Dipping Toes in Deep Learning

Hands-on implementation tiga fundamental architectures: FCN (autoencoder), CNN CIFAR 10 , RNN (time series). Setiap model diimplementasikan dari scratch dengan proper data pipeline dan evaluation metrics. [1]

Fully Connected Networks: Autoencoder Implementation

Denoising Autoencoder untuk MNIST

```
encoder = tf.keras.Sequential([
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.Dense(32, activation='relu')  # Latent space
])

decoder = tf.keras.Sequential([
    tf.keras.layers.Dense(128, activation='relu'),
    tf.keras.layers.Dense(784, activation='sigmoid')
])

autoencoder = tf.keras.Model(noisy_input, decoder(encoder(noisy_input)))
autoencoder.compile(loss='mse', optimizer='adam')
```

Architecture details: Symmetric encoder-decoder, bottleneck latent space 32D , MSE reconstruction loss.

Convolutional Neural Networks: CIFAR 10 Classifier

Modern CNN architecture:

```
model = tf.keras.Sequential([
    tf.keras.layers.Conv2D(32, 3, activation='relu', input_shape=(32, 32, 3)),
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Conv2D(64, 3, activation='relu'),
    tf.keras.layers.MaxPool2D(),
    tf.keras.layers.GlobalAveragePooling2D(),
    tf.keras.layers.Dense(10, activation='softmax')
])
```

Key improvements: BatchNorm, GlobalAvgPool2D (vs Flatten), data augmentation.

Recurrent Neural Networks: CO2 Time Series

GRU untuk multivariate time series:

```
model = tf.keras.Sequential([
    tf.keras.layers.LSTM(64, return_sequences=True, input_shape=(window_size, n_features)
    tf.keras.layers.Dropout(0.2),
    tf.keras.layers.LSTM(32),
    tf.keras.layers.Dense(1)
])

# Windowing function
def make_windows(data, window_size):
    windows = []
    for i in range(len(data) - window_size):
        windows.append(data[i:i+window_size])
    return np.array(windows)
```

Training strategy: Rolling window prediction, MAE loss untuk regression.

Kesimpulan

Chapter 4 memberikan solid foundation praktis untuk tiga core deep learning patterns sebelum menuju state-of-the-art models. [1]

Ringkasan dilanjutka n untuk chapters 5 15 dengan format serupa dalam response berikutnya jika diperlukan. Setiap bab diperluas 4x dengan code examples, technical details, best practices, dan production considerations.) [1]