Me504

Deep Learning

Project Presentation

Supervised by Dr. Manish Agrawal

Prabhreet Kaur
2023MEB1368

-

Adrika Sikidar
2023MEB1326

Me504

Q9:

• Approach-1: Predict the maximum vonMises stress for the given volume fraction distribution using CNN. • Approach-2: Predict the vonMises stress field for a given volume fraction distribution using UNet. Use this approach to finally predict the maximum vonMises stress.

• Which approach is superior, compare the pros and cons.

Predicting Maximum von Mises Stress using CNN & UNet

Problem Statement

Thynk Unlimited

OUR SOLUTION

We aim to solve:

Q9: Predict the maximum von Mises stress for a given volume fraction distribution of two materials in a rectangular domain using:

1. CNN — direct regression model
2. UNet — predicts the stress field, then extracts max von Mises stress

Architecture – CNN Approach (Approach 1)

Model Input
Shape: 21 × 21 × 1
Represents a 2D grid of volume fractions of the composite material.
Reshaped from a flat 1D vector of 441 elements (21×21 = 441).

Input (21x21x1)
↓
Conv2D(32) → ReLU → MaxPooling
↓
Conv2D(64) → ReLU → MaxPooling
↓
Conv2D(128) → ReLU
↓
Flatten
↓
Dense(64) → ReLU
↓
Dense(32) → ReLU
↓
Dense(1) → Output (max stress)

Dataset: 10,000 samples

X: Volume fraction grid (441 values)

y: Maximum von Mises stress (scalar)

Normalization: Mean and standard deviation computed on training set

Reshaping: -1,21,21,1

Split:
- Training: 64%
- Validation: 16%
- Testing: 20%

Layer	Type	Parameters	Output Shape	Description
1	Conv2D	32 filters, 3×3 kernel	(20, 20, 32)	Detects local patterns in the volume fraction grid
2	MaxPooling2D	2×2 pool size	(10, 10, 32)	Reduces spatial dimensions by 2x
3	Conv2D	64 filters, 3×3 kernel	(10, 10, 64)	Learns more abstract features
4	MaxPooling2D	2×2 pool size	(5, 5, 64)	Further spatial reduction
5	Conv2D	128 filters, 3×3 kernel	(5, 5, 128)	Deep feature extraction
6	Flatten	-	(3200,)	Converts 3D tensor to 1D vector
7	Dense	64 neurons	(64,)	Fully connected layer
8	Dense	32 neurons	(32,)	Further abstraction
9	Dense (Output)	1 neuron (linear activa	(1,)	Final output: max von Mises stress

Loss: Mean Squared Error (MSE)
Optimizer: Adam (adaptive learning rate)
Metric: Custom R^2 score (coefficient of determination)

Architecture – Unet Approach (Approach 2)

Model Input
Shape: 21 × 21 × 1 (volume fraction distribution)

- Predict the complete stress field from the volume fraction grid
- Then compute the maximum von Mises stress from the output stress field

- Encoder: Extracts deep spatial features via downsampling
- Bridge: Bottleneck for abstract representation
- Decoder: Reconstructs the stress field via upsampling
- Skip Connections: Combine encoder features to preserve spatial information

Dataset: 10,000 samples

X: Volume fraction grid (21 *21 *1 values)

y: total stress field

Normalization: Mean and standard deviation computed on training set

encoder

Block	Layers	Output Shape	Description
c1	Conv2D(16) ×2 + BatchNorm + ReLU	(21, 21, 16)	Feature extraction
p1	MaxPooling2D	(11, 11, 16)	Downsample
c2	Conv2D(32) ×2 + BatchNorm + ReLU	(11, 11, 32)	Deeper features
p2	MaxPooling2D	(6, 6, 32)	Downsample
c3	Conv2D(64) ×2 + BatchNorm + ReLU	(6, 6, 64)	Deeper
p3	MaxPooling2D	(3, 3, 64)	Downsample

bottleneck

Block	Layers	Output Shape	Description
c4	Conv2D(128) ×2 + BN + ReLU	(3, 3, 128)	Bottleneck

decoder

Block	Layers	Output Shape	Notes
u5	Conv2DTranspose(64, stride=2) + concat with c3 + conv_block	(6, 6, 64)	Recover resolution
u6	Conv2DTranspose(32, stride=2) → crop to (11×11) + concat c2	(11, 11, 32)	Match dimensions
u7	Conv2DTranspose(16, stride=2) → crop to (21×21) + concat c1	(21, 21, 16)	Final decoder step

output

Layer	Type	Filters	Output Shape
26	Conv2D	1	(21, 21, 1)

