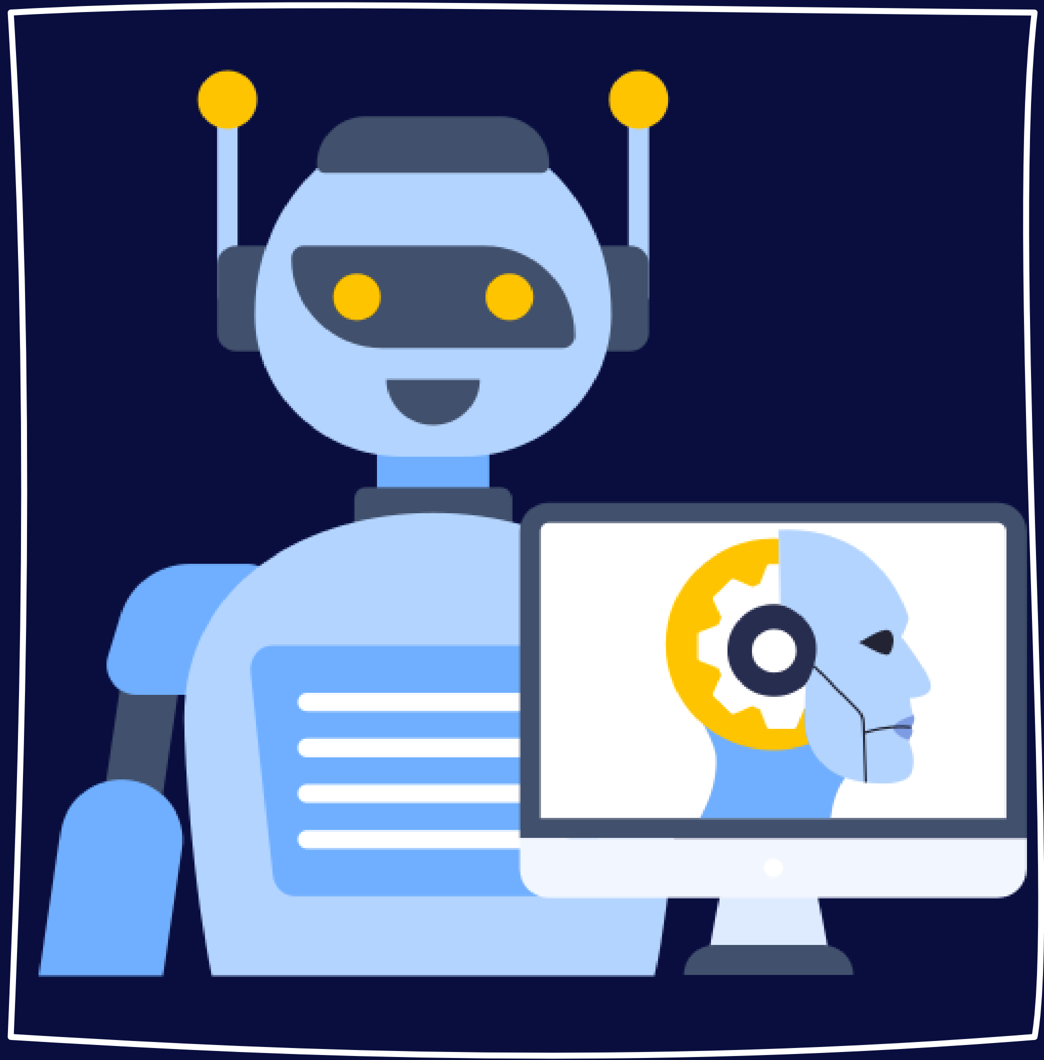
image classification

Ahmed Khaled Abd Ali
Hamada Ali Hamada
Ameer Wiesam Hathal



Introduction to Machine Learning

1

What is Machine Learning?

Machine Learning is a field of artificial intelligence that enables systems to learn from data, identify patterns, and make decisions with minimal human intervention. It's about teaching computers to **learn**, **adapt**, and **improve** over time.

2

How Does it Work?

At its core, ML involves feeding large **datasets** to algorithms. These algorithms then analyze the data, build models based on observed patterns, and use these models to predict or classify new, unseen data.

3

Key Applications

From personalized recommendations and medical diagnosis to self-driving cars and fraud detection, ML is transforming industries and everyday life. It empowers us to solve complex problems and automate intelligent tasks.

Navigating Training Challenges

Data Imbalance

We encountered significant issues with skewed datasets, where some classes had far more examples than others. This led to models biased towards dominant classes, impacting overall performance.

Overfitting

Our models sometimes learned the training data too well, memorizing noise and failing to generalize to new images. This resulted in poor performance on unseen data, a common hurdle in ML.

