Geopolymer analysis using PINN's

[1,*] [1,†]
[1]

Abstract

1 Introduction

The term geopolymer was coined by Joseph Davidovits in 1978. He suggested that the blocks were formed by pouring crushed limestone, clay, lime and water into molds. This mixture hardened into a fake, synthetic stone that he called, geopolymer. Using the same theory, scientists have created a new type of concrete that is 100 per cent free of cement! They use a variety of waste materials, industrial byproducts, and minimally processed natural materials instead of Portland Cement. These waste products are difficult to dispose of and they can affect the environment if left untreated. So, using them to significantly reduce the carbon footprint of concrete. They are :

Fly ash or pulverized fuel ash or PFA which is a fine, powdery residue from coal-burning power plants. It is mainly composed of silicon dioxide, aluminiumoxide and calcium oxide. Metakaolin is produced when china clay or kaolin is heated to 1500 degrees Fahrenheit or 800 degrees Celsius. It can be used to make tiles and traditional concrete. Ground granulated blast furnace slag or GGBS is a by-product of the steel industry. It is high in calcium silicate hydrates CSH which improves the strength, durability and appearance of concrete. Palm oil fuel ash or POFA is produced by burning palm oil shells and husks. It is a hazardous material that is usually sent to landfills, but it can be used to make cement. Every one of these raw materials performs differently and produces a different type of geopolymer cement. The simple overall reaction is that most of the waste productscontain silicon dioxide. They are mixed with an alkaline activator like sodium hydroxide or potassium hydroxide and sodium silicate or potassium silicate. The silicon oxygen silicon bonds break down and aluminium atoms penetrate them to form aluminosilicate gels. With more alkali, these gels harden into a geopolymer cement. This cement is mixed with aggregate and water to form geopolymer concrete. ADVANTAGES OF GEOPOLYMERS: Geopolymers are apparently stronger in tension and compression. They are resistant to various alkalis, salts, acids and corrosive substances. They also have a high sulphur resistance due to the lack of calcium compounds in their structure. Geopolymers have excellent waterproof properties. Its pores are smaller than 50 nanometers so big molecules like water can't enter the geopolymer matrix

It is also fire resistant. Unlike Portland cement, water in geopolymer concrete evaporates and does not explode the concrete from inside. Foamed geopolymer concrete can also have superior thermal insulation thanks to the trapped air inside the blocks. They can reduce the carbon footprint of concrete by up to 90 per cent. For every 1 ton of Portland cement made, around 0.9 tons of CO2 is released. Conversely, for every 1 ton of geopolymer cement made, only 0.2 tons of CO2is released. Geopolymers will help us stop mining and extracting raw materials from the earth because we can use waste and by-products of existing industries. It can speed up construction because it develops about 50 per cent of its strength in the first 3 days.

DISADVANTAGES OF GEOPOLYMERS: There isn't enough waste material to meet the demand for cement. The global consumption of cement is about 4 to 5 times the generation of industrial wastes like fly ash and slag. Another issue is the lack of standard and uniform waste materials. For example, fly ash will have a different alumina-silica ratio than slag, so their geopolymer cement mixes will perform differently. They are difficult to make because the geo-polymerization process is sensitive. Chemicals like sodium hydroxide are harmful to humans so the process requires special handling. And finally, the biggest drawback in my opinion is that they are up against powerful Portland cement manufacturers, some of whom have a monopoly in the concrete industry. They have control over code requirements and marketing which allows them to easily suppress competition.

The history of neural networks undoubtedly began in the late 19th century with scientific efforts to study human brain activity. In 1890, William James published his first study of brain activity patterns. In 1943, McCulloch and Pitts created a model of neurons that are still used in artificial neural networks. This model is divided into two parts:

*
†

In 1949, Donald Hebb published The Organization of Behavior, which shows the principles governing the learning of synaptic neurons. This law later referred to as Hebb learning in honour of Donald Hebb, is one of the simplest and simplest learning rules for artificial neural networks. In 1951, Narvin Minsky created the primary artificial neural network (ANN) while working at Princeton. "The Computer and the Brain" was published in 1958, a year after the death of John John von Neumann. in this book, von Neumann proposed many extreme changes within the way analysts model the brain.

