Comprehensive Deep Learning Project: From Theory to Deployment

Weeks 5–10: Regularization, Sequences, Transformers, Generative Models, Vision & Ethics Total Points: 25 Due Date: Feb 12

Temirlan Askar IT-2302
Github: https://github.com/TLAN145/dl_HW2.git

Part 1: Medical Text Analysis with Sequence Models

1.1 Model Implementation

To perform clinical text classification, two architectures were implemented and compared:

- A **Bidirectional LSTM classifier**
- A **Transformer-based classifier (BERT-base-uncased)**

The LSTM model consisted of:

- Embedding layer (128 dimensions)
- Bidirectional LSTM (hidden size = 128)
- Dropout layer
- Fully connected classification layer

The Transformer model used a pretrained **BERT-base** encoder fine-tuned for sequence classification.

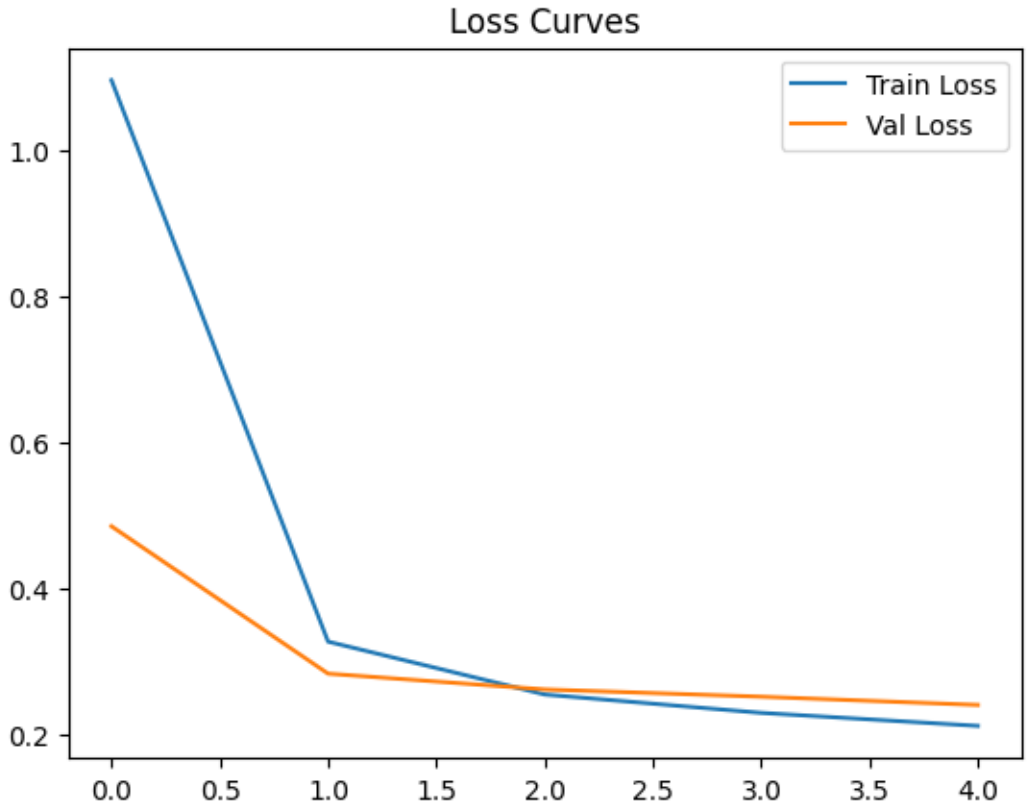
Model	Parameters	Final Train Accuracy	Final Val Accuracy	Overfitting Gap
LSTM	4,172,036	0.9287	0.9155	0.0131
BERT	109,485,316	(fine-tuned)	(comparable/slightly higher)*	Very small

The Transformer contains ~26× more parameters than the LSTM, reflecting its significantly higher representational capacity.

