: It is the maximum number of iterations the optimization algorithms will go through in iterative algorithms, such as gradient descent, for convergence in fitting the model.
- hidden_layer_sizes = (100,): It defines the number of neurons in each hidden layer of the neural network. In this architecture, there is a single hidden layer of 100 neurons.
- ctivation = 'relu': This parameter specifies the activation function used in neural networks. 'relu' represents Rectified Linear Unit, which is a very common activation used in deep learning models.

- solver = 'adam': This is an argument that defines the optimization algorithm which would update, in the training process, the weights and biases of the neural network. In other words, 'adam' stands for adaptive moment estimation. The optimization algorithm is quite effective when training with deep networks.
- alpha = 0.0001: This parameter represents the regularization strength applied to the neural network. It helps prevent overfitting by penalizing large weight values.

Combining these hyperparameters ensures that the ANN model is trained effectively, balancing convergence speed and thoroughness. The high F1 score indicates that the ANN model successfully captured the underlying patterns in the data, leading to accurate predictions and superior performance compared to other models.

5.3.2.2 Confusion Matrix Observations

The confusion matrices for the various machine learning models provide detailed insights into the classification performance of each model on the dataset. The models evaluated include LR, DTC, KNN, RFC, SVC, and ANN.

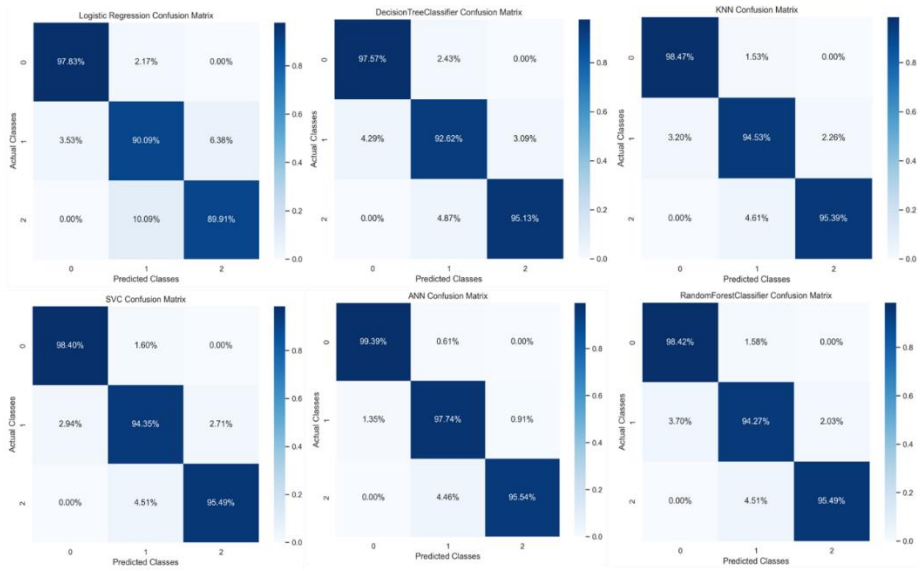

Figure 23: Confusion matrix for all models

The confusion matrices for the various machine learning models provide detailed insights into their classification performance on the dataset. Logistic Regression correctly classified 97.83% of Class 0 instances, with 2.17% misclassified as Class 1. For Class 1, 90.09% were correctly classified, with 3.53% misclassified as Class 0 and 6.38% as Class 2. Class 2 saw 89.91% correctly classified, with 10.09% misclassified as Class 1. Logistic Regression performs well but struggles to distinguish between Class 1 and Class 2. The DTC correctly classified 97.57% of Class 0 instances, with 2.43% misclassified as Class 1. For Class 1, 92.62% were correctly classified, with 4.29% misclassified as Class 0 and 3.09% as Class 2. Class 2 saw 95.13% correctly classified, with 4.87% misclassified as Class 1. The Decision Tree Classifier performs better than Logistic Regression, especially for Class 2.

KNN correctly classified 98.47% of Class 0 instances, with 1.53% misclassified as Class 1. For Class 1, 94.53% were correctly classified,

with 3.20% misclassified as Class 0 and 2.26% as Class 2. Class 2 saw 95.39% correctly classified, with 4.61% misclassified as Class 1. KNN shows improved performance for all classes compared to Logistic Regression and Decision Tree. The Random Forest Classifier correctly classified 98.42% of Class 0 instances, with 1.58% misclassified as Class 1. For Class 1, 94.27% were correctly classified, with 3.70% misclassified as Class 0 and 2.03% as Class 2. Class 2 saw 95.49% correctly classified, with 4.51% misclassified as Class 1. Random Forest performs similarly to KNN, with slight improvements in certain classifications.

The SVC correctly classified 98.40% of Class 0 instances, with 1.60% misclassified as Class 1. For Class 1, 94.35% were correctly classified, with 2.94% misclassified as Class 0 and 2.71% as Class 2. Class 2 saw 95.49% correctly classified, with 4.51% misclassified as Class 1. SVC shows performance comparable to KNN and Random Forest, maintaining high accuracy across all classes. The ANN correctly classified 99.39% of Class 0 instances, with 0.61% misclassified as Class 1. For Class 1, 97.74% were correctly classified, with 1.35% misclassified as Class 0 and 0.91% as Class 2. Class 2 saw 95.54% correctly classified, with 4.46% misclassified as Class 1. ANN outperforms all other models, showing the highest classification accuracy for Class 0 and Class 1 and maintaining strong performance for Class 2.

The confusion matrices indicate that the ANN model provides the best overall performance, achieving the highest accuracy for Class 0 and Class 1 and competitive accuracy for Class 2. KNN, Random Forest, and SVC also perform well, with slight variations in misclassification rates. While effective, Logistic Regression and Decision Tree show comparatively higher misclassification rates, particularly between Class 1 and 2. These observations highlight the superior capability of ANN in capturing complex patterns and relationships within the dataset.

5.4 Model Verification with Literature Data

To verify the accuracy of the ANN model in real-life scenarios, reported data from the literature were collected. This dataset included 52 unique compositions taken from 25 different sources. A comparison was made between the experimentally reported phases and the phases predicted by the model. The detailed comparison is shown in the table below.