Perceptron:The Perceptron was developed in 1958 by Frank Rosenblatt of Cornell University. Perceptron was an attempt to use neural network technology for character recognition. Perceptrons are linear systems that helped solve the problem that input classes are linearly separable in input space. In 1960, Rosenblatt introduced the Principles of Neurodynamics. This includes a number of his research and ideas on brain modelling. Research perceptrons and artificial neural networks was an early achievement, but many felt that the guarantees of those methods were limited. Among these were Marvin Minsky and Seymour Papert, who used the 1969 book Perceptron to disgrace ANNâs work and draw attention to the apparent limitations of ANNâs work. one of the limitations emphasized by Minsky and Papert was that the perceptron couldn't identify patterns that might not be linearly separated within the input space using the linear classification problem. Despite Perceptronâs disappointment with processing non-linearly separable data, this wasn't a technology-specific failure, but a matter of scale. Hecht-Nielsen demonstrated the two-layer perceptron (Mark) in 1990. this is a three-layer machine equipped to handle the problem of non-linear separation. Perceptron entered a period called the "quiet year" when ANNâs research had a minimal interest. The backpropagation algorithm, first developed by Werbos in 1974, was rediscovered in 1986 in the book "Learning Internal Representations by Error Propagation" by Rumelhart, Hinton, and Williams. Backpropagation is a form of gradient descent algorithm employed in artificial neural networks for reduction and curve fitting. In 1987, the Annual IEEE International ANN Conference for ANN Scientists was established. In 1987, the International Neural Network Society (INNS) was founded with the INNS Neural Networking Journal 1988.

Neural Network:

The power of neural computations comes from connecting neurons in a network. Each PE has weighted inputs, a transfer function and one output. The behaviour of a neural network is set by the transfer functions of its neurons, by the learning rule, and by the architecture itself. The weights are the adjustable parameters and, in that sense, a neural network is a parameterized system. The weighted sum of the inputs constitutes the activation of the neuron. The activation signal is passed through the transfer function to produce one output of the neuron. The transfer function introduces non-linearity to the network. During training, the inter-unit connections are optimized until the error in predictions is minimized and also the network reaches the required level of accuracy. Once the network is trained and tested it can be given new input information to predict the output. many varieties of neural networks are designed already and new ones are invented every week but all can be described by the transfer functions of their neurons, by the training rule, and by the connection formula.

ANN Layers

There are different layers for creating a neural network, certain connections and properties built with those neurons form a model that can predict and classify.

a. Input layer

The purpose of the input layer is to receive as input the values of the explanatory attributes for every observation. Usually, the number of input nodes in an input layer is equal to the number of explanatory variables. The input layer presents the patterns to the network, which communicates to at least one or more hidden layers.The nodes of the input layer are passive, meaning they are doing not change the data. They receive one value on their input and duplicate the worth of their many outputs. From the input layer, it duplicates each value and is sent to all the hidden nodes.

b. Hidden Layer

The Hidden layers apply given transformations to the input values inside the network. In this, incoming arcs that go from other hidden nodes or from input nodes connected to every node. It connects with outgoing arcs to output nodes or to other hidden nodes. in a hidden layer, the actual processing is finished via a system of weighted connections. There are also one or more hidden layers. The values entering a hidden node are multiplied by weights, a collection of predetermined numbers stored within the program. The weighted inputs are then added to produce one number.

c. Output layer

The hidden layers then link to an output layer. The output layer receives connections from hidden layers or from the input layer. It returns an output value that corresponds to the prediction of the response variable. In classification problems, there is usually only 1 output node. The active nodes of the output layer combine and alter the data to provide the output values. The ability of the neural network to provide useful data manipulation lies within the proper selection of the weights. This is different from conventional information processing.

