FLOWER RECOGNITION USING DEEP-LEARNING

Machine Learning

Insaf CHAIBI & Rania BEN GHARSSALLAH

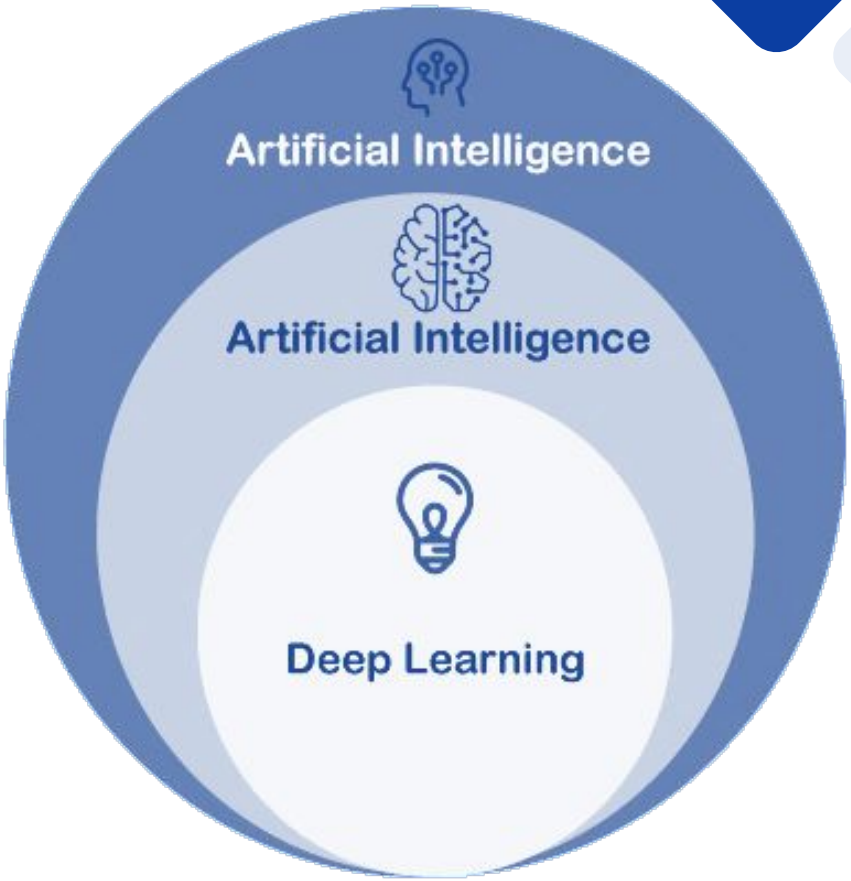
CONTENT

01

What is Deep-Learning?

What is Deep-Learning?

- Deep-Learning is a type of machine learning based on artificial neural networks in which multiple layers of processing are used to extract progressively higher level features from data.
- Deep learning models can recognize complex patterns in pictures, text, sounds, and other data to produce accurate insights and predictions.