Table 3: Model verification with experimentally reported data from the literature.

Molar Compositions	**Reported Phases**	**Predicted Phases**	**Ref.**
$Al1_{0.7}CoCrFeNi_{2.1}$	BCC	BCC	[58]
$Al1_{0.5}CoCrFeNi_{2.1}$	BCC	BCC+FCC	[58]
$Al_{0.9}CoCrFeNi_{2.1}$	BCC+FCC	BCC+FCC	[58]
$Al_{0.7}CoCrFeNi_{2.1}$	BCC+FCC	FCC	[58]
$Al1_{0.3}CoCrFeNi_{2.1}$	BCC+FCC	BCC+FCC	[58]
$Al1_{0.1}CoCrFeNi_{2.1}$	BCC+FCC	BCC+FCC	[58]
$Al_{0.5}CoCrFeNi_{2.1}$	FCC	FCC	[58]
$Al1_{0.5}CoCrFeNi$	BCC	BCC	[59]
$Al_{2.5}CoCrFeNi$	BCC	BCC	[59]
$AlCoCrFeNi_{2.1}$	BCC+FCC	BCC+FCC	[60]
$AlCo_{1.3}Cr_{0.7}Fe_{0.9}Ni_{1.9}$	BCC+FCC	BCC+FCC	[61]
$AlCo_{1.7}Cr_{0.8}Fe_{0.8}Ni_{1.5}$	BCC+FCC	BCC+FCC	[62]
$Al_{3.0}CoCrFeNi$	BCC	BCC	[59]
$Al_{2.0}CoCrFeNi$	BCC	BCC	[63]
$Al_{0.7}CoCrFeNi$	BCC+FCC	BCC+FCC	[63]
$Al_{0.3}CoCrFeNi$	FCC	FCC	[63]
$Al_{0.4}CoCrFeNi$	FCC	FCC	[63]
$Al_{0.5}CoCrFeNi$	BCC+FCC	BCC+FCC	[64]
$AlCoCrFeNi$	BCC+FCC	BCC	[65]
$Al_{0.5}Co_{1.5}CrFe_{1.8}Ni_{0.2}$	BCC+FCC	BCC	[65]
$AlCo_{2.0}CrFe_{2.7}Ni_{0.3}$	BCC+FCC	BCC	[65]

$AlCo_{1.9}CrFeNi$	BCC+FCC	BCC+FCC	[66]
$AlCo_{2.6}Cr_{0.9}Fe_{0.6}Ni_{1.1}$	BCC+FCC	BCC+FCC	[67]
$AlCo_{1.2}Cr_{0.7}Fe_{0.9}Ni_{1.9}$	BCC+FCC	BCC+FCC	[61]
$AlCo_{0.3}CrFe_{2.0}Ni_{1.7}$	BCC+FCC	BCC+FCC	[68]
$AlCoCr_{0.2}FeNi_{2.1}$	BCC+FCC	BCC+FCC	[69]
$AlCoCr_{0.5}FeNi_{2.1}$	BCC+FCC	BCC+FCC	[69]
$AlCoCr_{0.8}FeNi_{2.1}$	BCC+FCC	BCC+FCC	[69]
$AlCo_{1.7}Cr_{0.8}Fe_{0.8}Ni_{1.5}$	BCC+FCC	BCC+FCC	[62]
$Al_{1.3}Co_{0.9}Cr_{0.7}FeNi_{3.2}$	BCC+FCC	BCC+FCC	[70]
$AlCo_{1.3}Cr_{0.4}Fe_{0.9}Ni_{1.9}$	BCC+FCC	BCC+FCC	[71]
$AlCo_{1.9}Cr_{0.6}Fe_{0.4}Ni_{1.6}$	BCC+FCC	BCC+FCC	[71]
$AlCoCrFeNi_{2.2}$	BCC+FCC	BCC+FCC	[72]
$Al_{0.6}CoCrFeNi$	BCC+FCC	BCC+FCC	[73]
$AlCo_{1.8}Cr_{0.2}Fe_{0.2}Ni_{1.8}$	BCC+FCC	BCC+FCC	[74]
$AlCo_{1.7}Cr_{0.6}Fe_{0.6}Ni_{1.7}$	BCC+FCC	BCC+FCC	[75]
$AlCo_{0.6}CrFe_{1.4}Ni_{2.1}$	BCC+FCC	BCC+FCC	[76]
$AlCo_{0.8}CrFeNi_{2.3}$	BCC+FCC	BCC+FCC	[76]
$AlCoCr_{0.8}Fe_{1.2}Ni_{2.1}$	BCC+FCC	BCC+FCC	[76]
$AlCoCr_{1.2}Fe_{0.8}Ni_{2.1}$	BCC+FCC	BCC+FCC	[76]
$AlCo_{1.2}Cr_{0.8}FeNi_{2.1}$	BCC+FCC	BCC+FCC	[76]
$AlCo_{1.2}CrFe_{0.8}Ni_{2.1}$	BCC+FCC	BCC+FCC	[76]
$AlCo_{1.2}CrFeNi_{1.9}$	BCC+FCC	BCC+FCC	[76]
$AlCo_{0.6}CrFeNi_{2.4}$	BCC+FCC	BCC+FCC	[77]
$AlCo_{0.8}CrFeNi_{2.2}$	BCC+FCC	BCC+FCC	[77]
$AlCo_{0.2}CrFeNi_{2.8}$	BCC+FCC	BCC+FCC	[77]
$AlCo_{0.4}CrFeNi_{2.6}$	BCC+FCC	BCC+FCC	[77]
$Al_{1.2}CoCrFeNi_{2.8}$	BCC+FCC	BCC+FCC	[78]
$AlCo_{0.8}Cr_{0.8}Fe_{1.7}Ni_{1.5}$	BCC+FCC	BCC+FCC	[78]
$AlCo_{1.7}Cr_{0.6}Fe_{0.6}Ni_{1.8}$	BCC+FCC	BCC+FCC	[79]
$Al_{0.1}CoCrFeNi$	FCC	FCC	[80]
$Al_{0.2}CoCrFeNi$	FCC	FCC	[81]

The ANN model correctly predicted the phases for 47 out of 52 compositions, resulting in an accuracy of 90%. This high accuracy demonstrates the model's robustness in predicting the phases of alloys based on the provided compositions.