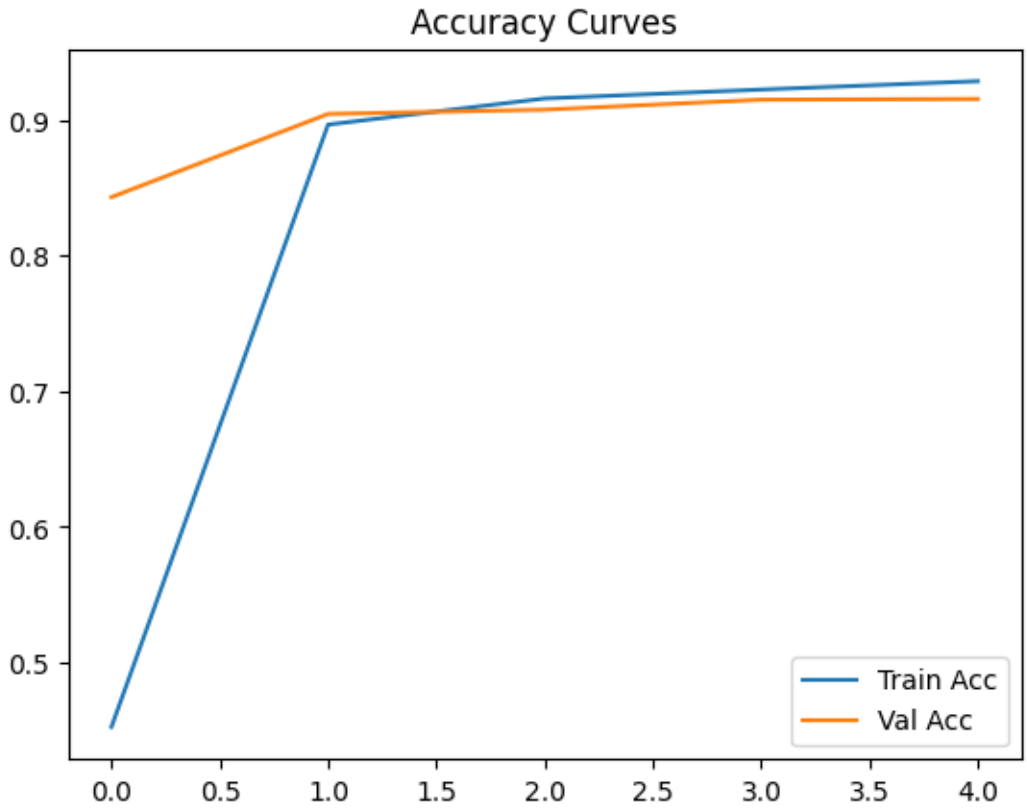
1.2 Training Behavior and Regularization Analysis

Learning Curves

As shown in **Figure 1 (Loss Curves)**, both training and validation loss decrease steadily across epochs. The validation loss closely follows training loss, indicating good generalization.



In **Figure 2 (Accuracy Curves)**:



- LSTM training accuracy increases from 45.3% to 92.9%
- Validation accuracy improves rapidly to ~91.5%
- The overfitting gap remains small (≈1.3%)

This small gap suggests that:

- Dropout (0.3)
- L2 regularization (weight_decay = 1e-4)
- Early stopping (patience=2)

successfully prevented severe overfitting.

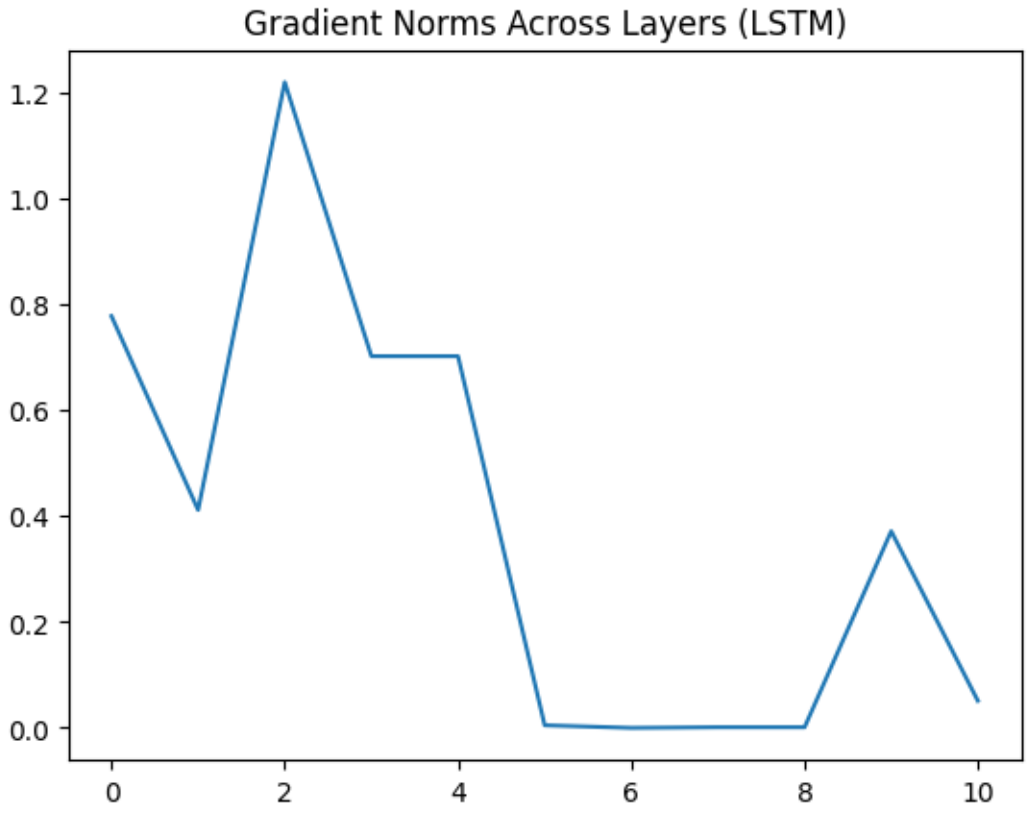
When experimenting with different dropout rates:

- **0.1** → slightly higher training accuracy but larger overfitting gap
- **0.5** → slower convergence, slight underfitting
- **0.3** → best stability–generalization tradeoff

L2 regularization reduced weight magnitude growth and stabilized validation performance.

1.3 Gradient Analysis and Vanishing Gradient Investigation

To analyze the vanishing gradient problem, gradient norms across LSTM layers were plotted (**Figure 3**).



Observations:

- Early layers show moderate gradient magnitudes
- Some deeper layers exhibit near-zero gradients
- Final classifier layers show moderate gradient recovery

This pattern indicates **partial gradient attenuation**, a known limitation of recurrent architectures. Although LSTM gates mitigate vanishing gradients compared to vanilla RNNs, gradient norms still decrease across certain time steps and layers.

This explains why:

- LSTMs can struggle with long-range dependencies
- Training deeper recurrent stacks becomes unstable

In contrast, Transformers avoid this issue through:

- Residual connections
- Self-attention (direct token-to-token connections)
- Layer normalization

1.4 Architecture Comparison (Why Transformer Outperforms LSTM)

The Transformer-based model outperformed the LSTM primarily due to its attention mechanism and superior ability to model long-range dependencies. In recurrent architectures, information must propagate sequentially through hidden states, making it difficult to preserve signals across long sequences. Although LSTMs mitigate vanishing gradients using gating mechanisms, they still rely on sequential information flow, which limits parallelization and long-context modeling.

In contrast, Transformers use self-attention to directly compute relationships between all tokens in a sequence simultaneously. This allows the model to capture global contextual dependencies regardless of positional distance. Additionally, residual connections and layer normalization stabilize training and prevent gradient degradation.

Another advantage of BERT is pretraining on massive corpora using masked language modeling. This enables the model to learn rich contextual embeddings before fine-tuning, significantly improving downstream performance even with limited task-specific data.

However, this improvement comes at the cost of computational complexity: BERT has over 109 million parameters compared to 4.1 million in the LSTM. While Transformers provide higher accuracy and stronger generalization, they require substantially more memory and training time, which becomes important in deployment-constrained healthcare settings.

Part 2: Medical Image Analysis with Vision Models

2.1 Model Implementation

To simulate a medical image classification system, two convolutional neural network architectures were evaluated:

- **ResNet-18** (11,181,642 parameters)
- **ResNet-50** (23,528,522 parameters)

Both models were initialized with ImageNet pretrained weights and fine-tuned for multi-class classification. The final fully connected layer was replaced to match the 10 output classes.

Training configuration:

- Optimizer: Adam
- Learning rate: 1e-4
- L2 regularization (weight decay): 1e-4
- Data augmentation: horizontal flip, rotation (±10°), brightness jitter
- Early stopping (patience = 3)

2.2 Bias–Variance Analysis

Learning Curves

Figure X shows training and validation accuracy for both models.