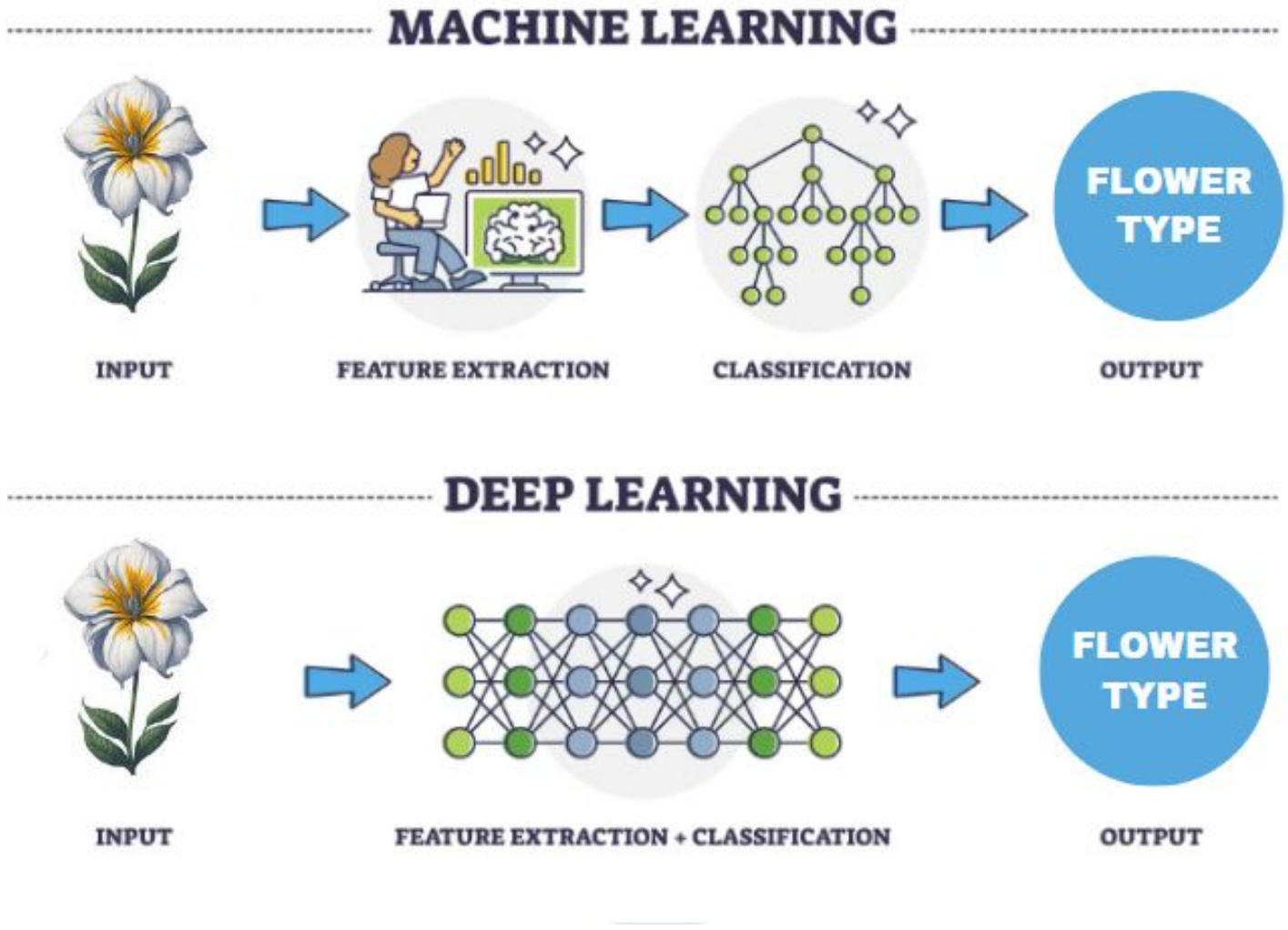


Deep-Learning Algorithms

02

ML Vs DL

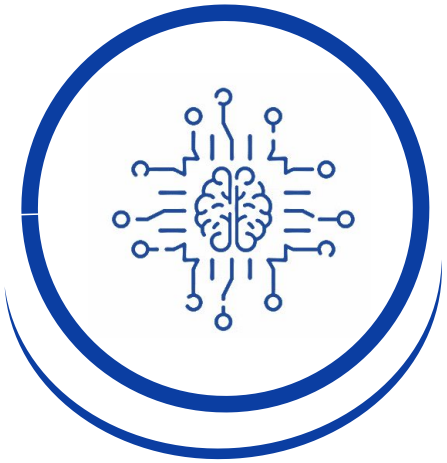
ML Vs DL


MACHINE LEARNING
FLOWER TYPE
INPUT
FEATURE EXTRACTION
CLASSIFICATION
OUTPUT
DEEP LEARNING
FLOWER TYPE
INPUT
FEATURE EXTRACTION + CLASSIFICATION
OUTPUT

DL Libraries or Frameworks

- **TensorFlow**: Developed by Google, TensorFlow is a versatile deep learning library known for its scalability, flexibility, and comprehensive ecosystem. It offers both high-level APIs, such as Keras for easy model development, and low-level APIs for fine-grained control over model architecture and training process.
- **PyTorch**: Created by Facebook's AI Research lab, PyTorch is a dynamic deep learning framework favored by researchers and developers for its intuitive interface and Pythonic syntax. PyTorch employs dynamic computation graphs, allowing for more flexibility in model construction and debugging. Its simplicity and ease of use make it ideal for rapid experimentation and prototyping of novel deep learning architectures.