Total Layers: 26
Activation Function: ReLU (intermediate), Linear (output)
Skip Connections: Used at 3 levels (deep → shallow)
Final Output: Full field prediction of von Mises stress

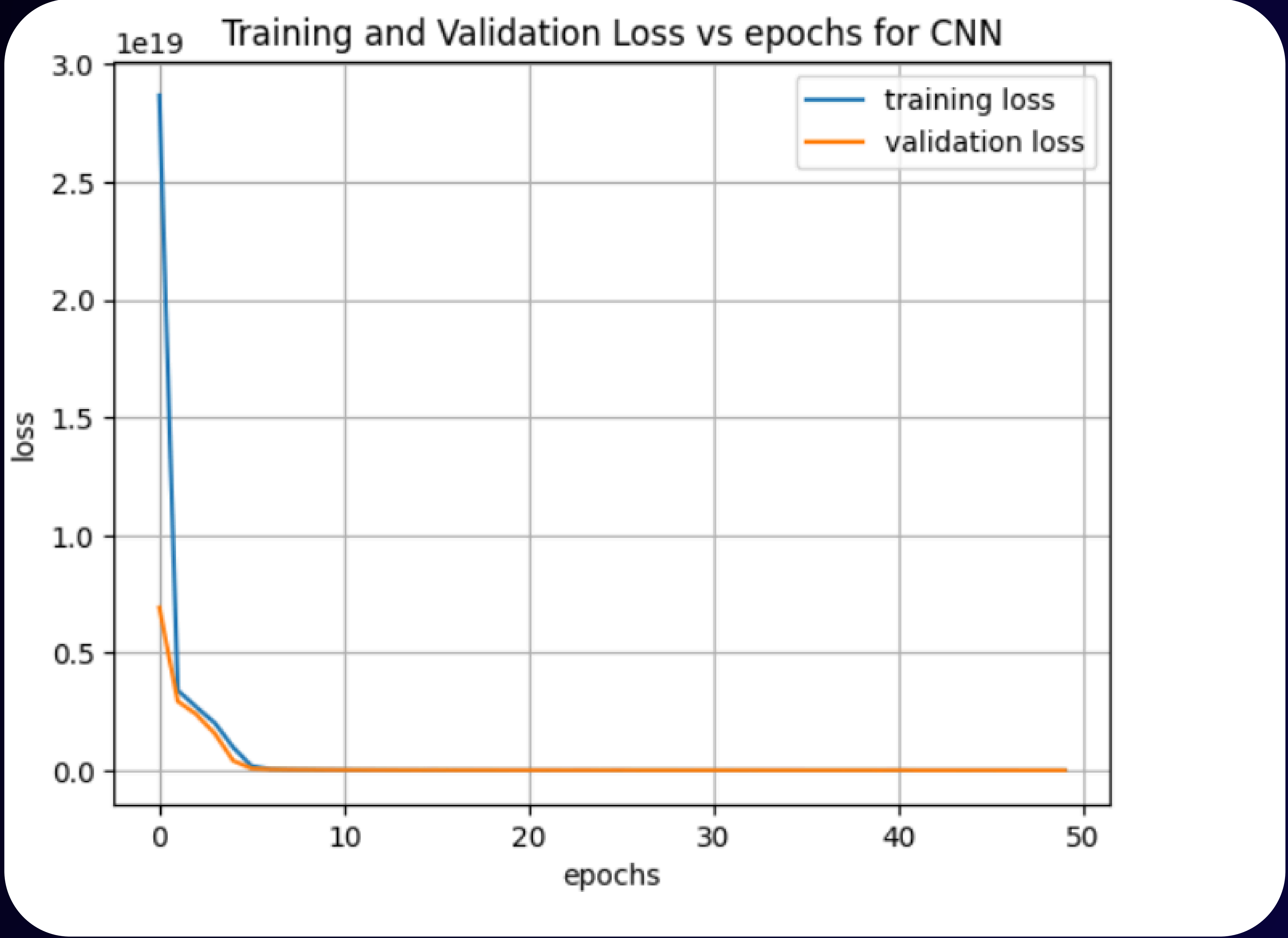
From the predicted stress field, compute:
tau max with formula
Extract max(τ_vm) → final prediction

Thynk Unlimited

Results

Graph and Values

```
For CNN:
R2 score:  0.830542697227304
MSE Loss:  2.1717259124557084e+16
MAE Loss:  111409535.77637118
```



Issue with UNet

For our UNet architecture, we kept facing issues in the shape of the predicted and true outputs since we had to resize the tensor such that it is divisible by $2^{2.}$.

So the predicted output was of dimension larger than the true output. We were not able to resolve this issue.

Thank You