5.5 Model Accuracy with CALPHAD Data

Further model accuracy was tested with CALPHAD data for alloy composition ranges beyond 35%. In this testing, the dataset included compositions where each element ranged from 5% to 80%. This dataset does not adhere to the traditional definition of high entropy alloys but was used to assess the model's performance outside the training range. The training data ranged from 5% to 35% for all elements. This extended testing explored the relationship between phases formed and design parameters. The model successfully identified the patterns with 94.83% accuracy. A total of 5,830 samples were used for this testing, and the phases were calculated using Thermo-Calc.

These results demonstrate the model's strong predictive capabilities within and beyond the initial training range, showcasing its potential for broader applications in alloy phase prediction.

5.6 Experimental Verification of Machine Learning Model

5.6.1 Selection of Composition

To experimentally validate the machine learning design approach followed in this thesis, the composition of $Al_{15}Co_5Cr_5Fe_{45}Ni_{30}$ has been chosen in the Al-Co-Cr-Fe-Ni system. This specific composition was selected primarily because it lies outside the elemental range considered for Latin Hypercube Sampling (LHS) but aligns with the phase formation rules derived from the machine learning results, as shown in Table 5.2. Additionally, it has been noted that to achieve higher ductility in the Al-Co-Cr-Fe-Ni system, the FCC phase percentage should be greater than that of the BCC phase in a two-phase microstructure. This can be achieved through higher contents of either Co, Ni, or Fe. Due to the high cost of Co and Ni, Fe-rich (45%) composition has been considered, which also contributes to making the designed alloy lightweight.

Drawing inspiration from $Al_{18}Co_{13}Cr_{10}Fe_{14}Ni_{45}$ reported by Zhang *et al.* [70], which exhibits a yield strength of 623 MPa with more than 7% ductility, the composition $Al_{15}Co_5Cr_5Fe_{45}Ni_{30}$ has been designed by carefully adjusting each element to fit within the phase formation rules.

Table 4: Phase formation rules for BCC plus FCC dual-phase alloys

SN	**Design Parameter**	**Min**	**Max**	**$Al_{15}Co_5Cr_5Fe_{45}Ni_{30}$ Alloy**
1	VEC	6.6	8.21	7.8
2	Hmix	-16.64	-5.22	-9.384
3	Smix	11.43	13.36	10.84695596
4	δ	3.21	6.39	5.287434443
5	$\Delta\chi$	0.094	0.134	0.102591423
6	Tm	1564.05	1876.41	1670.76555
7	Ω	1.15	4.04	1.931236183

5.6.2 X-Ray Diffraction (XRD) Analysis

XRD pattern of the AlCoCrFeNi high-entropy alloy was obtained using the Panalytical Empyrean instrument with Cu Kα radiation (λ = 1.54 Å). The XRD pattern is shown in Figure 5.10.

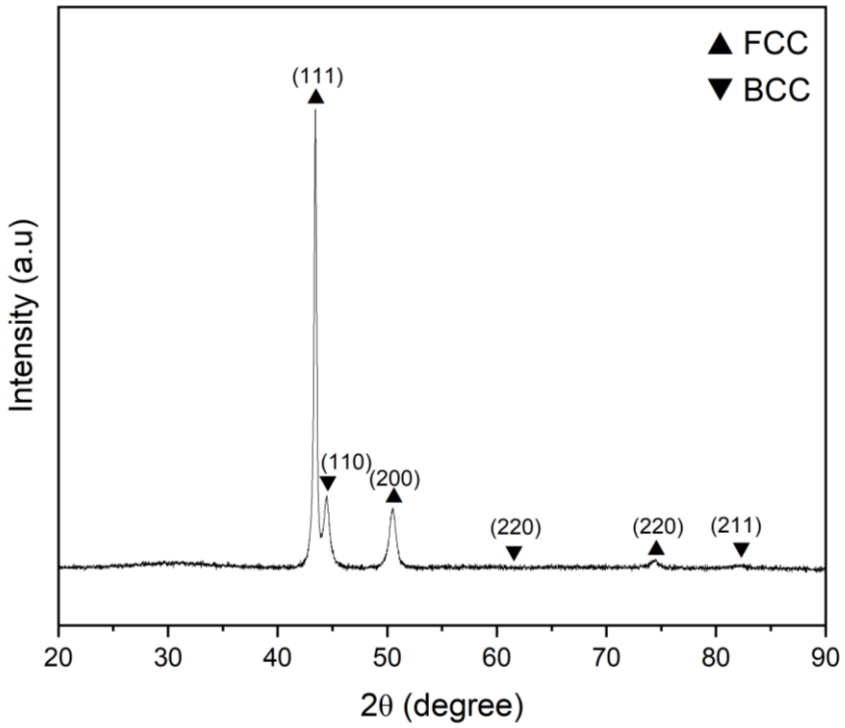

Figure 24: XRD analysis of the as-cast sample

By analyzing the XRD pattern, it is evident that the sample consists of two distinct phases: Body-Centered Cubic (BCC) and Face-Centered Cubic (FCC). The identification of these phases was based on the position and intensity of the diffraction peaks.

FCC Phase:

- The prominent peak at approximately 2θ = 43.5° corresponds to the (111) plane of the FCC structure. This peak strongly indicates the FCC phase presence in the alloy.
- Additional peaks at around 50.5° and 74° are indexed as (200) and (220) planes, respectively, confirming the presence of the FCC phase.

BCC Phase:

- The peaks observed at 2θ values of approximately 44.5°, 64°, and 82° correspond to the (110), (200), and (211) planes of the BCC structure.
- The presence of these peaks distinctly indicates the BCC phase in the alloy.