Learning Types



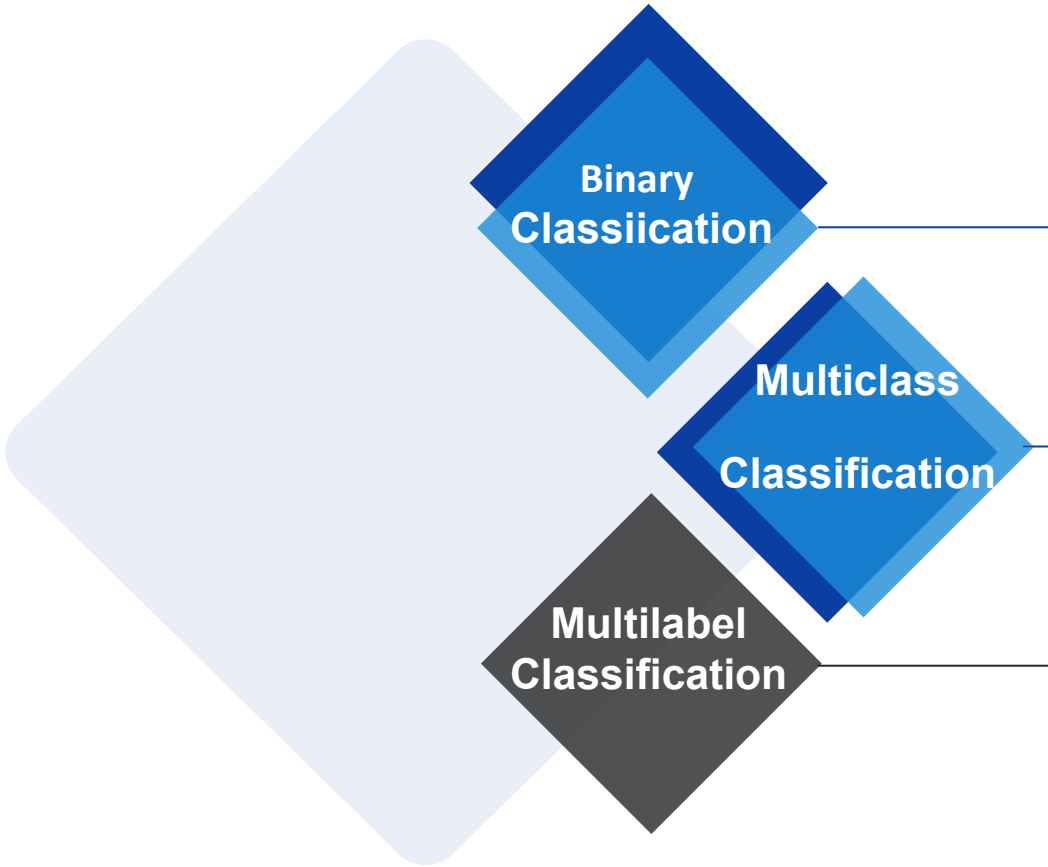
Supervised Learning

Unsupervised Learning

Semi-supervised Learning

Reinforcement Learning

Classification Types



Binary Classification

Binary classifiers can only classify data into two categories.

Multiclass Classification

Multiclass classifiers can classify data into more than two categories.

Multilabel Classification

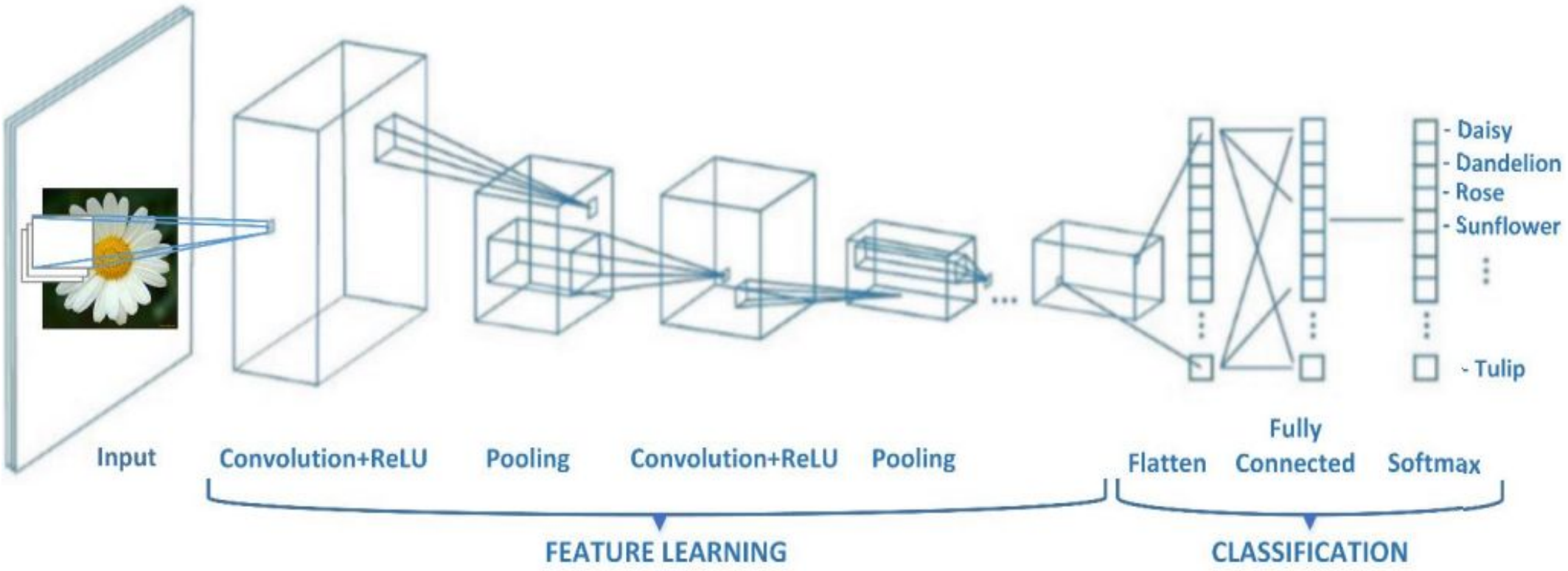
Multilabel classifiers assign or tag the data to zero or more categories.

03 CNN and Transfer Learning Architecture

CNN Architecture

- A Convolutional Neural Network (CNN) is a type of deep learning algorithm primarily used for image recognition and processing tasks. It is inspired by the human visual system and consists of multiple layers of neurons that analyze visual data. CNNs are adept at automatically learning features from raw input data, making them highly effective in tasks such as image classification, object detection, and facial recognition.