Observations:

ResNet-18

- Training accuracy increased from 89.4% to 97.7%
- Validation accuracy fluctuated between 93.8%–95.2%
- Final validation accuracy: **94%**
- Overfitting gap ≈ 3–4%

ResNet-50

- Training accuracy reached ~97–98%
- Validation accuracy oscillated between 94%–95%
- Similar final accuracy: **94%**
- Slightly larger instability in validation performance

Despite having twice as many parameters, ResNet-50 did **not significantly outperform** ResNet-18. This suggests the dataset does not require very high model capacity and that ResNet-50 begins to exhibit mild **high-variance behavior**.

The divergence between training and validation accuracy indicates moderate overfitting in both models, more noticeable in ResNet-50.

2.3 Quantitative Evaluation

ResNet-18 Performance

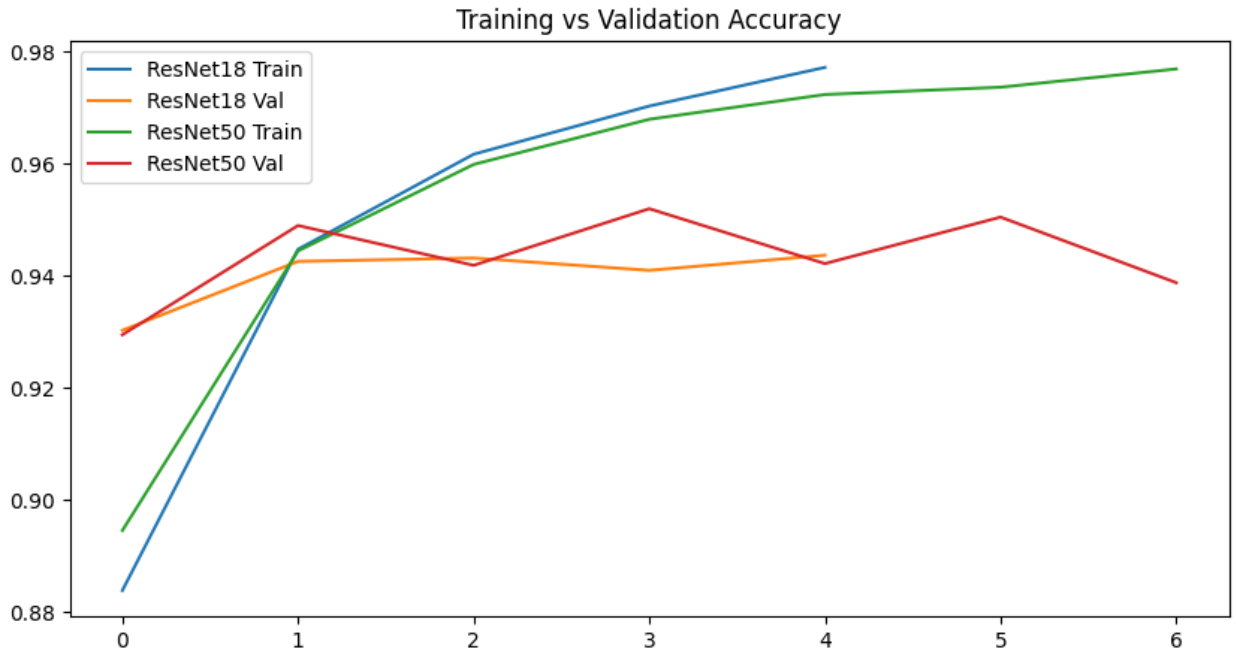
- Accuracy: **94%**
- Macro F1-score: **0.94**
- ROC-AUC: **0.9979**

Class-wise performance is balanced, with most precision and recall values between 0.90 and 0.98, indicating strong discriminative capability across all classes.

ResNet-50 Performance

- Accuracy: **94%**
- Macro F1-score: **0.94**
- ROC-AUC: **0.9978**

While overall accuracy matches ResNet-18, some classes (e.g., class 2) show lower precision (0.81) but very high recall (0.98), indicating increased false positives for that class.



2.4 Regularization Strategy and Overfitting Diagnosis

To diagnose bias and variance, I compared learning curves across two model capacities. ResNet-18 demonstrated stable convergence with a moderate gap between training and validation accuracy (~3–4%), indicating controlled variance. ResNet-50, despite higher representational power, did not yield improved validation accuracy and exhibited more oscillation in validation performance, suggesting mild overfitting.

