

CS3317 Artificial Intelligence

Coursework 4: Convolutional Neural Network

[Example Report — For Reference Only]

Name: Zhang San Student ID: 522030XXXXX

May 27, 2026

Note: This is an example report showing the expected structure and depth of analysis. All numerical results shown as `__` are placeholders—replace them with your own experimental results. Do not copy the analysis verbatim; write your own observations based on your experiments.

1 CNN Architecture

1.1 SimpleCNN

[Describe your SimpleCNN architecture in a layer-by-layer format. Include the input/output shape at each stage so the reader can follow the tensor dimensions. For example:]

```
Input:  (input channels) x 28 x 28
Conv1:  ... filters, ...x... kernel  -> ... x ... x ...
Activation + Pooling                  -> ... x ... x ...
Conv2:  ... filters, ...x... kernel  -> ... x ... x ...
Activation + Pooling                  -> ... x ... x ...
Flatten                                -> ...
FC1:    ... -> ..., Activation
FC2:    ... -> 10
```

Total parameters: _____

Design choices: *[Explain why you chose this architecture. Consider: Why this number of conv layers for 28×28 images? Why increase filter count as spatial dimensions shrink? Why this kernel size? What does max pooling provide?]*

1.2 Training Results

Table 1: SimpleCNN results.

Metric	Value
Best Val Accuracy	__._ %
Test Accuracy	__._ %
Training Time	~__ min

[Briefly discuss: How quickly does the model converge? Is there a gap between training and validation accuracy (overfitting)?]

[Insert your training_summary.png here]

Figure 1: SimpleCNN training summary: loss (left) and accuracy (right).

2 Hyperparameter Exploration

[Explore at least 2 of the following: optimizer choice, learning rate, architecture variations, batch size. For each, provide a comparison table and analysis.]

2.1 Optimizer Comparison

Table 2: Optimizer comparison on SimpleCNN.

Optimizer	Learning Rate	Test Acc	Convergence
-----	-----	---.-%	~__ epochs
-----	-----	---.-%	~__ epochs
-----	-----	---.-%	~__ epochs

[Insert your optimizer comparison plot here]

Figure 2: Training curves for different optimizers.

Analysis: *[Compare convergence speed and final accuracy. Why might adaptive optimizers behave differently from SGD on CNNs?]*

2.2 Architecture Variations

[Try different model sizes or configurations. Show the tradeoff between model capacity and performance.]

Analysis: *[Are there diminishing returns as model size increases? Is FashionMNIST complex enough to benefit from larger models?]*

Table 3: Architecture exploration.

Config	Test Acc	#Params	Notes
-----	__._ %	-----	-----
-----	__._ %	-----	-----
-----	__._ %	-----	-----

3 Data Augmentation

[Describe the augmentation transforms you applied (e.g., horizontal flip, rotation, translation) and why they make sense for this dataset.]

Table 4: Effect of data augmentation on SimpleCNN.

Setting	Test Accuracy	Train-Val Gap
No augmentation	__._ %	__._ %
With augmentation	__._ %	__._ %

[Discuss: Does augmentation improve test accuracy? Does it reduce overfitting (smaller train-val gap)? Which augmentations are most effective and why?]

4 DeepCNN (Bonus)

[Describe your DeepCNN architecture. What advanced techniques did you use (e.g., batch normalization, dropout, residual connections, adaptive pooling)? Explain the motivation for each design choice.]

Table 5: DeepCNN results.

Metric	Value
Best Val Accuracy	__._ %
Test Accuracy	__._ %
Training Time	~__ min

[Compare with SimpleCNN. How much does the advanced architecture improve accuracy? Is the improvement worth the added complexity?]

5 Cross-Coursework Comparison

[Discuss the following:]

- [Why does CNN outperform MLP on image data? Consider: spatial structure preservation, parameter sharing, translation invariance via pooling.]*
- [How does the parameter efficiency of CNNs compare to fully connected layers?]*
- [What is the key takeaway from the progression: linear model $\rightarrow$ MLP $\rightarrow$ CNN? What role does inductive bias play?]*

Table 6: Performance comparison across all courseworks on FashionMNIST.

Method	CW	Test Accuracy	#Parameters
Logistic Regression	CW2	__._ %	_____
MLP	CW3	__._ %	_____
SimpleCNN	CW4	__._ %	_____
DeepCNN (bonus)	CW4	__._ %	_____

[Insert a bar chart comparing test accuracy across CWs]

Figure 3: Test accuracy progression from logistic regression to CNN.

6 Conclusion

[Summarize 2–4 key findings across the entire CW2–CW4 progression. For example:]

- [How does the CNN compare to previous methods, and what architectural properties drive the improvement?]*
- [What did you learn about optimizer and hyperparameter selection for CNNs?]*
- [How does data augmentation act as a regularizer?]*
- [What is the overarching lesson from the linear $\rightarrow$ MLP $\rightarrow$ CNN progression about matching model inductive bias to data structure?]*