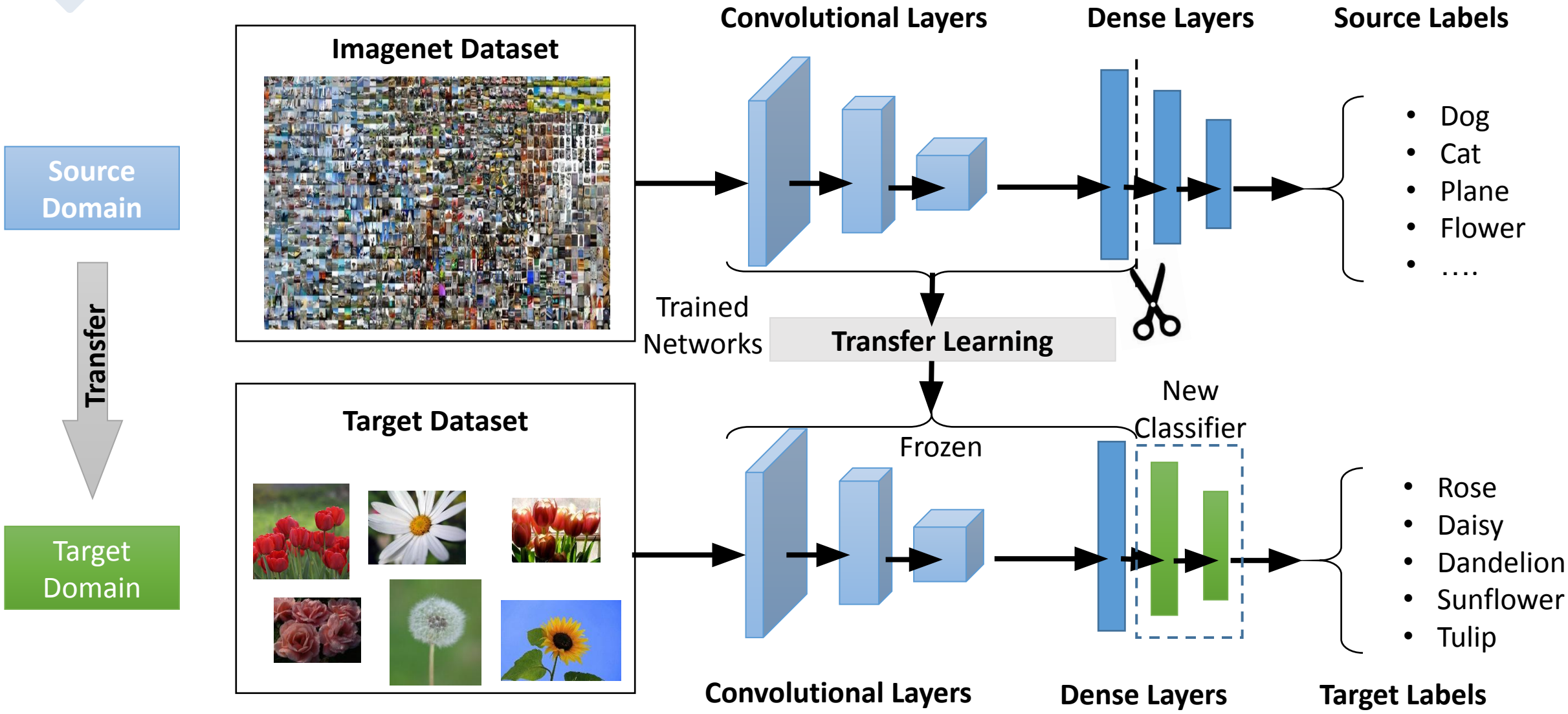
CNN Architecture



Transfer Learning Architecture

- Transfer learning, is the reuse of a pre-trained model on a new problem. In transfer learning, a machine exploits the knowledge gained from a previous task to improve generalization about another. For example, in training a classifier to predict whether an image contains plants, you could use the knowledge it gained during training to recognize flowers.

Transfer Learning Architecture



Transfer Learning's Advantages

- Reducing training time,
- Improving performance with limited data,
- Aiding generalization to new tasks,
- Facilitating domain adaptation,
- Avoiding overfitting.

04

Data-mining Life Cycle

Data-mining Life Cycle

Data collection → Data preprocessing → Models learning → Evaluation → Integration

5 classes of flower types

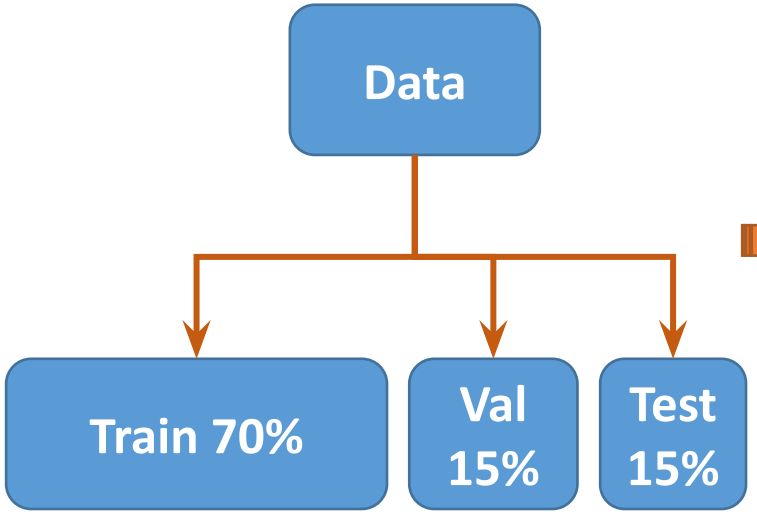
- Daisy
- Dandelion
- Rose
- Sunflower
- Tulip