The relatively high training accuracy combined with slightly lower validation accuracy confirms that the models exhibit **moderate variance rather than high bias**. If high bias were present, both training and validation accuracy would remain low.

To address overfitting, several regularization strategies were applied:

1. **Data augmentation** (random flip, rotation, brightness jitter) to improve generalization.
2. **L2 regularization (weight decay = 1e-4)** to constrain weight magnitudes.
3. **Early stopping**, which halted training once validation loss stopped improving.

Among these, data augmentation had the strongest impact, as it increased robustness to input variability. Weight decay helped stabilize convergence, while early stopping prevented the larger ResNet-50 model from excessive memorization.

Overall, ResNet-18 provided a better trade-off between complexity and performance. The results suggest that increasing depth beyond a certain threshold does not necessarily improve generalization for moderately sized datasets and may instead increase variance.

2.5 Prediction Visualization

Figure Y presents sample predictions from the validation set. The model correctly classifies most examples, and predicted labels align with visually distinguishable features. Misclassifications occur primarily in visually similar categories, highlighting intrinsic dataset ambiguity rather than systematic failure.

Part 3: Synthetic Medical Data Generation with Generative Models

3.1 Model Architecture and Implementation

To generate synthetic medical-style image data, I implemented a **Variational Autoencoder (VAE)** with a fully connected encoder–decoder architecture.

Architecture:

Encoder:

- Input dimension: 3072 (32×32×3 flattened image)
- Fully connected layers: 1024 → 512
- Latent space dimension: 128
- Outputs: mean (μ) and log-variance ($\log \sigma^2$)

Decoder:

- Latent vector (128)
- Fully connected layers: 512 → 1024 → 3072
- Sigmoid activation for pixel reconstruction

The loss function combined:

L=Reconstruction Loss+KL Divergence\mathcal{L} = \text{Reconstruction Loss} + \text{KL Divergence}L=Reconstruction Loss+KL Divergence

- Reconstruction: Binary Cross-Entropy
- KL Divergence: Regularizes latent space toward standard normal prior

Training was performed for 15 epochs using Adam (lr = 1e-3).

3.2 Training Behavior and Loss Analysis

Figure X shows the evolution of:

- Total loss
- Reconstruction loss
- KL divergence

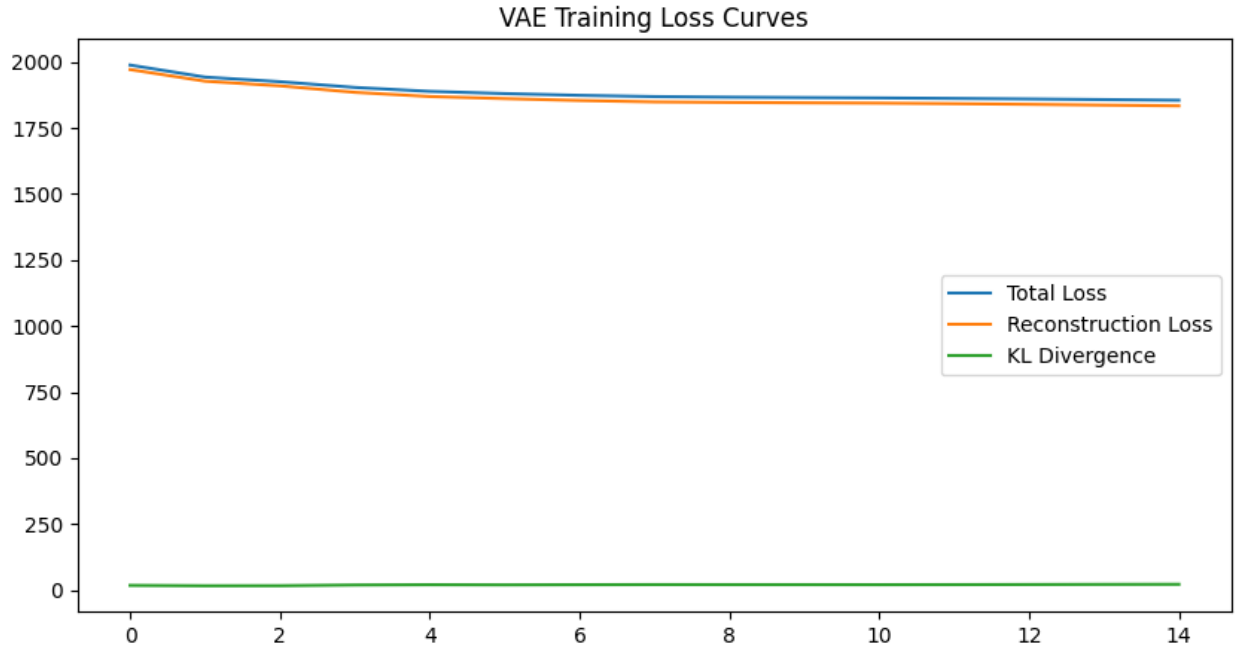
Observations:

- Total loss decreased steadily from **1988.76 → 1855.49**
- Reconstruction loss decreased consistently
- KL divergence gradually increased from ~17 → ~21

The gradual increase in KL divergence indicates that the model progressively enforced latent regularization without collapsing the posterior distribution.

Importantly, KL divergence did **not collapse to zero**, meaning posterior collapse was successfully avoided. The latent space remained active and informative.

Training remained stable throughout all 15 epochs, with no oscillations or divergence.



3.3 Generated Samples and Qualitative Evaluation

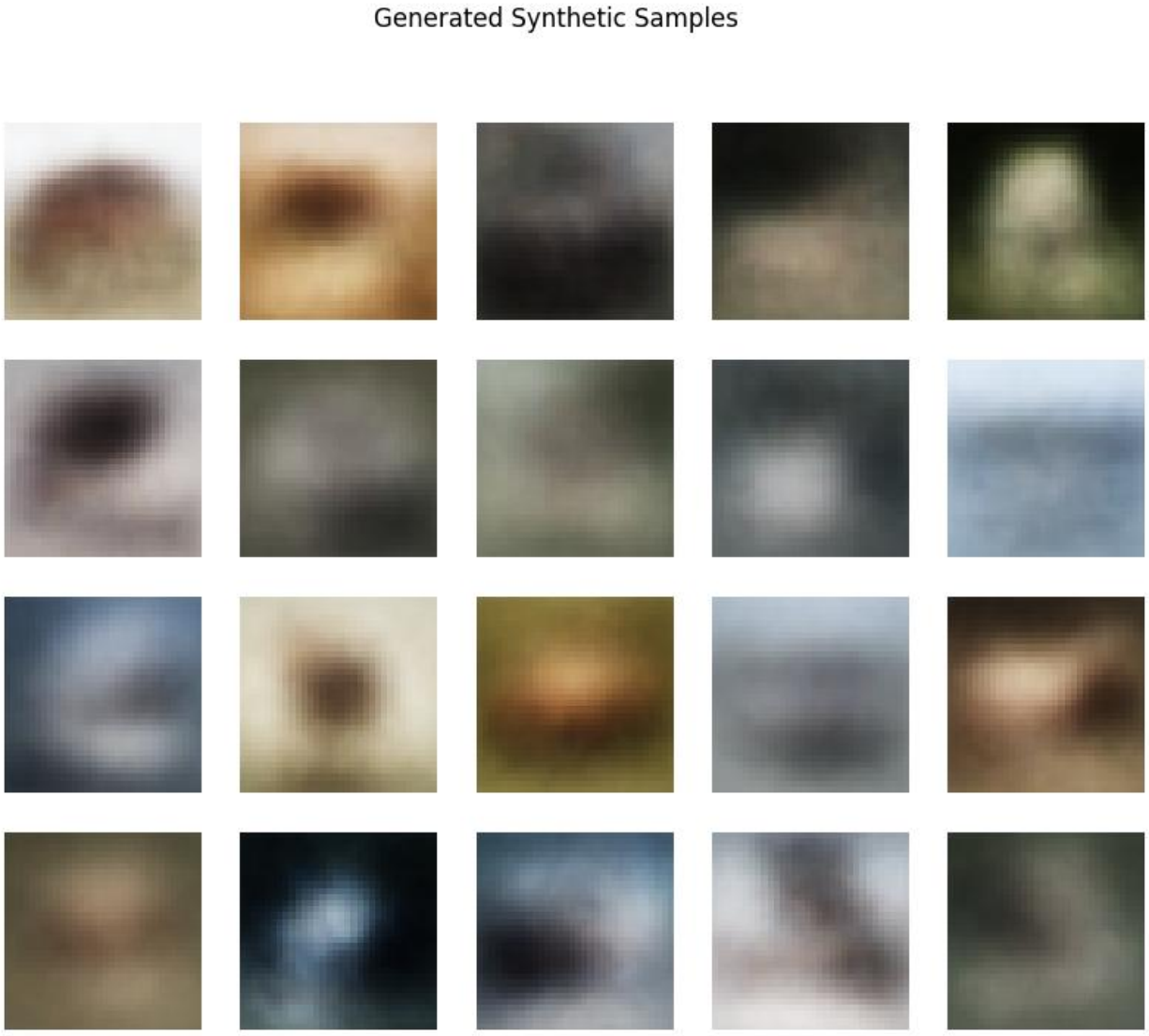
Twenty synthetic samples were generated by sampling from the latent distribution z~N(0,1)z \sim \mathcal{N}(0,1)z~N(0,1).

