PriMera Scientific Engineering
Volume 4 Issue 4 April 2024
DOI: 10.56831/PSEN-04-118
ISSN: 2834-2550

PriMera Scientific Publications

Inverse Problem Analysis of Phase Fraction Prediction in Aluminum Alloys Using Differentiable Deep Learning Models

Type: Review Article
Received: February 26, 2024
Published: March 20, 2024

Citation:
Yu Okano., et al. "Inverse Problem Analysis of Phase Fraction Prediction in Aluminum Alloys Using Differentiable Deep Learning Models". PriMera Scientific Engineering 4.4 (2024): 03-09.

Yu Okano*, Takeshi Kaneshita*, Shimpei Takemoto and Yoshishige Okuno

Resonac Corporation, 8, Ebisu-Cho, Kanagawa-Ku, Yokohama, 2210024, Japan

***Corresponding Author:** Yu Okano, Takeshi Kaneshita, Resonac Corporation, 8, Ebisu-Cho, Kanagawa-Ku, Yokohama, 2210024, Japan.

Abstract

In recent years, there has been an increasing demand for the optimization of alloy properties, driven by the growing complexity of end products and the need to reduce development costs. In general, Thermo-Calc based on the CALPHAD method, which calculates the thermodynamic state of an alloy, is widely used for efficient alloy development. However, a challenge in alloy exploration using Thermo-Calc is the need for specialized computational skills and the significant computational effort required due to the extensive range of calculation conditions for numerous alloys. Consequently, we have developed a deep learning model that rapidly and accurately predicts the temperature-dependent changes in equilibrium phase fractions for 6000 series aluminum alloys (Al-Mg-Si based alloys), which are widely used in industry, using calculations from Thermo-Calc. We developed the architecture of the deep learning model based on the Transformer, which is commonly used in natural language processing tasks. The model is capable of performing calculations more than 100 times faster than ThermoCalc. Furthermore, by leveraging backpropagation of errors in the trained model, we developed a method to estimate the alloy composition for the phase fraction results calculated based on Thermo-Calc.

Keywords: Deep Learning; Inverse Problem; CALPHAD; Transformer

Introduction

As products become increasingly complex and there is a growing demand for cost reduction in development, the need for optimizing the properties of alloys that make up these products is also escalating. For instance, aluminum alloys are garnering attention as essential metal materials for lightweighting final products. It is widely recognized that the various phases that form within these aluminum alloys have a direct correlation with their mechanical properties. Therefore, by utilizing phase diagrams to calculate phase fractions and to identify the compositional and thermal ranges where these phases are stable, it becomes possible to optimally control the mechanical properties of the alloys. Currently, in the realm of alloy design through computational methods, ThermoCalc [1], based on the CALPHAD (Calculation of Phase Diagrams, Computer Coupling of Phase Diagrams

and Thermochemistry) approach, is widely utilized. In aluminum alloys, a forward problem approach is commonly adopted, where the alloy composition and manufacturing processes are controlled to optimize the metal's microstructure, using CALPHAD methods for property optimization. On the other hand, from the perspective of efficient materials exploration, adopting an inverse problem approach can rapidly identify the alloy composition best suited for targeted properties, such as strength, heat resistance, and corrosion resistance. This has the potential to significantly accelerate the material development cycle.

Therefore, we have developed a deep learning model based on the Transformer [2] architecture that can calculate phase fractions in the industrially prevalent 6000 series aluminum alloys at a much faster rate than the CALPHAD method. Additionally, utilizing the deep learning model we developed, we have also created an algorithm for inverse problem-solving that can estimate the composition of additive elements in an alloy when phase fractions calculated by the CALPHAD method are provided. This approach aims to significantly increase the efficiency of material exploration.