The clear separation and identification of peaks corresponding to BCC and FCC structures confirm that the AlCoCrFeNi alloy exists as a dual-phase material under the studied conditions.

5.6.3 Differential Scanning Calorimetry (DSC)

DSC analysis was performed to investigate phase transitions in the as-cast sample. The DSC curve, depicted in Figure 5.11, illustrates the heat flow as a function of temperature, elucidating the alloy's thermal characteristics throughout heating and cooling cycles.

Heating Cycle

During the heating cycle, the DSC curve shows two distinct endothermic peaks.

1. **FCC Phase Transformation:**
    - The first endothermic peak, appearing at a lower temperature (around 1350°C), corresponds to the transformation of the FCC phase. This peak represents the energy absorbed as the FCC phase undergoes a phase transition, possibly melting or dissolving into the BCC phase.
2. **BCC Phase Transformation:**
    - The second endothermic peak, at a slightly higher temperature (around 1400°C), corresponds to the transformation of the BCC phase. This peak indicates the energy absorbed during the phase transition of the BCC

phase, possibly melting or transitioning to another structure.

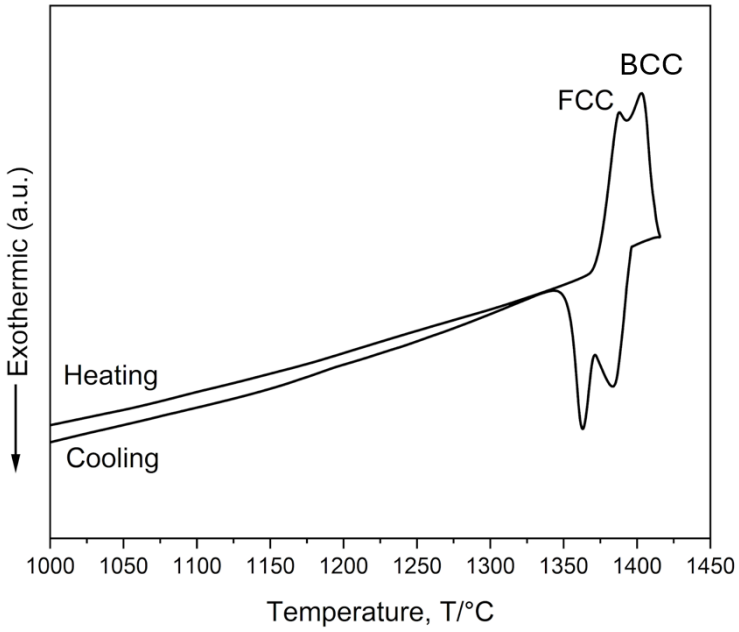

Figure 25: DSC Analysis of as-cast sample

Cooling Cycle

During the cooling cycle, the DSC curve shows two distinct exothermic peaks, mirroring the endothermic events observed during heating.

1. **BCC Phase Reformation:**

    - The first exothermic peak, appearing at a higher temperature during cooling (around 1400°C), corresponds to the reformation of the BCC phase. This peak indicates the release of energy as the BCC phase solidifies or reforms from the melt or a different phase.

2. **FCC Phase Reformation:**

    - The second exothermic peak, at a slightly lower temperature (around 1350°C), corresponds to the reformation of the FCC phase. This peak signifies the release of energy as the FCC phase solidifies or reforms from the BCC phase or the melt.

5.6.4 Optical Microscopy Analysis

The microstructure of the as-cast sample underwent analysis using an optical microscope (ZEISS Axio Vert. A). The optical micrograph, displayed in Figure [X], unveils the dual-phase composition of the alloy, comprising both Face-Centered Cubic (FCC) and Body-Centered Cubic (BCC) phases. The distinct labeling of the phases is evident in the image.

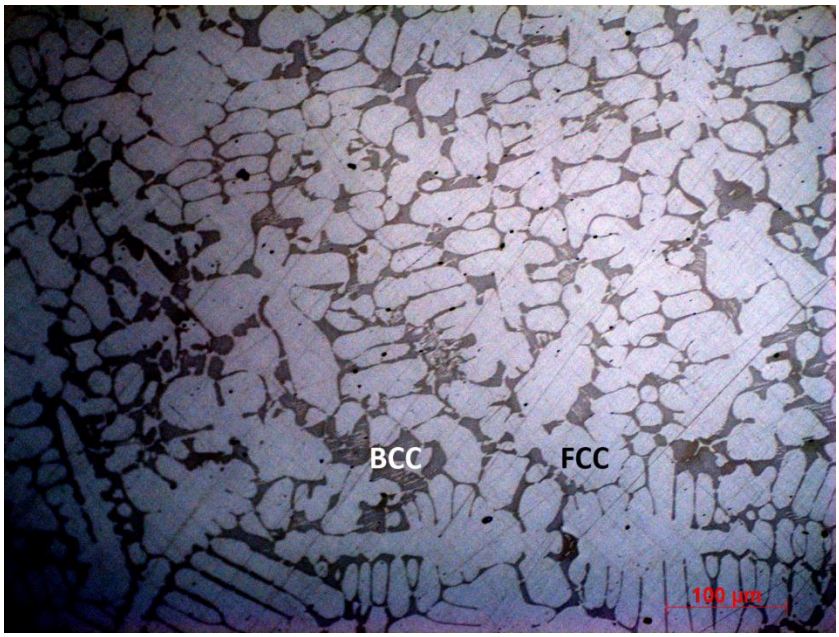

Figure 26: Optical image of the as-cast sample

Phase Distribution

• **FCC Phase:** The predominant phase observed in the micrograph, the FCC phase, permeates the structure extensively and constitutes the bulk

of the area. It exhibits larger grains with a more consistent shape than the BCC phase, and it forms a uniform distribution throughout.

• **BCC Phase:** Identified as the secondary phase, the BCC phase is dispersed within the FCC matrix. Comprising smaller grains of irregular shapes, these grains are less uniform and tend to aggregate into smaller clusters in comparison to the FCC phase.