2 Different neural networks and PINN

2.1 ANNs and their propertiesl

SINGLE LAYER FEED FORWARD NETWORK: in this kind of network, weâve got only two layers, i.e. input layer and the output layer but the input layer doesnât count because no computation is performed during this layer. Output Layer is made when different weights are applied to input nodes and also the cumulative effect per node is taken. After this, the neurons collectively give the output layer to compute the output signals. MULTI-LAYER FEED FORWARD NETWORK: This network encompasses a hidden layer internal to the network and has no direct contact with the external layer. The existence of 1 or more hidden layers enables the network to be computationally more robust. There are not any feedback connections within which the model outputs are fed back to itself. SINGLE NODE WITH IT OWN FEEDBACK: When outputs are directed back as inputs to the identical layer or proceeding layer nodes, then it finally ends up in feedback networks. Recurrent networks are feedback networks with closed loops. SINGLE LAYER RECURRENT NETWORK: This network is a single-layer network with a feedback connection during which the processing elementâs output will be directed back to itself or to other processing elements or both. A recurrent neural network is a class of artificial neural networks where the connection between nodes forms a directed graph along a sequence. this allows it to exhibit dynamic temporal behaviour for a time sequence. Unlike feedforward neural networks, RNNs can use their internal state (memory) to process sequences of inputs. MULTILAYER LAYER RECURRENT NETWORK: in this kind of network, processing element output is directed to the processing element within the identical layer and in the preceding layer forming a multilayer recurrent network. They perform the identical task for every element of the sequence, with the output being dependent on the previous computations. Inputs arenât needed at each time step. the main feature of a multilayer recurrent network is its hidden state, which captures information about a sequence.

2.2 PINN vs ANN

Physics-informed neural networks (PINN's) are a type of universal function approximators that can embed the knowledge of any physical laws that govern a given data-set in the learning process, and can be described by partial differential equations (PDEs). They overcome the low data availability of some biological and engineering systems that makes most state-of-the-art machine learning techniques lack robustness, rendering them ineffective in these scenarios. The prior knowledge of general physical laws acts in the training of neural networks (NNs) as a regularization agent that limits the space of admissible solutions, increasing the correctness of the function approximation.

Physics-informed neural networks (PINNs) have gained popularity across different engineering fields due to their effectiveness in solving realistic problems with noisy data and often partially missing physics. In PINNs, automatic differentiation is leveraged to evaluate differential operators without discretization errors, and a multitask learning problem is defined in order to simultaneously fit observed data while respecting the underlying governing laws of physics. applications of PINNs have been presented in various prototype heat transfer problems, targeting in particular realistic conditions not readily tackled with traditional computational methods. For example in, the authors considered forced and mixed convection with unknown thermal boundary conditions on the heated surfaces and aim to obtain the temperature and velocity fields everywhere in the domain, including the boundaries, given some sparse temperature measurements. Stefan problem for two-phase flow, aiming to infer the moving interface, the velocity and temperature fields as well as the different conductivities of a solid and a liquid phase, given a few temperature measurements inside the domain. There are various realistic industrial applications related to power electronics to highlight the practicality of PINNs and the effective use of neural networks in solving general heat transfer problems of industrial complexity. Taken together, the results demonstrate that PINNs not only can solve ill-posed problems, which are beyond the reach of traditional computational methods, but they can also bridge the gap between computational and experimental heat transfer.

Most of the physical laws that govern the dynamics of a system can be described by partial differential equations. For example, the Navier Stokes equations are a set of partial differential equations derived from the conservation laws (i.e., conservation of mass, momentum, and energy) that govern fluid mechanics. The solution of the Navier Stokes equations with appropriate initial and boundary conditions allows the quantification of flow dynamics in a precisely defined geometry. However, these equations cannot be solved exactly and therefore numerical methods must be used (such as finite differences, finite elements and finite volumes). In this setting, these governing equations must be solved while accounting for prior assumptions, linearization, and adequate time and space discretization.