Formulation of Inverse Problems in Deep Learning

Let θ denote the parameters of the deep learning model, the process by which the model outputs y from an input x can be formulated as shown in Equation (1). In this context, training the model involves minimizing the loss function $J(\theta)$, which corresponds to the average error between the predicted values y_{pred} and the true values y_{true} across the entire dataset, as defined in Equation (2). The optimal model parameters θ^* are identified by minimizing $J(\theta)$, as described in Equation (3). The optimal parameters θ^* can be determined through iterative updates using gradient descent, as outlined in Equation (4). The gradient of the loss function can be efficiently computed using the backpropagation algorithm.

$$y = f_\theta(x) \tag{1}$$

$$J(\theta) = \frac{1}{N}\sum_{i=1}^{N} L(y^i_{pred}, y^i_{true}) \tag{2}$$

$$\theta^* = argmin_\theta \frac{1}{N}\sum_{i=1}^{N} L(f_\theta(x^i), y^i_{true}) \tag{3}$$

$$\theta = \theta - \alpha\nabla_\theta J(\theta) \tag{4}$$

In this paper, the inverse problem, given a trained model, is defined as follows: when a particular output y^* is provided, the objective is to find a specific input x^* that results in the model outputting y^*. This is formulated as Equation (6), where the goal is to find x^* that minimizes the loss function between the given y^* and $f_\theta(x)$, as defined in Equation (5). The value of x^* can be determined through iterative updates using gradient descent and backpropagation, in a manner similar to that described in Equation (4), as outlined in Equation (7). During the model training phase, the gradient of θ is calculated with respect to the loss function for the entire dataset, as described in Equation (3) and outlined in Equation (4). However, in solving the specific inverse problem of finding x^*, the gradient is calculated with respect to a loss function that depends on x as defined in Equation (5). This represents a key difference between the two approaches. In the initial stages of iterative updates for x, the value of $J(x)$ decreases as x is updated. However, as the updates continue, $J(x)$ eventually converges to a specific value. If the converged value of $J(x)$ is sufficiently small, then the model's output $f_\theta(x^*)$ and y^* can be considered to be essentially identical, implying that the desired x^* has been successfully determined.

$$J(x) = L\big(y^*, f_\theta(x)\big) \tag{5}$$

$$x^* = argmin_x L\big(y^*, f_\theta(x)\big) \tag{6}$$

$$x = x - \alpha\nabla_x J(x) \tag{7}$$

Training Data

To calculate the phase fractions of 6000 series aluminum alloys based on the CALPHAD method, Thermo-Calc was used. When calculating phase fractions, we defined 11 phases consisting of *Liquid*(*Al*), *Al*, *Al9Fe2Si2*, *Al8Fe2Si2*, *Al15*(*Mn, Fe*)*3Si2*, *Mg2*(*Si, Sn*), (*Al, Si*)*3Ti*(*Lt*), (*Al, Si*)*11Cr4*, *DO23-Al3*(*Ti, Zr*), *Liquid*(*Solder*), and *Pb*. Based on these phases, we generated a total of 841,689 training data points. In the Japanese Industrial Standards (JIS), which set the typical alloy standards in Japan, specific series are designated for the 6000 series of aluminum alloys. These include the 6010, 6013, 6060, 6061, 6063, 6066, 6070, 6101, 6181, and 6351 series. For each of these series, the upper and lower limits for the values of the alloying elements are specified, as shown in Table 1. The alloying elements include 12 types: *Si, Fe, Cu, Mn, Mg, Cr, Zn, Ti, Zr, B, Bi*, and *Pb*. To generate the training data, we first randomly selected alloys that meet the standard specifications. For each selected alloy, we randomly determined the composition of each alloying element within the specified upper and lower limits. The composition of *Al* was set so that the sum of all alloying elements equaled 1. Using this set of 13 values to define the alloy composition, we then calculated the phase fractions using Thermo-Calc. Using this established set of 13 values to define the alloy composition, we calculated the phase fractions using Thermo-Calc. For 70 temperature points ranging from 100°C to 790°C in 10°C increments, we used the phase fraction values from the 11 pre-defined phases as the training data. The reason for obtaining values at 10°C intervals is to optimize memory usage and reduce computation time during inference with the trained model. Note that when visualizing the phase fractions as shown in Fig. 1, we display values at 1°C intervals, which have been interpolated linearly.

Elements	Si	Fe	Cu	Mn	Mg	Cr	Zn	Ti	Zr	B	Bi	Pb	Al
Min	0.60	0.00	0.60	0.20	0.80	0.00	0.00	0.00	0.00	0.00	0.00	0.00	Bal.
Max	1.00	0.50	1.10	0.80	1.20	0.15	0.25	0.10	0.05	0.05	0.05	0.05	Bal.