Microstructural Observations

- **Dendritic Structure:** The micrograph clearly shows a dendritic structure, indicative of the solidification pattern of the alloy. The dendritic arms are primarily composed of the FCC phase, with the BCC phase filling the interdendritic regions.
- **Grain Boundaries:** The grain boundaries between the FCC and BCC phases are well-defined. The interface between the two phases is sharp, suggesting good phase separation and minimal interdiffusion.
- **Phase Fraction:** By visual inspection, it is evident that the FCC phase constitutes the majority of the phase fraction, while the BCC phase makes up a smaller portion. This phase distribution is consistent with the expected dual-phase nature of the alloy.

Chapter 6: Summary and Future Directions

This thesis focuses on developing a machine-learning model to predict phase formation in high-entropy alloys (HEAs), specifically within the Al-Co-Cr-Fe-Ni system. Latin Hypercube Sampling (LHS) achieved comprehensive compositional coverage, generating a robust dataset of 75,000 samples. These samples underwent thermodynamic calculations via the TC-Python API, resulting in labeled data detailing the phases formed and their quantities. This process culminated in a dataset comprising 75,000 labeled data points.

Post-cleaning, which excluded data points with more than two phases, 54,056 data points remained, categorized into BCC (27,189), BCC+FCC (16,850), and FCC (10,017). Subsequent preprocessing and scaling prepared the data for machine learning model training, which included SVM, Random Forest, Decision Tree, Logistic Regression, KNN, and ANN.

EDA offered critical insights into the dataset, identifying significant correlations and patterns. Through Pearson correlation and permutation feature importance analyses, four key features were identified: ΔHmix, $\Delta\chi$, VEC, and δ were crucial factors in both model training and evaluation.

The model's performance was assessed using the F1 score metric. The ANN showed the most outstanding performance, with an F1 score of 98.17. KNN, SVM, and Random Forest followed closely, all scoring above 96.5%. Logistic Regression and Decision Tree models also demonstrated strong performance with F1 scores above 90%. The ANN model was selected for future predictions and verified against literature-reported data, correctly predicting 47 out of 52 compositions and achieving an accuracy above 90%.

The study's primary objective was to develop a low-density, HEA with a dual-phase BCC+FCC structure. The composition

$Al_{15}Co_5Cr_5Fe_{45}Ni_{30}$ was identified and experimentally confirmed to possess the desired dual-phase structure, validated through XRD and DSC analyses, indicating dual-phase composition.

Future research will focus on an in-depth study of the mechanical properties of the developed alloys. Also, on the applicability of the machine learning approach to quickly discover novel compositions across alloy systems with better functional and mechanical properties. Additionally, a web-based, user-friendly interface is being developed to allow users to process both single and multiple compositions via Excel sheets. This interface will enhance the accessibility and application of the developed machine-learning model for HEA phase prediction.

References

[1] J.W. Yeh, S.K. Chen, S.J. Lin, J.Y. Gan, T.S. Chin, T.T. Shun, C.H. Tsau, S.Y. Chang, Nanostructured high-entropy alloys with multiple principal elements: Novel alloy design concepts and outcomes, Adv Eng Mater 6 (2004) 299–303. https://doi.org/10.1002/adem.200300567.

[2] B. Cantor, I.T.H. Chang, P. Knight, A.J.B. Vincent, Microstructural development in equiatomic multicomponent alloys, Materials Science and Engineering: A 375–377 (2004) 213–218. https://doi.org/10.1016/J.MSEA.2003.10.257.

[3] L. Qiao, Y. Liu, J. Zhu, A focused review on machine learning aided high-throughput methods in high entropy alloy, J Alloys Compd 877 (2021). https://doi.org/10.1016/j.jallcom.2021.160295.

[4] J.W. Yeh, Recent progress in high-entropy alloys, Annales de Chimie: Science Des Materiaux 31 (2006) 633–648. https://doi.org/10.3166/acsm.31.633-648.

[5] O.N. Senkov, G.B. Wilks, D.B. Miracle, C.P. Chuang, P.K. Liaw, Refractory high-entropy alloys, Intermetallics (Barking) 18 (2010) 1758–1765. https://doi.org/10.1016/J.INTERMET.2010.05.014.

[6] O.N. Senkov, G.B. Wilks, J.M. Scott, D.B. Miracle, Mechanical properties of Nb25Mo25Ta25W25 and V20Nb20Mo20Ta20W20 refractory high entropy alloys, Intermetallics (Barking) 19 (2011) 698–706. https://doi.org/10.1016/J.INTERMET.2011.01.004.

[7] L. Qiao, Aorigele, Z. Lai, J. Zhu, A promising new class of multi-component alloys with exceptional mechanical properties, J Alloys Compd 847 (2020) 155929. https://doi.org/10.1016/J.JALLCOM.2020.155929.

[8] L. Qiao, A. Bao, Z. Lai, Y. Liu, J. Zhu, Y. Wang, Alloy design and properties optimization of multi-component alloy based on solidification characteristics, Materials Science and Engineering: A 805 (2021) 140576. https://doi.org/10.1016/J.MSEA.2020.140576.

[9] T. Borkar, B. Gwalani, D. Choudhuri, C. V. Mikler, C.J. Yannetta, X. Chen, R. V. Ramanujan, M.J. Styles, M.A. Gibson, R. Banerjee, A combinatorial assessment of AlxCrCuFeNi2 ($0 < x < 1.5$) complex concentrated alloys: Microstructure, microhardness, and magnetic properties, Acta Mater 116 (2016) 63–76. https://doi.org/10.1016/J.ACTAMAT.2016.06.025.

[10] Y. Zhang, Z.P. Lu, S.G. Ma, P.K. Liaw, Z. Tang, Y.Q. Cheng, M.C. Gao, Guidelines in predicting phase formation of high-

entropy alloys, MRS Commun 4 (2014) 57–62. https://doi.org/10.1557/mrc.2014.11.