To understand physics-informed neural networks the basics of neural networks are crucial. The goal of the neural network is to approximate a function f between input vector x to output y. Mapping between the input and output is given by $y = f(x, \theta$ where theta consists of weights(w) and bias(b). If one has multiple inputs x_1 x_2 and x_n they are multiplied by corresponding weights to a neuron and this multiplication is affected through matrix operation in which you have multiplication with weights and addition of a bias term which goes to an input of a neuron and the output is generated through an activation function which

could be a sigmoid or many others that exist. Theta is essential for the mapping of input and output as it contains the weights and bias and theta is learnt by iterating over the data. In a multi-layer perceptron, input data is processed over multiple layers of neurons each layer i of neuron is associated with implementing a certain function f(i) and finally the data actually follows from input to output in feed-forward nature in one direction only and is essentially generated through a chain of multiple functions $y = f3(f2(f1(x)$, the idea is that given a sufficient depth of layers and number of neurons we can approximate y.Back propagation refers to the method of computing the gradient it is essential to train the neural network to learn from the data the gradient function of a scalar is a function f(X) and the gradient for a function f(x) is the differentiation of that function so basically the gradients are the differentiation of the functions and gradient a vector function y as a function of x but where y and x are vectors is given by jacobian matrix such that j of i j the individual element of those matrices is given by the equation $\delta(f(x)i)/\delta(xj)$.The idea of availability of gradients y1 and y2 is very essential this is the first key ingredient that allows us to impose physics on neural networks. The next important aspect is the Loss function is measure of deviation or inequity or cost you want to minimize the choice of loss function depends on nature of problem for example msc for regression or cross-entropy for binary or multi-class classification problems and the learning algorithm here refers to gradient descent technique used to adjust the weights to minimize L of theta since L of theta is a measure of deviation we want to ensure that the cost or L of theta is minimized and the direction of training of weights is in the direction of decreasing L of theta and is given by theta dash ,the loss function is given by new theta $\theta' = \theta - \eta\nabla$ L(θ) where η is the learning rate ∇ is the gradient descent and L(θ) is the learning function.Second ingredient FOR PINN is a loss function generation or construction of a custom model of theta the third ingredient necessary for pins which we leverage is the neural network learning algorithm itself. PINN are used when we collect some data we fit a model and we try to explain the world around uswe don't have a lot of data but we have very good understanding of the correlation and the functional dependencies between different parameters, where we know this the governing equations and a noise data without all the boundary conditions physics simply put is essentially study of matter energy and its properties and the mathematical framework on which physics rests is essentially the differential equations which describes the interrelationships of different material properties, for example, one-dimensional heat equation. It is this mathematical framework of differential equation which is used to describe the physics around us which is actually used to impose the physics on the neural network. There are different methods to solve differential equations like analytical solutions and for non-linear and more complicated differential equations we break these types of differential equations into smaller linear differential equations and solve that using the finite difference method. We can solve these by using neural networks, first key element needed for this is a way to support any functional form of the solution $y = f(x)$ this is actually supported through neural networks through the idea of universal approximation. Secondly, we should have availability of derivatives so that we can construct a differential equation and arrive at a solution easily as derivatives are readily available in the neural network through the back prop and the jacobian and chain rules. Thirdly a neural network can work as an optimizer to actually provide y as a solution. When x input goes through the feed-forward neural network structure and generates y and assume y is equal to the neural network of x and is the full functional form of solution to the differential equation. Now we train the weights to arrive at y this is done by creating a custom loss function in a typical loss function we have L(θ) which is essentially a difference between y'(function running in the NN) and y, and that loss is minimised to generate new weights and train the neural network but to solve the differential equation (when $y = y'$) is to create loss function to include residual(loss) of the differential equation, the initial condition and the boundary condition and these three additional constraints added in loss function helps solve the differential equation.