Table 1: Upper and Lower Composition Limits of JIS Standard 6013 Series Aluminum Alloy.

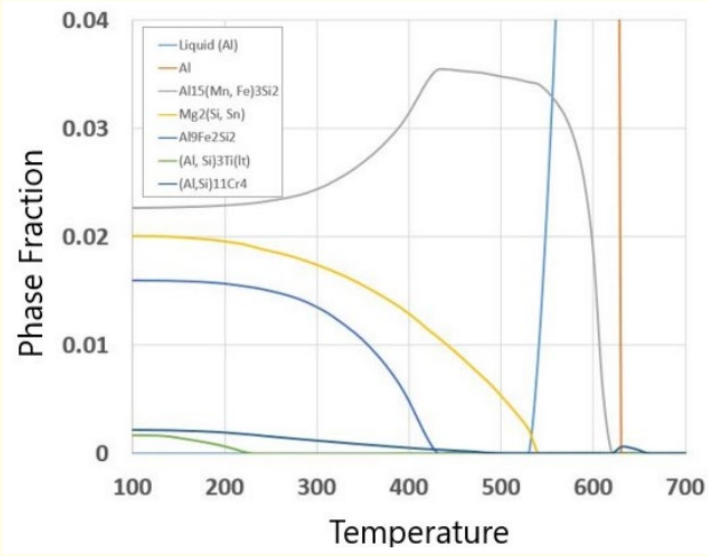

Figure 1: Example of Phase Fraction Calculation.

Model Architecture

The Transformer is a deep learning model that has been employed for various tasks, including natural language processing and image recognition [3], as illustrated in Figure 2. We utilized the architecture of the Transformer to predict phase fractions from the alloy compositions of aluminum alloys. Modifications specific to phase fraction prediction were made solely to the Input Embeddings of both the Encoder and Decoder.

As shown in Table 1, the alloy composition consists of a set of 13 values. We directly fed this 13-value set into the Input Embedding of the Encoder. In the Encoder's Input Embedding, a fully connected layer combined with an activation function was used to transform each of the 13 alloy composition values into a 1536-dimensional embedding vector. The parameters of this fully connected layer are subject to optimization during training.

The Decoder is designed to output phase fractions; thus, it generates 11-dimensional vectors for each of 70 temperature points, ranging from 100°C to 790°C in 10°C increments. Each value in the 11-dimensional vector represents the phase fraction of one of the 11 predefined phases used in the phase fraction calculations, and the sum of phase fractions at each temperature point is constrained to be 1. In the Transformer architecture, it is necessary to maintain a consistent dimensionality for the embedding vectors between the Encoder and Decoder to perform Attention calculations. Therefore, in the Decoder's Input Embedding, we employed a combination of a fully connected layer and an activation function to transform the 11-dimensional vector into a 1536-dimensional embedding vector. The parameters of this fully connected layer are subject to learning, similar to those in the Encoder.

The Transformer architecture involves multiple hyperparameters. In the model used for this study, the hyperparameters are as follows:

- Number of Heads in Multi-Head Attention: 24.
- Number of Encoder Stacks: 2.
- Number of Decoder Stacks: 3.

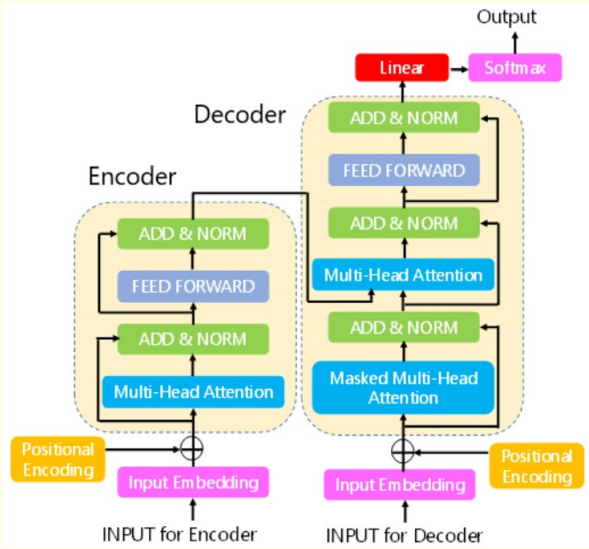

Figure 2: Transformer Architecture.