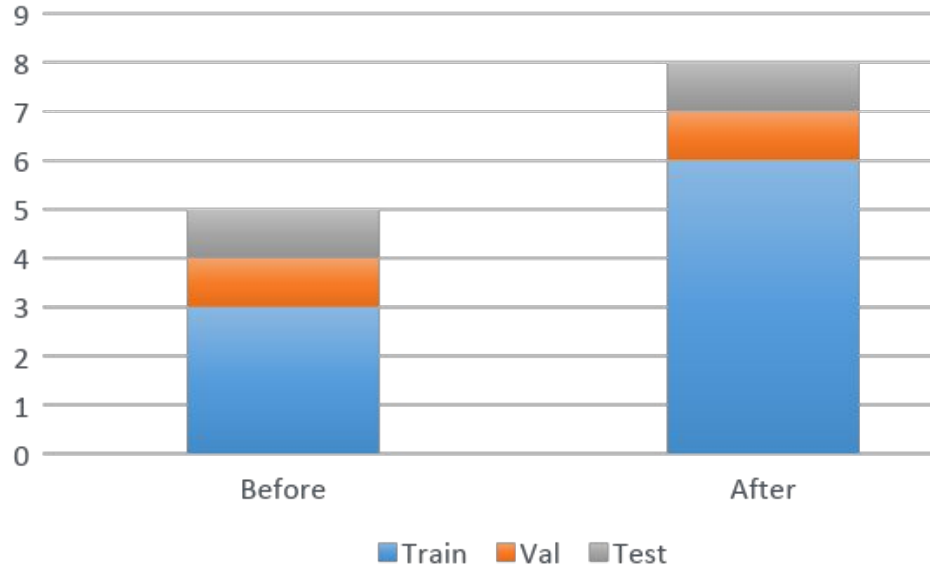
Data-mining Life Cycle

Data collection → Data preprocessing → Models learning → Evaluation → Integration



Data splitting

Data augmentation



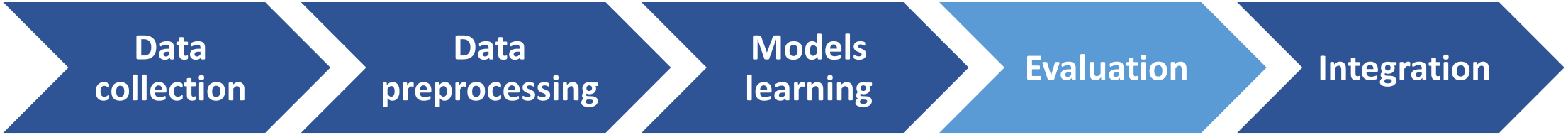
Data-mining Life Cycle

Data collection → Data preprocessing → Models learning → Evaluation → Integration

CNN + Transfer learning

- InceptionV3
- VGG16
- ResNet50

Data-mining Life Cycle



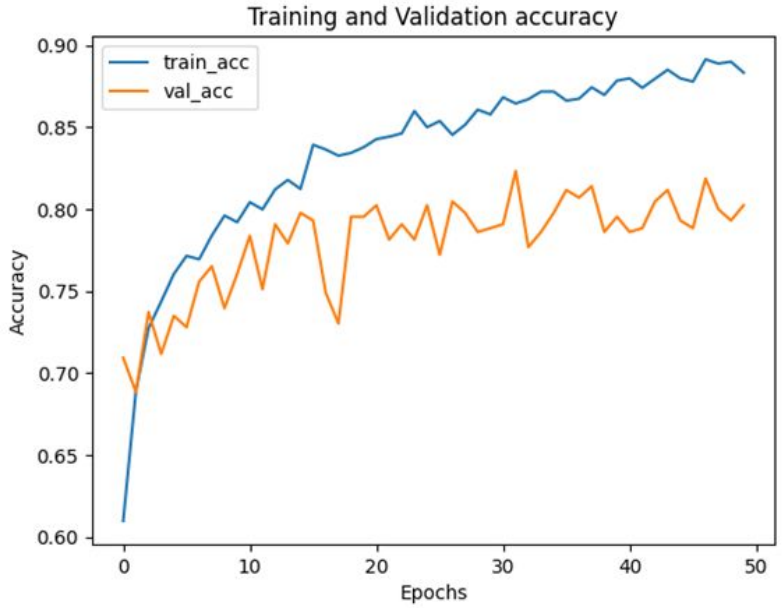
Comparison between CNN and Transfer Learning models (Macro AVG)