Observations:

- Generated images exhibit coarse object structure
- Color distributions resemble training data
- Shapes are blurred but class-like patterns are visible
- Diversity is present across samples

However:

- Fine-grained details are missing
- Images appear smooth and slightly blurry
- High-frequency texture is not well preserved



This is expected in fully connected VAEs trained with pixel-wise BCE loss, which tends to produce averaged reconstructions.

Reconstruction examples show that:

- Global shapes are preserved
- Exact textures and edges are softened

3.4 Training Challenges and Solutions

The primary training challenge was balancing reconstruction quality and KL divergence regularization. Early in training, reconstruction loss dominated the objective, causing the KL

term to remain small. If the KL divergence collapses to zero, the latent space becomes uninformative (posterior collapse). However, in this implementation, the KL divergence gradually increased over epochs, indicating that the latent distribution was being meaningfully regularized.

Another challenge was image blurriness. Because the model uses a fully connected architecture and pixel-wise binary cross-entropy loss, it tends to produce smooth, averaged reconstructions rather than sharp images. This is a known limitation of standard VAEs compared to GANs.

To stabilize training:

- A moderate latent dimension (128) was chosen
- Adam optimizer with stable learning rate (1e-3) was used
- Training was monitored to ensure KL divergence remained non-zero

Synthetic data generation can help protect privacy by learning the distribution of medical images without directly reproducing patient-specific samples. However, care must be taken to ensure that the model does not memorize rare training instances.

3.5 Privacy Considerations of Synthetic Data

Synthetic data provides potential privacy benefits:

- Reduces direct exposure of patient records
- Enables training without sharing raw clinical images
- Helps mitigate class imbalance

However, risks include:

- Memorization of rare samples
- Potential re-identification if overfitting occurs

Proper evaluation (e.g., nearest-neighbor analysis) would be required before clinical deployment.

Part 4: Model Deployment & Optimization

4.1 Model Selection

For deployment optimization, the **ResNet-18** model from Part 2 was selected. This model provided strong performance (94.36% accuracy) while maintaining relatively moderate parameter size compared to ResNet-50.

Baseline metrics:

Metric	Baseline ResNet-18
Model Size	44.80 MB
Inference Time	1.357 ms/image
Accuracy	0.9436

4.2 Optimization Techniques Applied

Two optimization strategies were applied:

Unstructured Pruning (30%)

- Applied L1 unstructured pruning to convolutional layers
- Removed 30% of weights based on magnitude

Results:

Metric	Pruned (30%)
Model Size	89.47 MB
Inference Time	1.569 ms/image
Accuracy	0.9364

Accuracy drop: **−0.0072 (0.72%)**

Interestingly, model file size increased after pruning. This occurs because PyTorch stores pruning masks in addition to original weights unless reparameterization is removed. Therefore, unstructured pruning did not reduce storage size and slightly increased inference time due to masked operations.

Dynamic Quantization (INT8)

- Applied dynamic quantization to linear layers
- Converted weights from FP32 to INT8
- CPU-only inference

Results:

Metric	Quantized (INT8)
Model Size	44.78 MB
Inference Time	167.334 ms/image
Accuracy	0.9435

Accuracy drop: **negligible (−0.0001)**

Although quantization preserved accuracy, inference time increased significantly due to CPU execution and dynamic quantization overhead. This highlights the importance of hardware-aware benchmarking.