Computational Limits

Training large models on extensive datasets demanded substantial computational resources. Limited GPU availability and processing power often constrained our experimentation and iteration speed.

Project Overview

Our Starting Point

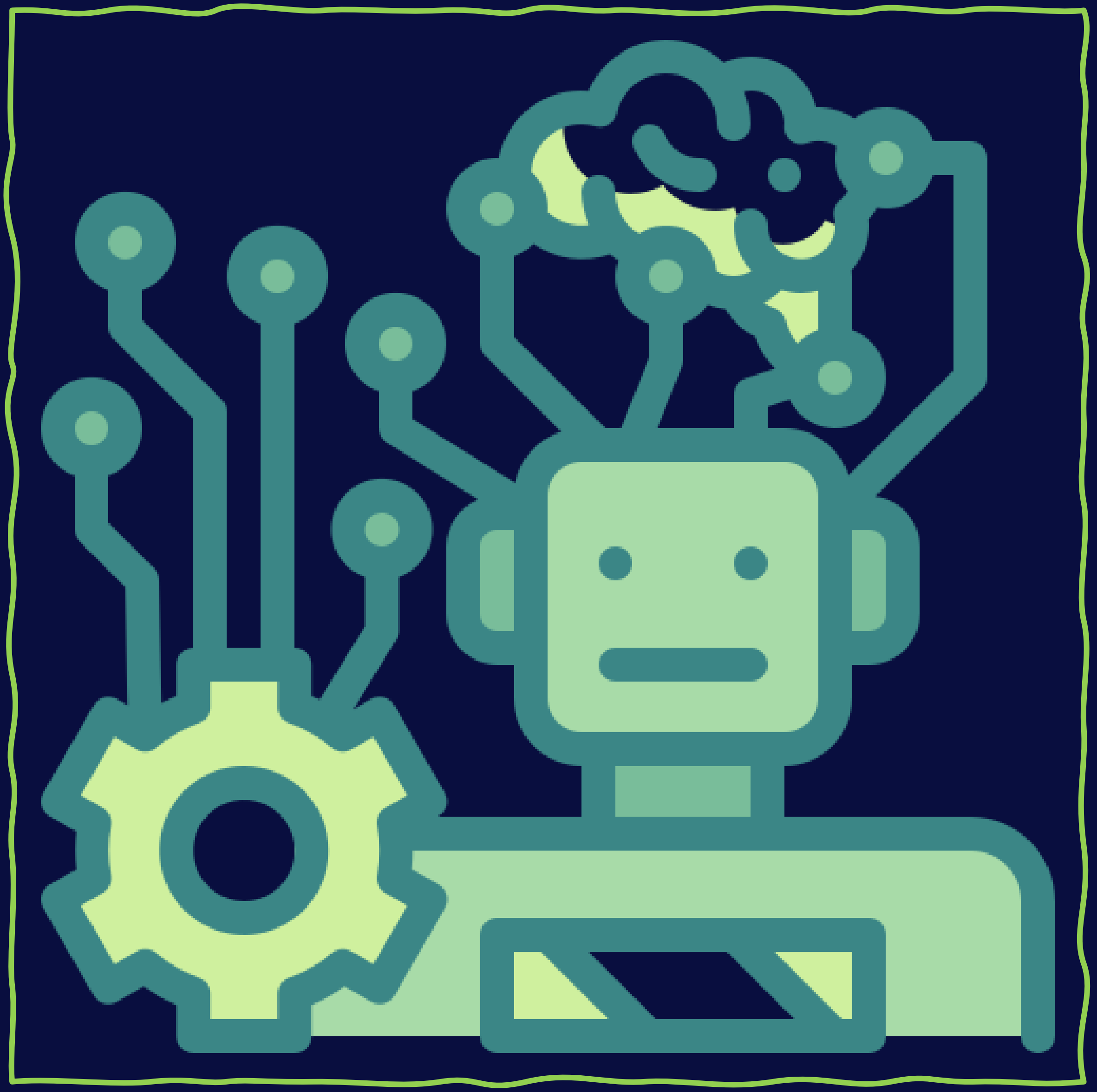
Our project is an image classification type. The goal was to develop a robust model capable of accurately identifying and categorizing objects within a diverse set of images.

In our project, we used a special data set called CUB_200_2011, which is a practical data set. This data set contains 200 classes, each class representing a type of bird.

From MobileNet to Custom Solution

Original MobileNet Limitations

Our initial approach leveraged MobileNet, known for its efficiency. However, it should performance very good on our specific dataset, But this is not the case:
The project began with the MobileNet architecture, trained on a laptop using an Integrated GPU.
Result : Despite two weeks of tuning, the maximum validation accuracy achieved was only 6%.
Solution : A pivot to the EfficientNetB3 model was implemented to leverage its higher capacity.
Impact : This model switch immediately boosted the baseline accuracy to 80%.