Model Training

In the context of phase fraction prediction, the range of phase fractions that are practically relevant extends from approximately 10^{-3} to 1. Consequently, to learn phase fractions accurately, it is essential to efficiently propagate the error gradient over this broad range. Additionally, the graph of the phase fractions exhibits complex behavior around 600°C, the approximate melting point of aluminum alloys, due to phase transitions occurring in this temperature range. Therefore, when calculating the loss function, we divided the temperature range of interest into three segments: (a) the low-temperature region from 100°C to 300°C, (b) the high-temperature region from 490°C to 700°C, and (c) the full temperature range from 100°C to 790°C. For the loss function related to the phase fraction error, we employed three different types: (1) Mean Squared Logarithmic Error, (2) Cross-Entropy Loss, and (3) Mean Absolute Error. Additionally, to ensure the continuity of the predicted phase fractions with respect to temperature, we also included the Mean Absolute Error of the temperature-dependent differences in phase fractions as a fourth component of the loss function. Training was conducted by applying these four loss functions to each of the three temperature segments (a), (b), and (c).

Model Prediction Result

Table 2 presents the loss values for 17,980 validation data points for the deep learning models. For comparison, the table includes prediction results from three types of deep learning models: Transformer, Seq2Seq [4], and a network composed solely of fully connected layers. In Figure 3, we present the calculated results from Thermo-Calc alongside the predictive outcomes from deep learning models. We found that the Transformer model exhibited the highest prediction accuracy, providing forecasts nearly equivalent to the results obtained from Thermo-Calc. Based on the results from Table 3, we confirmed that when using a GPU, the Transformer model can calculate phase fractions more than 100 times faster than Thermo-Calc.

Model	Mean Squared Logarithmic Error
Dense Network	3.79×10^{-4}
Seq2Seq	5.1×10^{-5}
Transformer	2.45×10^{-5}

Table 2: Comparison of Loss on Validation Data.

Model	Calculation Time using GPU	Calculation Time using CPU
CALPHAD	51 seconds	51 seconds
Transformer	0.49 seconds	4.0 seconds

Table 3: Comparison of Calculation Time.

Inverse Problem Result

We validated whether it is possible to estimate x^*, corresponding to the alloy composition under computational conditions, from y^*, which corresponds to phase fractions calculated by Thermo-Calc, based on Equation (7). However, as per Equation (7), updating the alloy composition values could result in negative figures depending on the learning rate set. Therefore, a constraint was added to ensure that each component of the alloy composition is greater than or equal to 0 during the update. Additionally, constraints were imposed to ensure that the upper limit for each alloy composition also falls within the range specified for 6000-series alloys by the JIS standards. If these constraints were not applied during the value update step, the model's input values would deviate from the range of the training data, resulting in decreased predictive accuracy and preventing the loss function from converging.

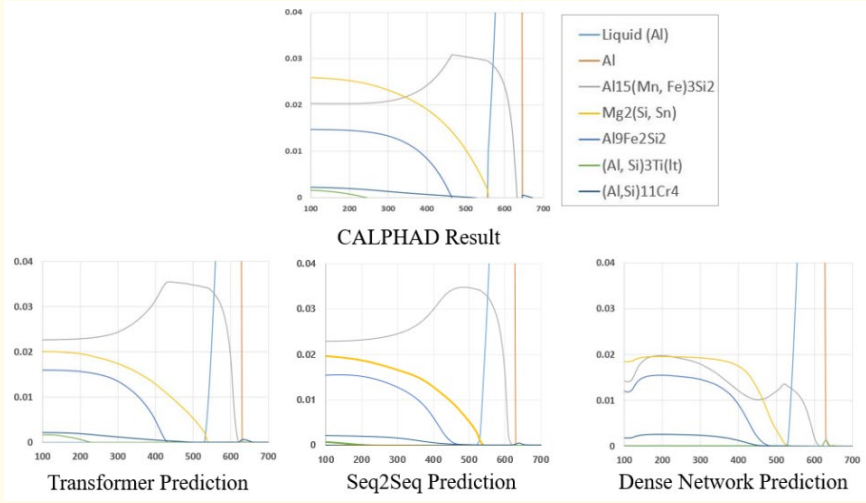

Figure 3: Comparisons Between CALPHAD Method and Model Predictions.