2.3 Advantage of PINN over other methods for solving PDE's

As a generic methodology or "holy grail," machine learning (for pattern recognition) is unlikely to replace PDE solvers because established numerical methods are madeto be reliable, effective, and accurate, and they are well supported by theoretical frameworks. Machine learning offers the possibility of finding hidden and practical patterns in highly complicated data sets, etc., and is primarily used for purposes other than resolving PDEs. As a result, machine learning can be thought of as an addition to the existing and previously successful toolbox.Recently, partial differential equations can be solved using machine learning algorithms (PDEs). They can be divided into two categories: learning the solution operator and approximating the solution function. The former is exemplified by the Physics-Informed Neural Network (PINN), and the latter is exemplified by the Fourierneural operator (FNO). Both strategies have drawbacks. In PINN, optimization is difficult and prone to error, particularly in multi-scale dynamic systems. Since it performs supervised learning on a predetermined dataset, FNO does not experiencethis optimization problem, although getting such data can be prohibitively expensive or impractical. Here, we merge the operating-learning and function-optimization frameworks to create the physics-informed neural operator (PINO).Here, we merge the operating-learning and function-optimization frameworks to create the physics-informed neural operator (PINO). Over both PINN and FNO models, this integrated strategy increases convergence rates and accuracy. In the operator-learning stage, PINO studies numerous instances of the parametric PDE family to learn the solution operator. For the querying instance of the PDE, PINO optimizes the pre-trained operator ansatz during the test-time optimization phase. Studies reveal that PINO outperforms earlier ML techniques on several well-known PDE families while maintaining FNO's incredible speedup over

solvers. Where other standard ML algorithms fail to converge, PINO successfully solves difficult long temporal transient flows and Kolmogorov flows.What are the advantages and disadvantages of numerical methods to solve partial differential equations Advantages: They are often easy to use.They can produce results quickly.They can solve equations where an analytic solution is impossible.Disadvantages:They are not as elegant as analytic solutions.They do not provide any insight into generalizations.An exact value may not be clear.

It is necessary to do a more in-depth analysis of the convergence rate, convergencetime, and memory need of the approach as it scales to larger dimensions. Before these techniques can be applied independently as plug-and-play in the real world and compete with the well-established solvers, specialised optimizers that can handle the specific loss dynamics also need to be researched, better memory scaling to be achieved, and convergence time to be slashed. Additionally, it is hoped that as the field develops, the lacking "common wisdom" will show up.

2.4 Different kinds of PINN's

PINN solves two main classes of problems data-driven solution and data-driven discovery of partial differential equations. depending on the nature and arrangement of the available data, we devise two distinct sorts of algorithms, namely continuous time and discrete time models. the primary type of model forms a brand new family of data-efficient Spatio-temporal function approximators, while the latter type allows the use of arbitrarily accurate implicit RungeâKutta time-stepping schemes with an infinite number of stages and also XPINNs are called extended physics-informed neural network which is a generalized space-time domain decomposition approach for the physics-informed neural networks (PINNs) to solve nonlinear partial differential equations on arbitrary complex-geometry domains.

2.5 PINN's use cases

1. **Convection Heat Transfer: Unknown Thermal Boundary Conditions.**
a. Forced Convection.
b. Forced Convection in an Enclosure.
c. Flow Past Cylinder.
2. **Two-Phase Stefan Problems**
3. **Applications to Power Electronics**
1. **Heat Transfer in Electronic Chips.**

2.6 Challenges of PINN's

PINN is employed to solve a selected niche of problems only, like where we know the governing equation, initial and boundary conditions and some amount of information regarding the process, it canât solve only data-based problems or only physics-based problems. within the PINN framework, initial and boundary conditions don't seem to be analytically satisfied, thus they have to be included within the loss function of the network to be simultaneously learned with the differential equation (DE) unknown functions. Having competing objectives during the networkâs training can result in unbalanced gradients while using gradient-based techniques, which causes PINNs to often struggle to accurately learn the underlying DE solution.