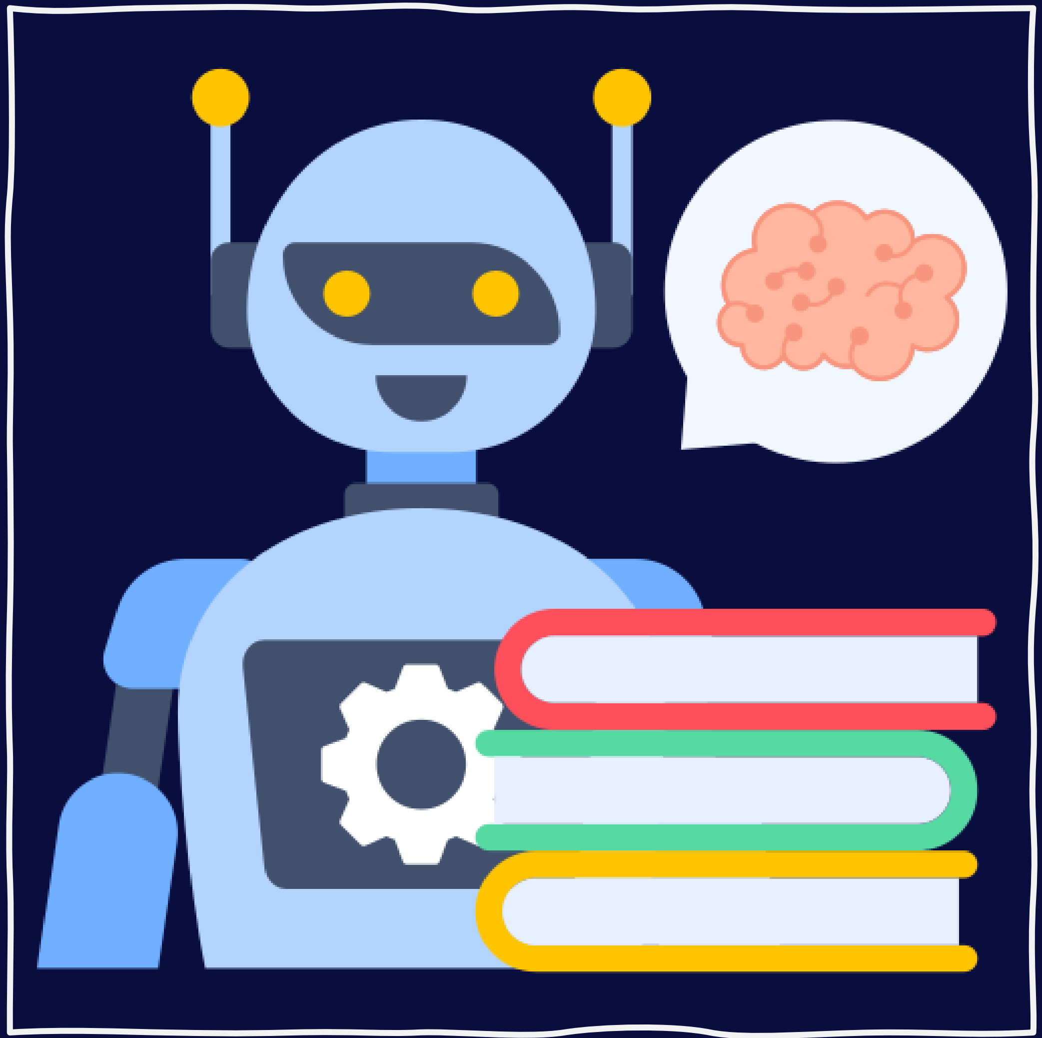
Why EfficientNetB3 is the Right Choice ?

- ❑ Powerful: It is much better at seeing the small, detailed differences in images (which MobileNet couldn't do).
- ❑ Smart Design: It is both strong and efficient, giving us high accuracy without needing a super powerful training machine.
- ❑ Key Takeaway: For complex image tasks, we need a model with high feature power, not just one designed for speed.

Strategic Data Augmentation

To combat overfitting and improve generalization, we implemented advanced data augmentation techniques:

- ❑ Random rotations and flips.
- ❑ Color jittering and contrast adjustments.
- ❑ Applying various filters to simulate real-world
- ❑ conditions.