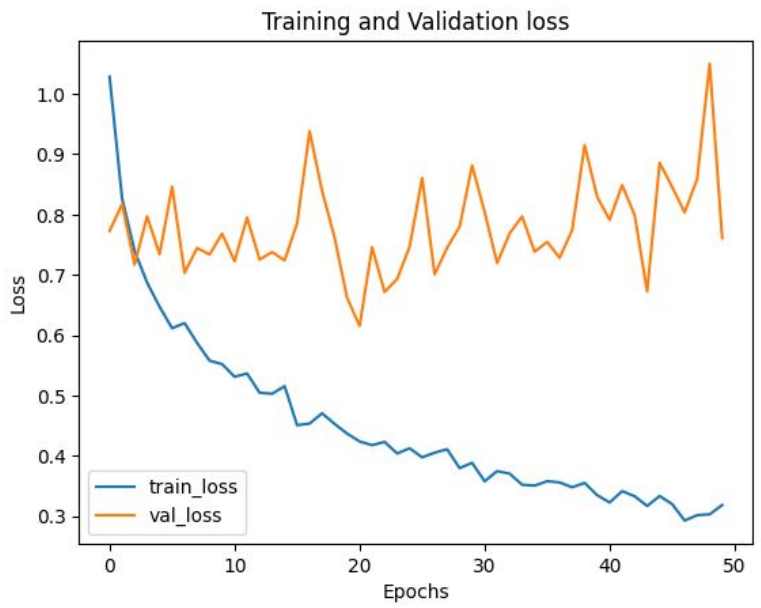
	Accuracy	Precision	Recall	F1-score	Loss
CNN	0.78	0.79	0.78	0.78	1.59
InceptionV3	0.62	0.64	0.61	0.60	1.18
VGG16	0.69	0.69	0.68	0.68	1.13
ResNet50	0.88	0.87	0.86	0.88	0.3

Data-mining Life Cycle

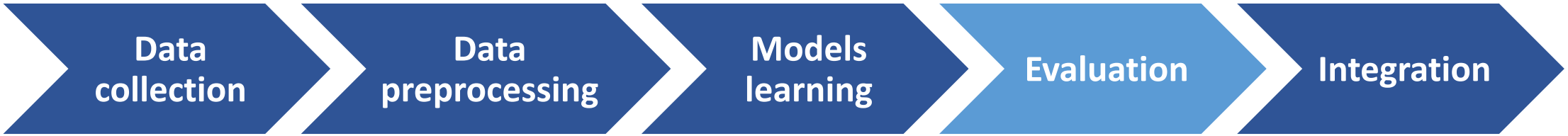
Data collection → Data preprocessing → Models learning → Evaluation → Integration

Overfitting





Data-mining Life Cycle



Overfitting solutions

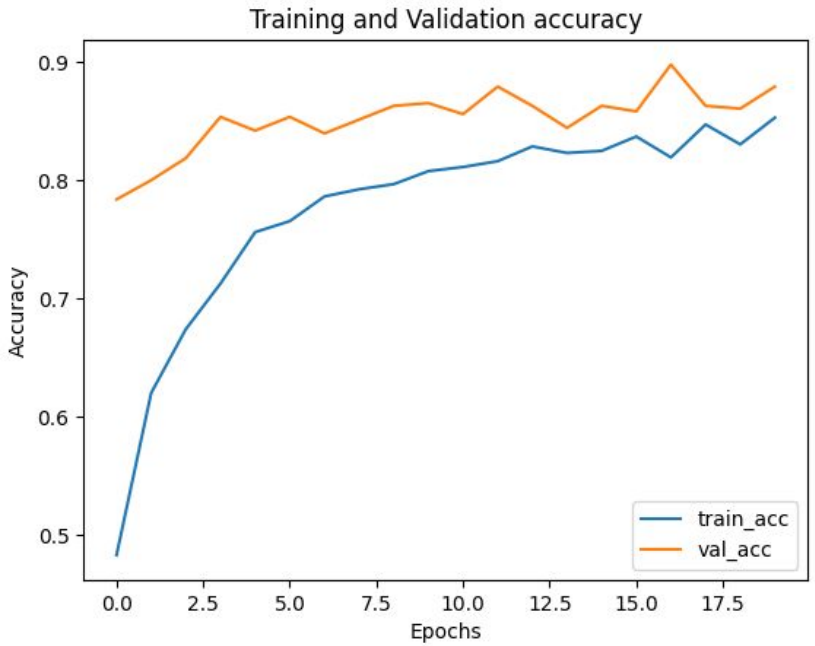
- Data Augmentation
- Regularization:

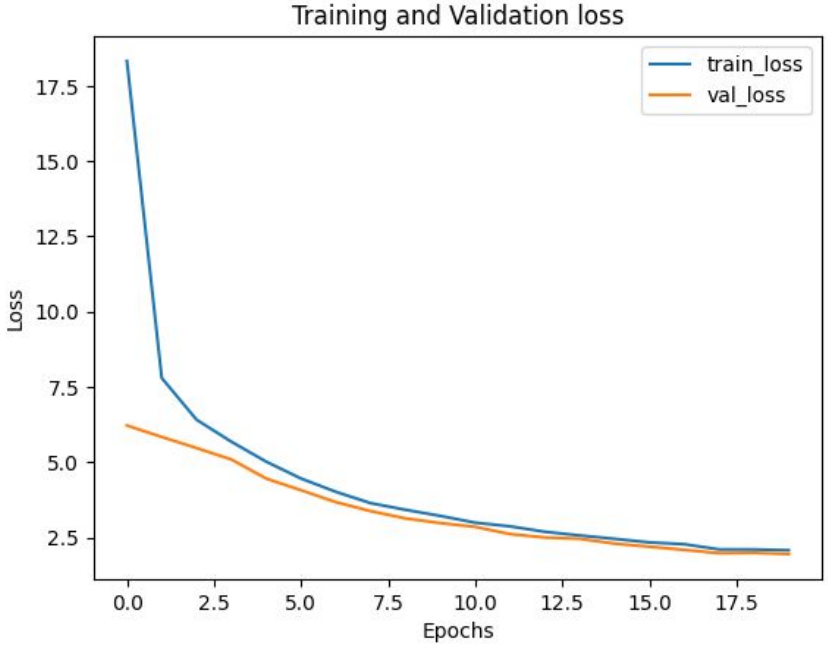
L1 and L2 Regularization

- Dropout
- Early Stopping
- Simplifying the Model
- Cross-Validation
- Transfer learning

Data-mining Life Cycle

Data collection → Data preprocessing → Models learning → Evaluation → Integration





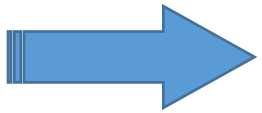
Data-mining Life Cycle

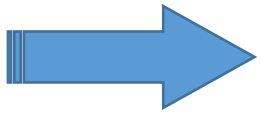
Data collection → Data preprocessing → Models learning → Evaluation → Integration