3 PINN's and Geopolymers

3.1 Mathematical analysis of Geopolymer's

The differential equation for chemical reaction can be written in the form of rate of the reactants we are gaining and negative for the reactants we are losing. Lets look at simple reactions when A reactant change into B reactant with a rate constant K1. So we can write,

$$\delta(A)/\delta(t) = -K1[A] \tag{1}$$

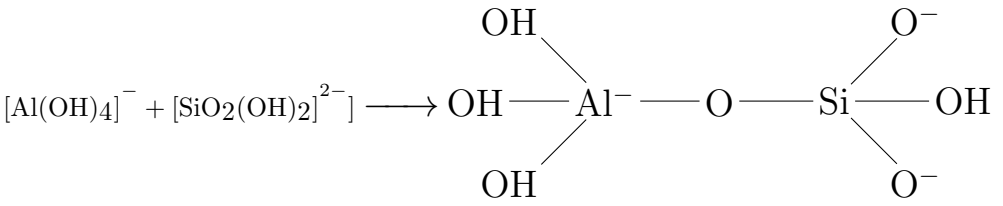
and the [A] depends on the concentration of the reactant A and

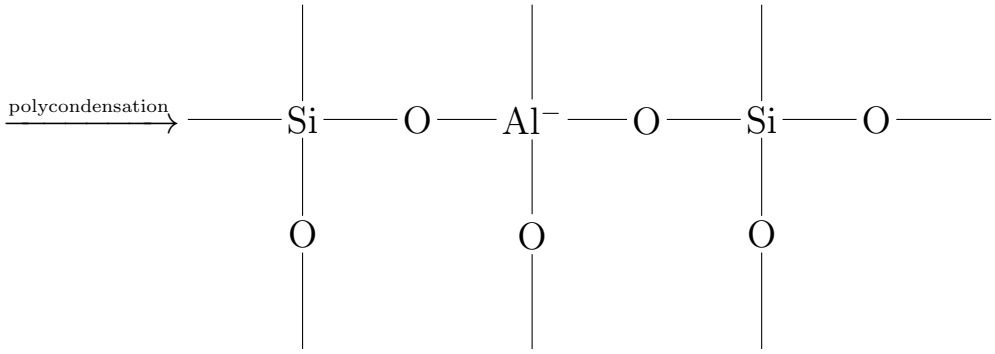
$$\delta(B)/\delta(t) = -K1[B] \tag{2}$$

the [B] depends on the concentration of the reactant B more complex reactions like multiple reactants and intermediate stages and different rate constants with reverse direction reaction which can be also modeled using this type of rate concentration model. Similar kind of analysis should be done for this geopolymer reaction.

Geopolymerisation is a highly complex process, and so a combination of a number of different experimental techniques will be required to provide a detailed understanding of the mechanistic steps involved in the conversion of aluminosilicate precursors to geopolymers. The manipulation of a sample without exposure to an external environment type of analysis of reacting geopolymers slurries by infrared spectroscopy and high-energy X-ray diffractometry provides valuable information on the nanoscale, which is complemented measurements using calorimetry and rheology. Mathematical modeling provides additional insight into the progress of the reactions. To provide a fully detailed understanding of geopolymerisation kinetics, careful application of existing techniques as well as the development of new techniques will be required, as the reaction process is an immensely complex one with processes taking place on a variety of length scales.With these data including the date of the reactants concentration during the reaction and their specific rate constants with the knowledge of initial condition (initial concentration of the reactants) and the boundary conditions (the limits of the concentration of the reactans) we can model the geopolymers mathematically .Now these information gives us the ability to model the kinetics of the system allows the design of mixtures for specific applications, and allows better quality control during production. The testing of the system can be done by comparison of the solution of the differential equations with experimental calorimetric data .The second important thing is modelling the heat equations can help us in the better and faster production of geopolymer.THE CHEMICAL REACTION OF GEOPOLYMER

$$2SiO_2.Al_2O_3 + 3OH^- + 3H_2O \longrightarrow 2[Al.OH_4]^- + [SiO_2.OH_2]^{-2}$$





3.2 Different properties predicted for Geopolymers by NN/PINN

The reaction model involves the time and temperature as the input and the change in concentration as the output the domain of time is between 0 to 1 on the x2 axis and the domain temperature is between -1 to 1 on the x1 axis. On the first graph, the equation was modelled with the approximation that the change in concentration by the first order equation is negligible as the concentrations of the reactants are very small so the temperature change gives a major impact on the change in concentrations.