Furthermore, due to the constraint that the sum of all alloy compositions must equal 1, the composition of *Al* was not updated using Equation (7). Instead, after updating the values of the other alloying elements, it was set such that the sum of all alloying elements' compositions equals 1.

Multiple initial values of x were selected within the compositional range of 6000-series alloys according to JIS standards for the application of inverse problem analysis. It is desirable for the phase fractions corresponding to these initial values to resemble the shape of y^*. Among these initial values, the one yielding the lowest final loss function value was adopted.

As illustrated in Fig. 4, we confirmed that an alloy composition capable of replicating the given phase fractions could be determined. Although the initial phase fractions differ significantly in shape from the target phase fractions, it was confirmed that they eventually converge to a shape largely identical to the target phase fractions.

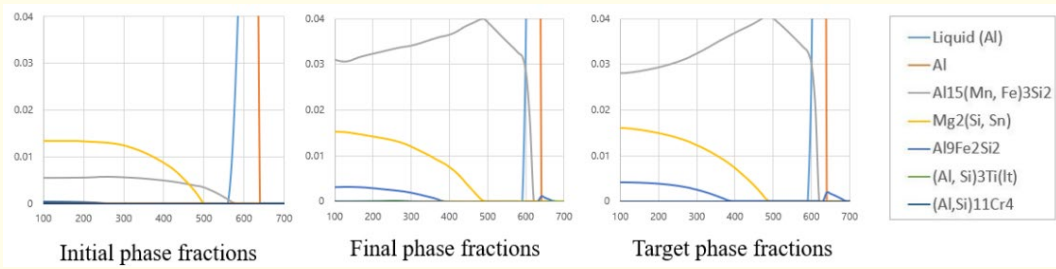

Figure 4: Comparison of Phase Fraction Changes Due to Alloy Composition Updates.

The initial and final alloy compositions corresponding to Fig. 4, along with their comparison to the Ground Truth (GT), are listed in Table 4. It is observed that the estimated values for the key elements are generally close to those of the GT.

Elements	Si	Fe	Cu	Mn	Mg	Cr	Zn	Ti	Zr	B	Bi	Pb	Al
Initial alloy composition	1.0	0.5	1.1	0.8	1.2	0.1	0.25	0.1	0.00	0.00	0.00	0.00	Bal.
Final alloy composition	0.21	0.62	0.61	1.10	1.00	0.12	0.06	0.01	0.00	0.00	0.00	0.00	Bal.
Ground Truth	0.20	0.63	0.66	1.17	0.81	0.26	0.36	0.18	0.00	0.00	0.00	0.00	Bal.

Table 4: Final Alloy Composition After Update.

Conclusion

In this study, we explored methods for predicting phase fractions in aluminum alloys and solving inverse problems using deep learning models. By employing the Transformer architecture for prediction, we achieved comparable accuracy to the CALPHAD method while being approximately 100 times faster. Furthermore, by utilizing the model trained through inverse problem analysis, we were able to estimate the alloy composition conditions for the phase fraction results calculated based on the CALPHAD method.

References

1. B Sundman, B Jansson and J-O Andersson. "The thermo-calc databank system". Calphad 9.2 (1985): 153-190.
2. A Vaswani., et al. "Attention is all you need". in Advances in Neural Information Processing Systems (2017): 5998-6008. Author, F., Author, S., Author, T.: Book title. 2nd edn. Publisher, Location (1999).
3. Alexey Dosovitskiy., et al. "An image is worth 16x16 words: Transformers for image recognition at scale". arXiv preprint arXiv:2010.11929 (2020).
4. Ilya Sutskever, Oriol Vinyals and Quoc V Le. "Sequence to sequence learning with neural networks". Advances in neural information processing systems 27 (2014).