Flask

django

FastAPI





Predicted Result

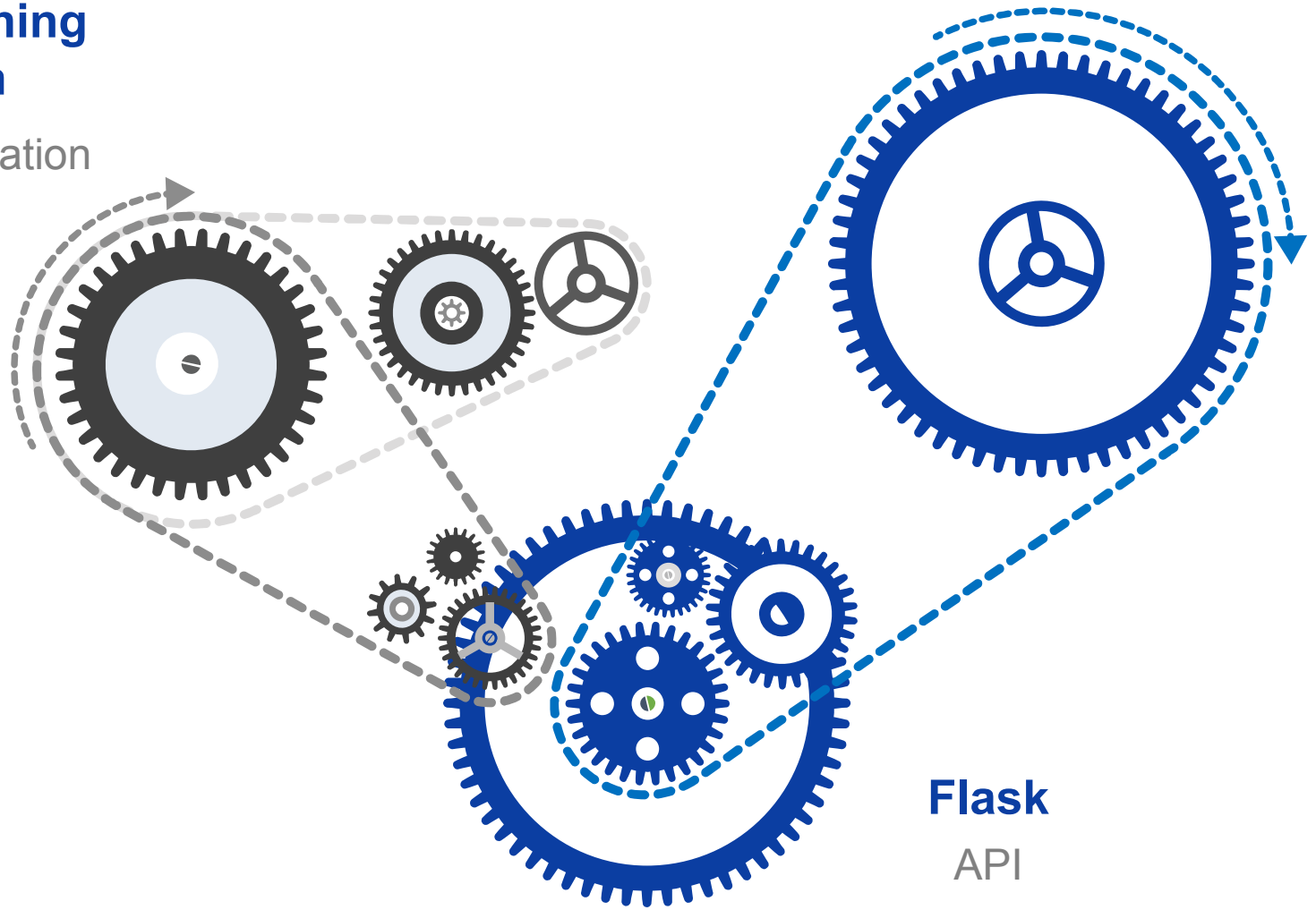
=

Flower Type

System Architecture

Deep Learning System

Flower Classification
Python



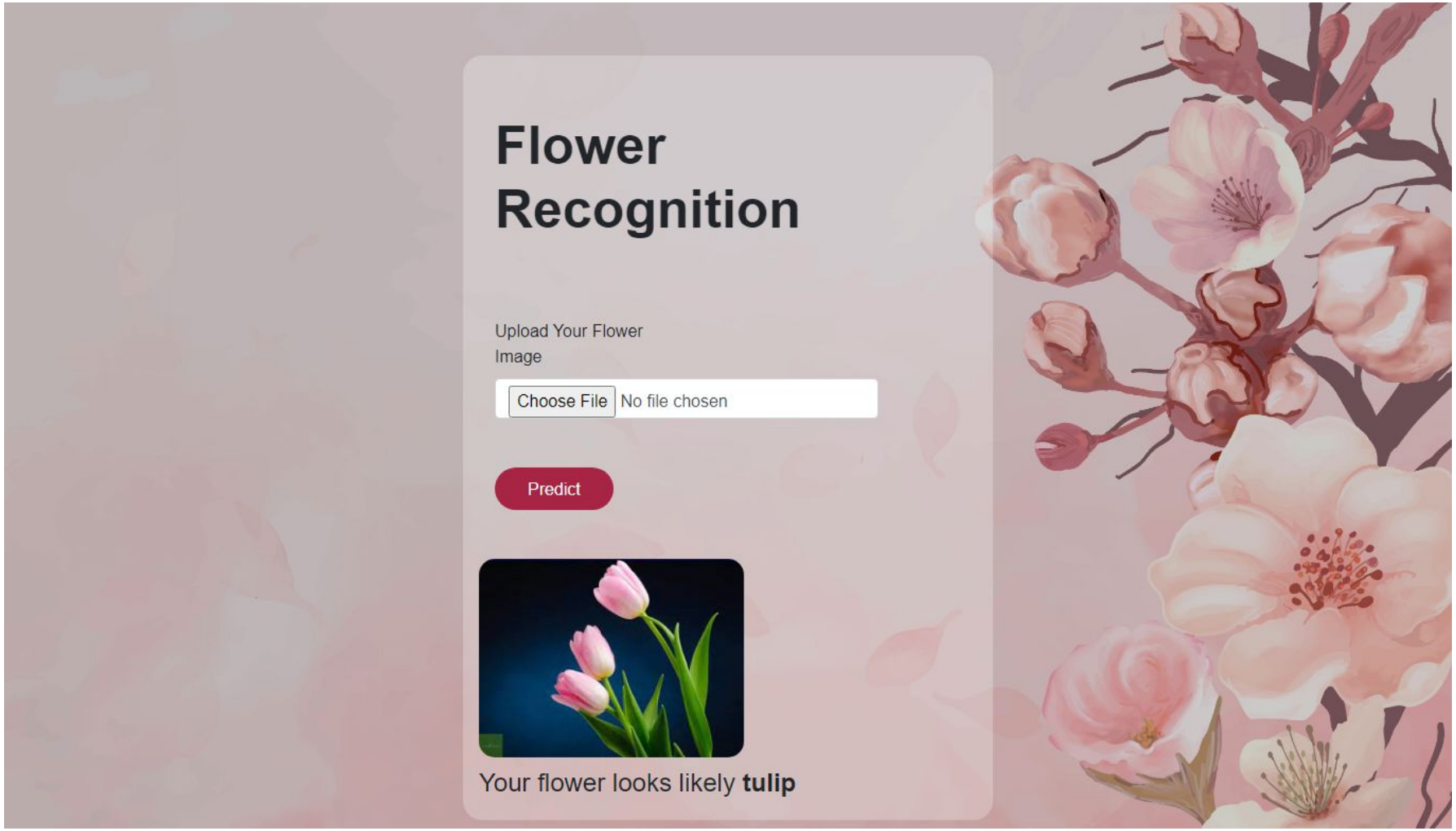
Web Application

HTML & CSS

Flask

API

UI Screenshot



References

- https://www.kaggle.com/datasets/alxmamaev/flowers-recognition/data
- https://www.tensorflow.org/tutorials/images/cnn?hl=fr
- https://builtin.com/data-science/transfer-learning
- https://www.v7labs.com/blog/overfitting-vs-underfitting#:~:text=It%20is%20different%20from%20overfitting,the%20input%20and%20the%20output.
- https://www.simplilearn.com/tutorials/artificial-intelligence-tutorial/ai-vs-machine-learning-vs-deep-learning#:~:text=The%20major%20difference%20between%20deep,layers%20of%20artificial%20neural%20networks.

THANK YOU FOR YOUR ATTENTION

DO YOU HAVE ANY QUESTIONS