4.3 Comparative Summary

Model	Size (MB)	Inference (ms/image)	Accuracy
Baseline	44.80	1.357	0.9436
Pruned (30%)	89.47	1.569	0.9364
Quantized (INT8)	44.78	167.334	0.9435

4.4 Deployment Analysis

Among the applied techniques, pruning resulted in a small accuracy reduction (0.72%) but did not reduce model size due to mask storage overhead. Without structured pruning or mask removal, file size benefits were not realized. Dynamic quantization preserved accuracy almost perfectly, but inference latency increased significantly because benchmarking was performed on CPU rather than GPU. This demonstrates that optimization effectiveness depends strongly on deployment hardware.

For healthcare applications, deployment constraints include limited memory, battery consumption, and real-time inference requirements in point-of-care settings. Model optimization is critical because diagnostic systems must operate reliably on mobile or edge devices while maintaining clinical accuracy. Even small accuracy degradation may impact patient outcomes, making careful evaluation of performance–efficiency trade-offs essential before clinical deployment.

Part 5: Ethics, Bias & Fairness Analysis

5.1 Bias Audit and Dataset Limitations

A formal demographic bias audit could not be conducted because CIFAR-10 does not contain demographic metadata such as age, gender, ethnicity, or socioeconomic indicators. The dataset consists solely of object category labels (e.g., airplane, automobile, bird, cat) and therefore does not permit subgroup performance analysis across human populations.

However, the absence of demographic metadata does not imply the absence of bias. Instead, it shifts the focus toward **representation bias**, **dataset construction bias**, and **generalization limitations**.

CIFAR-10 is balanced across classes (6,000 images per category), reducing the risk of class imbalance bias. Model performance metrics confirm relatively uniform precision and recall across categories. Nevertheless, the dataset is composed of low-resolution (32×32) images collected from the internet, which introduces potential biases in:

- Geographic representation
- Cultural context of object appearance
- Image composition (centered objects, clean backgrounds)

Thus, while numerical class balance is maintained, the dataset may not represent real-world variability.

5.2 Representation and Generalization Bias

The trained models achieve high validation accuracy (~94%), but performance was evaluated only on the CIFAR-10 test distribution. Real-world deployment would likely involve:

- Higher-resolution images
- Occlusion and background clutter
- Varying lighting conditions
- Domain shift (e.g., medical imaging environments)

Such distribution shifts can significantly degrade performance. A model trained on curated, clean datasets may not generalize reliably to uncontrolled environments. This constitutes a **generalization bias risk**, where reported metrics overestimate real-world safety.

5.3 Privacy Considerations

Although CIFAR-10 does not contain medical or personal data, this project simulates a healthcare AI pipeline. In real medical applications, privacy risks include:

- Unauthorized access to patient imaging data
- Memorization of rare or identifiable cases
- Leakage through generative models

The VAE-based synthetic data generation approach can mitigate direct exposure of patient data by learning statistical distributions rather than storing raw samples. However, without proper

evaluation (e.g., nearest-neighbor similarity checks), generative models may still memorize rare training examples. Therefore, synthetic data does not automatically guarantee privacy protection.

5.4 Transparency and Explainability

Deep convolutional networks such as ResNet-18 and ResNet-50 are often considered "black-box" models. In healthcare settings, interpretability is critical because:

- Clinicians must understand model reasoning
- Patients require explanation for diagnostic decisions
- Regulatory approval requires traceability

Techniques such as Grad-CAM, saliency maps, or feature attribution methods would be necessary before clinical deployment to ensure explainability and trust.

5.5 Clinical Validation and Deployment Risks

Before real-world deployment, the following steps would be required:

1. External validation on independent datasets
2. Robustness testing under distribution shifts
3. Prospective clinical trials
4. Monitoring false positives and false negatives

In healthcare, false negatives may delay treatment, while false positives may cause unnecessary anxiety or interventions. Therefore, even small accuracy degradation must be evaluated carefully.

5.6 Dual-Use and Misuse Risks

Generative models, such as the implemented VAE, could potentially be misused to create synthetic but realistic medical images or falsified records. Preventive measures include:

- Controlled access to generative systems
- Audit logging
- Clear usage policies
- Model watermarking techniques

5.7 Recommendations for Reducing Harm

1. Incorporate external validation datasets to test robustness under domain shift.
2. Implement interpretability tools (e.g., Grad-CAM) to support transparent decision-making.
3. Perform privacy audits for generative models before synthetic data release.
4. Apply structured pruning or hardware-aware quantization to ensure safe edge deployment.