[11] D.B. Miracle, O.N. Senkov, A critical review of high entropy alloys and related concepts, Acta Mater 122 (2017) 448–511. https://doi.org/10.1016/J.ACTAMAT.2016.08.081.

[12] Z.P. Lu, H. Wang, M.W. Chen, I. Baker, J.W. Yeh, C.T. Liu, T.G. Nieh, An assessment on the future development of high-entropy alloys: Summary from a recent workshop, Intermetallics (Barking) 66 (2015) 67–76. https://doi.org/10.1016/J.INTERMET.2015.06.021.

[13] Q. He, Y. Yang, On lattice distortion in high entropy alloys, Front Mater 5 (2018). https://doi.org/10.3389/fmats.2018.00042.

[14] Y. Zou, S. Maiti, W. Steurer, R. Spolenak, Size-dependent plasticity in an Nb25Mo25Ta25W25 refractory high-entropy alloy, Acta Mater 65 (2014) 85–97. https://doi.org/10.1016/J.ACTAMAT.2013.11.049.

[15] K.Y. Tsai, M.H. Tsai, J.W. Yeh, Sluggish diffusion in Co–Cr–Fe–Mn–Ni high-entropy alloys, Acta Mater 61 (2013) 4887–4897. https://doi.org/10.1016/J.ACTAMAT.2013.04.058.

[16] S. Ranganathan, Alloyed pleasures: Multimetallic cocktails, 2003.

[17] J. Chen, X. Zhou, W. Wang, B. Liu, Y. Lv, W. Yang, D. Xu, Y. Liu, A review on fundamental of high entropy alloys with promising high–temperature properties, J Alloys Compd 760 (2018) 15–30. https://doi.org/10.1016/J.JALLCOM.2018.05.067.

[18] Y. Zhang, X. Yang, P.K. Liaw, Alloy design and properties optimization of high-entropy alloys, JOM 64 (2012) 830–838. https://doi.org/10.1007/s11837-012-0366-5.

[19] W. Kohn, L.J. Sham, PHYSICAL REVIEW Self-Consistent Equations Including Exchange and Correlation Effects*, n.d.

[20] Y. Zeng, M. Man, K. Bai, Y.W. Zhang, Revealing high-fidelity phase selection rules for high entropy alloys: A combined CALPHAD and machine learning study, Mater Des 202 (2021) 109532. https://doi.org/10.1016/J.MATDES.2021.109532.

[21] T. Hastie, R. Tibshirani, J. Friedman, Springer Series in Statistics The Elements of Statistical Learning Data Mining, Inference, and Prediction, n.d.

[22] M. Jordan, J. Kleinberg, B. Schölkopf, Pattern Recognition and Machine Learning, n.d.

[23] K.P. Murphy, Machine learning : a probabilistic perspective, n.d.

[24] Pattern Classification, 2016. https://www.researchgate.net/publication/228058014.

[25] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction Second edition, in progress, n.d.

[26] D.W. Hosmer, Stanley. Lemeshow, R.X. Sturdivant, Applied logistic regression., n.d.

[27] W.Y. Loh, Classification and regression trees, Wiley Interdiscip Rev Data Min Knowl Discov 1 (2011) 14–23. https://doi.org/10.1002/widm.8.

[28] L. Breiman, Random Forests, 2001.

[29] J.H. Friedman, 999 REITZ LECTURE GREEDY FUNCTION APPROXIMATION: A GRADIENT BOOSTING MACHINE 1, 2001.

[30] C. Cortes, V. Vapnik, L. Saitta, Support-Vector Networks Editor, Kluwer Academic Publishers, 1995.

[31] T.M. Cover, P.E. Hart, Approximate formulas for the information transmitted bv a discrete communication channel, 1952.

[32] I. Goodfellow, Y. Bengio, A. Courville, Deep Learning, n.d.

[33] S. Singh, N.K. Katiyar, S. Goel, S.N. Joshi, Phase prediction and experimental realisation of a new high entropy alloy using machine learning, Sci Rep 13 (2023). https://doi.org/10.1038/s41598-023-31461-7.

[34] G. Vazquez, S. Chakravarty, R. Gurrola, R. Arróyave, A deep neural network regressor for phase constitution estimation in the high entropy alloy system Al-Co-Cr-Fe-Mn-Nb-Ni, NPJ Comput Mater 9 (2023). https://doi.org/10.1038/s41524-023-01021-8.

[35] Q. Guo, X. Xu, X. Pei, Z. Duan, P.K. Liaw, H. Hou, Y. Zhao, Predict the phase formation of high-entropy alloys by compositions, Journal of Materials Research and Technology 22 (2023) 3331–3339. https://doi.org/10.1016/J.JMRT.2022.12.143.

[36] O. Elgack, B. Almomani, J. Syarif, M. Elazab, M. Irshaid, M. Al-Shabi, Molecular dynamics simulation and machine learning-based analysis for predicting tensile properties of high-entropy FeNiCrCoCu alloys, Journal of Materials Research and Technology 25 (2023) 5575–5585. https://doi.org/10.1016/J.JMRT.2023.07.023.

[37] C. Zeng, A. Neils, J. Lesko, N. Post, Machine learning accelerated discovery of corrosion-resistant high-entropy alloys, Comput

Mater Sci 237 (2024) 112925. https://doi.org/10.1016/J.COMMATSCI.2024.112925.

[38] N. Qu, Y. Zhang, Y. Liu, M. Liao, T. Han, D. Yang, Z. Lai, J. Zhu, L. Yu, Accelerating phase prediction of refractory high entropy alloys via machine learning, Phys Scr 97 (2022). https://doi.org/10.1088/1402-4896/aca2f2.

[39] J. Gao, Y. Wang, J. Hou, J. You, K. Qiu, S. Zhang, J. Wang, Phase Prediction and Visualized Design Process of High Entropy Alloys via Machine Learned Methodology, Metals (Basel) 13 (2023) 283. https://doi.org/10.3390/met13020283.