Results Summary: Fine-Tuning for Precision

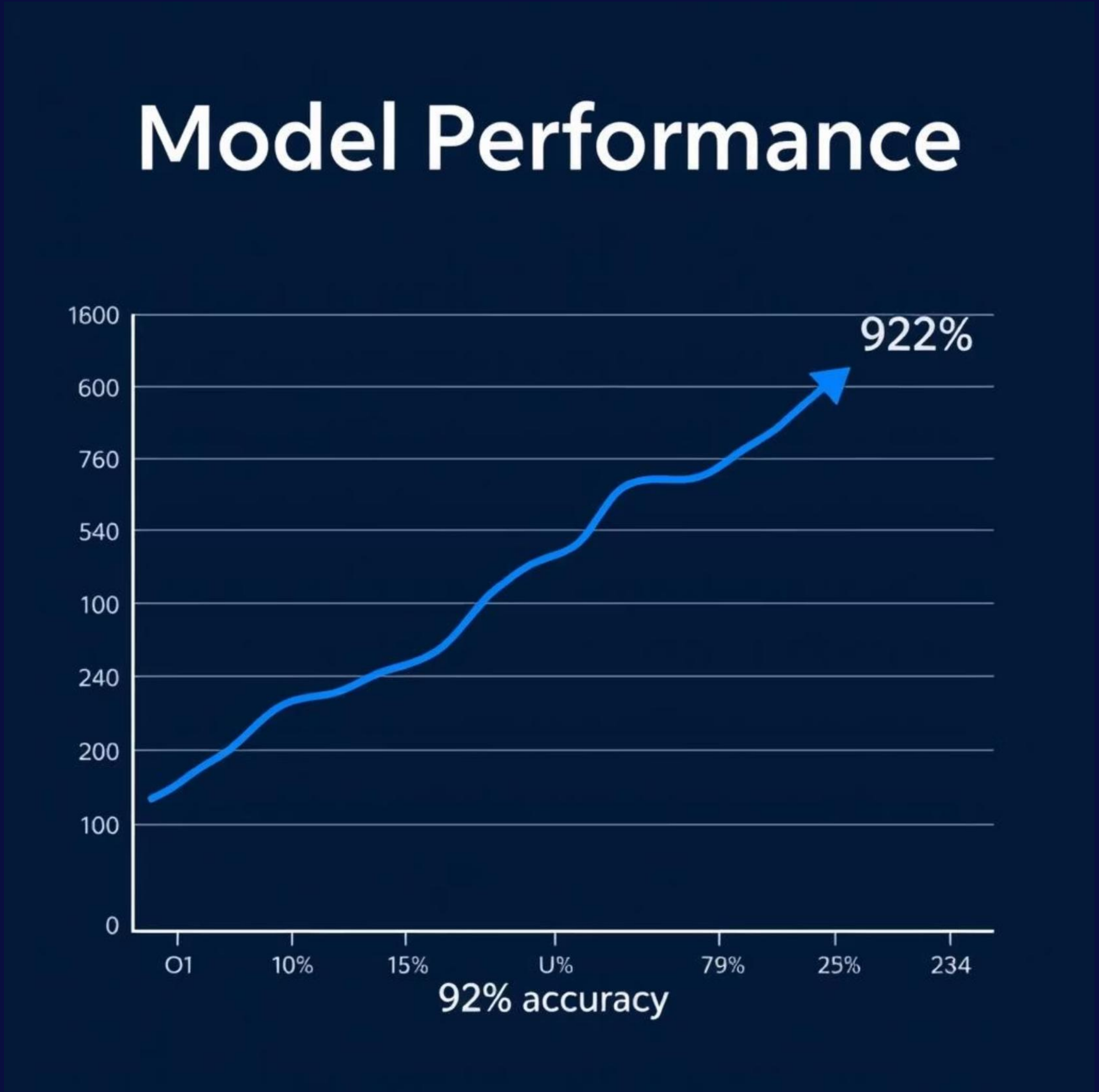
Current Model Accuracy: 82%

After rigorous training and fine-tuning, our model achieved an impressive 82% accuracy on the validation set. This represents a substantial improvement over the initial MobileNet approach.

Fine-Tuning Process

Achieving this level of precision involved:

- **Hyperparameter optimization: Adjusting learning rates, batch sizes, and regularization.**
- **Transfer learning: Leveraging pre-trained weights to speed up convergence.**
- **Iterative validation: Continuously testing and refining the model's performance.**



Key Lessons Learned & Future Directions

Importance of Data Quality

Clean, balanced, and diverse data is paramount. High-quality data augmentation proved critical in enhancing model robustness and preventing overfitting.

Iterative Experimentation

ML projects thrive on continuous experimentation. Don't changing models or approaches is a very good way to learn when initial results are unsatisfactory.

Resource Management

Adequate computational resources (especially GPUs) are vital for efficient development and iteration. Plan for infrastructure needs early.