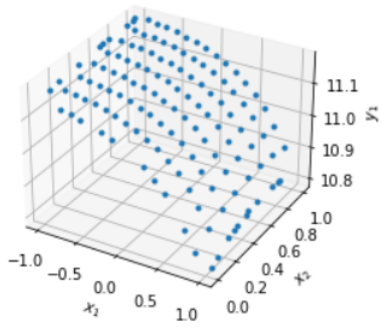


Figure 1: Temperature approximation

From the graph, we can infer that the concentration is changing with the temperature in an inverse parabolic fashion and is mostly the same in all instances of time. On the second graph, the equation was modelled with the approximation that the change in concentration depends upon the first order equation(time and concentration based) and the temperature-dependent Arrhenius equation equally.

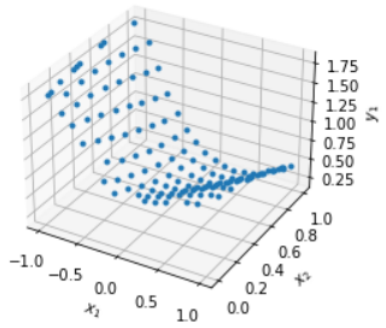


Figure 2: Temperature and fist order approximation

From the graph, we can infer that the change in concentration with respect to temperature is steeper(faster) and the concentration varies with different instances of time also.The boundary and the initial conditions used on both graphs are that at any temperature when time is zero the concentration is the initial concentration and at the temperature is -1(lowest value) the reaction doesnât occur so the concentration doesnât change at any instances of time it remains as the initial concentration and when the temperature reaches the 1(highest temperature) the reaction happens instantaneously so the concentration becomes zero at any instances of time

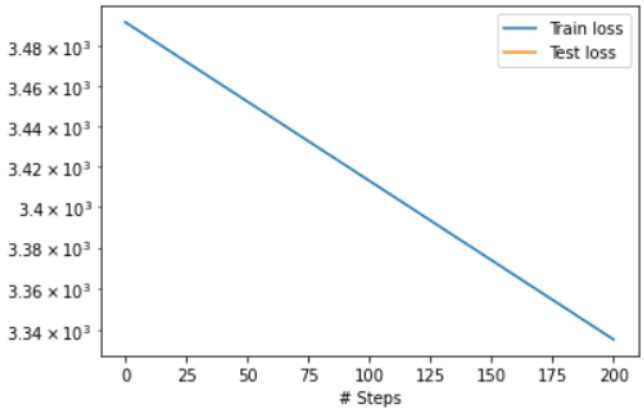


Figure 3: Train loss

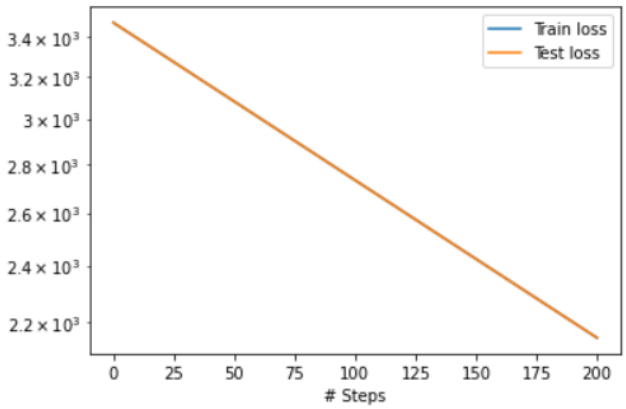


Figure 4: Testing loss

These are the standard results of the good optimisers as the no of steps increases the test loss and train loss decreases linearly

4 Conclusion

With the proper experimental results of the geopolymer reactions, the rate constants and the Arrhenius constant values can be found and the accurate measurement of the concentrations can be done at a lot of time steps as these can be given as the data for the neural network to make it the best optimiser for this reaction model approximation using this we can minimize the concentration of the costly and rare materials by adjusting the temperatures, we can also find the optium temperature and the lower bound and upper bound temperature and a good catalyst can be found using the PINN as we can tell the activation energy this can maximize the production of the geopolymers.

Acknowledgment: .