[40] W. Zhang, P. Li, L. Wang, F. Wan, J. Wu, L. Yong, Explaining of prediction accuracy on phase selection of amorphous alloys and high entropy alloys using support vector machines in machine learning, Mater Today Commun 35 (2023) 105694. https://doi.org/10.1016/J.MTCOMM.2023.105694.

[41] N. Qu, Y. Liu, Y. Zhang, D. Yang, T. Han, M. Liao, Z. Lai, J. Zhu, L. Zhang, Machine learning guided phase formation prediction of high entropy alloys, Mater Today Commun 32 (2022) 104146. https://doi.org/10.1016/J.MTCOMM.2022.104146.

[42] J. Berry, R. Snell, M. Anderson, L.R. Owen, O.M.D.M. Messé, I. Todd, K.A. Christofidou, Design and Selection of High Entropy Alloys for Hardmetal Matrix Applications Using a Coupled Machine Learning and Calculation of Phase Diagrams Methodology, Adv Eng Mater (2024). https://doi.org/10.1002/adem.202302064.

[43] D.Q. Zhao, S.P. Pan, Y. Zhang, P.K. Liaw, J.W. Qiao, Structure prediction in high-entropy alloys with machine learning, Appl Phys Lett 118 (2021). https://doi.org/10.1063/5.0051307.

[44] L. Naveen, P. Umre, P. Chakraborty, M.R. Rahul, S. Samal, R. Tewari, Development of single-phase BCC refractory high entropy alloys using machine learning techniques, Comput Mater Sci 238 (2024). https://doi.org/10.1016/j.commatsci.2024.112917.

[45] W. Huang, P. Martin, H.L. Zhuang, Machine-learning phase prediction of high-entropy alloys, Acta Mater 169 (2019) 225–236. https://doi.org/10.1016/j.actamat.2019.03.012.

[46] S. Swateelagna, M. Singh, M.R. Rahul, Explainable Machine Learning based approach for the design of new refractory high entropy alloys, Intermetallics (Barking) 167 (2024). https://doi.org/10.1016/j.intermet.2024.108198.

[47] H. Ren, R.R. Chen, X.F. Gao, T. Liu, G. Qin, S.P. Wu, J.J. Guo, High-performance AlCoCrFeNi high entropy alloy with marine application perspective, Journal of Materials Research and Technology 25 (2023) 6751–6763. https://doi.org/10.1016/j.jmrt.2023.07.135.

[48] M. Tokarewicz, M. Grądzka-Dahlke, Review of recent research on alcocrfeni high-entropy alloy, Metals (Basel) 11 (2021). https://doi.org/10.3390/met11081302.

[49] C. Chen, Y. Zuo, W. Ye, X. Li, Z. Deng, S.P. Ong, A Critical Review of Machine Learning of Energy Materials, Adv Energy Mater 10 (2020). https://doi.org/10.1002/aenm.201903242.

[50] M. Wang, Y. Cui, X. Wang, S. Xiao, J. Jiang, Machine Learning for Networking: Workflow, Advances and Opportunities, IEEE Netw 32 (2018) 92–99. https://doi.org/10.1109/MNET.2017.1700200.

[51] B. Andriy Burkov, The Hundred-Page Machine Learning, n.d.

[52] Intro to Statistical Learning with Python, (n.d.).

[53] A. Géron, Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow : concepts, tools, and techniques to build intelligent systems, n.d.

[54] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42 (2000) 55–61. https://doi.org/10.1080/00401706.2000.10485979.

[55] S. Guo, C.T. Liu, Phase stability in high entropy alloys: Formation of solid-solution phase or amorphous phase, Progress in Natural Science: Materials International 21 (2011) 433–446. https://doi.org/10.1016/S1002-0071(12)60080-X.

[56] A. Takeuchi, A. Inoue, Classification of Bulk Metallic Glasses by Atomic Size Difference, Heat of Mixing and Period of Constituent Elements and Its Application to Characterization of the Main Alloying Element, n.d.

[57] X. Yang, Y. Zhang, Prediction of high-entropy stabilized solid-solution in multi-component alloys, Mater Chem Phys 132 (2012) 233–238. https://doi.org/10.1016/J.MATCHEMPHYS.2011.11.021.

[58] X. Zhang, L. Liu, K. Yao, K. Duan, F. Wu, R. Zhao, Y. Zhang, J. Shang, M. Chen, The evolution of eutectic microstructure and mechanical properties of AlxCoCrFeNi2.1 high-entropy alloys, J Mater Res 37 (2022) 2082–2092. https://doi.org/10.1557/s43578-022-00620-3.

[59] C. Li, J.C. Li, M. Zhao, Q. Jiang, Effect of alloying elements on microstructure and properties of multiprincipal elements high-entropy alloys, J Alloys Compd 475 (2009) 752–757. https://doi.org/10.1016/j.jallcom.2008.07.124.

[60] J.X. Chen, T. Li, Y. Chen, F.H. Cao, H.Y. Wang, L.H. Dai, Ultra-strong heavy-drawn eutectic high entropy alloy wire, Acta Mater 243 (2023). https://doi.org/10.1016/j.actamat.2022.118515.

[61] L. Ma, J. Wang, P. Jin, Microstructure and mechanical properties variation with Ni content in Al0.8CoCr0.6Fe0.7Nix (x = 1.1, 1.5, 1.8, 2.0) eutectic high-entropy alloy system, Mater Res Express 7 (2020). https://doi.org/10.1088/2053-1591/ab6580.

[62] X. Zhong, Q. Zhang, M. Ma, J. Xie, M. Wu, S. Ren, Y. Yan, Dynamic compressive properties and microstructural evolution of Al1.19Co2CrFeNi1.81 eutectic high entropy alloy at room and cryogenic temperatures, Mater Des 219 (2022). https://doi.org/10.1016/j.matdes.2022.110724.

[63] Y.F. Kao, T.J. Chen, S.K. Chen, J.W. Yeh, Microstructure and mechanical property of as-cast, -homogenized, and -deformed AlxCoCrFeNi ($0 \leq x \leq 2$) high-entropy alloys, J Alloys Compd 488 (2009) 57–64. https://doi.org/10.1016/j.jallcom.2009.08.090.

[64] C.M. Lin, H.L. Tsai, Evolution of microstructure, hardness, and corrosion properties of high-entropy Al0.5CoCrFeNi alloy, Intermetallics (Barking) 19 (2011) 288–294. https://doi.org/10.1016/j.intermet.2010.10.008.

[65] Ł. Rogal, Z. Szklarz, P. Bobrowski, D. Kalita, G. Garzeł, A. Tarasek, M. Kot, M. Szlezynger, Microstructure and Mechanical Properties of Al–Co–Cr–Fe–Ni Base High Entropy Alloys Obtained Using Powder Metallurgy, Metals and Materials International 25 (2019) 930–945. https://doi.org/10.1007/s12540-018-00236-5.

[66] R.J. Vikram, K. Gupta, S. Suwas, Design of a new cobalt base nano-lamellar eutectic high entropy alloy, Scr Mater 202 (2021). https://doi.org/10.1016/j.scriptamat.2021.113993.

[67] A. Shafiei, S. Rajabi, A cobalt-rich eutectic high-entropy alloy in the system Al–Co–Cr–Fe–Ni, Appl Phys A Mater Sci Process 125 (2019). https://doi.org/10.1007/s00339-019-3084-9.

[68] Q. Cheng, T.H. Fang, P. Xie, Y. Zhao, J.H. You, X.D. Xu, Impact of surface gradient structures on mechanical properties of a dual-phase AlCrFe2(Ni0.85Co0.15)2 multi-component eutectic alloy, J Alloys Compd 882 (2021). https://doi.org/10.1016/j.jallcom.2021.160591.

[69] H. Jiang, L. Li, R. Wang, K. Han, Q. Wang, Effects of Chromium on the Microstructures and Mechanical Properties of AlCoCrxFeNi2.1 Eutectic High Entropy Alloys, Acta Metallurgica Sinica (English Letters) 34 (2021) 1565–1573. https://doi.org/10.1007/s40195-021-01303-4.

[70] J.Q. Zheng, M.L. Wang, W.N. Jiao, L.J. Zou, Y. Di, Effect of Ti Addition on Microstructure Evolution and Mechanical Properties of Al18Co13Cr10Fe14Ni45 Eutectic High-Entropy Alloys, Acta Metallurgica Sinica (English Letters) 36 (2023) 1493–1501. https://doi.org/10.1007/s40195-023-01564-1.

[71] A. Shafiei, Design of Eutectic High Entropy Alloys, Metall Mater Trans A Phys Metall Mater Sci 53 (2022) 4349–4361. https://doi.org/10.1007/s11661-022-06831-x.

[72] Y. Lu, X. Gao, L. Jiang, Z. Chen, T. Wang, J. Jie, H. Kang, Y. Zhang, S. Guo, H. Ruan, Y. Zhao, Z. Cao, T. Li, Directly cast bulk eutectic and near-eutectic high entropy alloys with balanced strength and ductility in a wide temperature range, Acta Mater 124 (2017) 143–150. https://doi.org/10.1016/j.actamat.2016.11.016.

[73] Z. Fu, W. Chen, H. Wen, Z. Chen, E.J. Lavernia, Effects of Co and sintering method on microstructure and mechanical behavior of a high-entropy Al0.6NiFeCrCo alloy prepared by powder metallurgy, J Alloys Compd 646 (2015) 175–182. https://doi.org/10.1016/j.jallcom.2015.04.238.

[74] W. Jiao, T. Li, G. Yin, T. He, T. Li, Y. Lu, Hot deformation characteristics and microstructure evolution of Al20Co36Cr4Fe4Ni36 eutectic high entropy alloy, Mater Charact 204 (2023). https://doi.org/10.1016/j.matchar.2023.113180.

[75] D.H. Chung, J. Lee, Q.F. He, Y.K. Kim, K.R. Lim, H.S. Kim, Y. Yang, Y.S. Na, Hetero-deformation promoted strengthening and toughening in BCC rich eutectic and near eutectic high entropy alloys, J Mater Sci Technol 146 (2023) 1–9. https://doi.org/10.1016/j.jmst.2022.10.036.

[76] Y. Dong, J. Yuan, Z. Zhong, S. Liu, J. Zhang, C. Li, Z. Zhang, Accelerated design eutectic-high-entropy-alloys using simple empirical rules, Mater Lett 309 (2022). https://doi.org/10.1016/j.matlet.2021.131340.

[77] Y. Dong, Z. Yao, X. Huang, F. Du, C. Li, A. Chen, F. Wu, Y. Cheng, Z. Zhang, Microstructure and mechanical properties of AlCoxCrFeNi3-x eutectic high-entropy-alloy system, J Alloys Compd 823 (2020). https://doi.org/10.1016/j.jallcom.2020.153886.

[78] X. Jin, Y. Zhou, L. Zhang, X. Du, B. Li, A new pseudo binary strategy to design eutectic high entropy alloys using mixing enthalpy and valence electron concentration, Mater Des 143 (2018) 49–55. https://doi.org/10.1016/j.matdes.2018.01.057.

[79] Z. Yang, Z. Wang, Q. Wu, T. Zheng, P. Zhao, J. Zhao, J. Chen, Enhancing the mechanical properties of casting eutectic high entropy alloys with Mo addition, Appl Phys A Mater Sci Process 125 (2019). https://doi.org/10.1007/s00339-019-2506-z.

[80] Y. Fu, C. Dai, H. Luo, D. Li, C. Du, X. Li, The corrosion behavior and film properties of Al-containing high-entropy alloys in acidic solutions, Appl Surf Sci 560 (2021). https://doi.org/10.1016/j.apsusc.2021.149854.

[81] Y.F. Kao, T.D. Lee, S.K. Chen, Y.S. Chang, Electrochemical passive properties of AlxCoCrFeNi (x = 0, 0.25, 0.50, 1.00) alloys in sulfuric acids, Corros Sci 52 (2010) 1026–1034. https://doi.org/10.1016/j.corsci.2009.11.028.