Artificial Intelligence and Knowledge Engineering
Assignment 5

Piotr Syga

May 31, 2025

1 Excercise goal

The aim of this assignment is to gain practical experience with training and using a multilayer perceptron model in classification tasks. As an example, the classification task will involve biometric identity verification based on facial images.

2 Theoretical Background

Neural networks are among the most important classes of systems used in machine learning. They are inspired by the structure and function of biological neural systems and their mechanisms of learning. The fundamental computational unit in any artificial neural network is the neuron, which processes input signals using an activation function, introducing nonlinearity into the transformation, and forwards the result to subsequent layers of the model. Neurons are organized into layers: an input layer, one or more hidden layers, and an output layer. The interconnections between them are parameterized by weights, which are modified during the training process.

In the context of this assignment, we focus on the multilayer perceptron (MLP) model. This is a feedforward neural network in which data flows from the input layer to the output layer without feedback loops. This model is applicable to both classification (which is our focus here) and regression tasks, depending on the nature of the output variable and the selected loss function.

MLP training is typically performed using the backpropagation algorithm in a supervised learning regime, i.e., using labeled training data. During this iterative process, the network weights are updated to minimize the prediction error with respect to a specified loss function. Common choices include the mean squared error for regression and cross-entropy loss for classification. The training process relies on gradient descent, with each optimization step updating the weights using:

$$\theta_{t+1} = \theta_t - \eta \cdot \nabla_\theta J(\theta) \ ,$$

where θ is the parameter vector of the network, η is the learning rate, and $J(\theta)$ is the loss function.

One of the key aspects of neural network design is the proper preparation of input data. Beyond ensuring label availability, input data must be standardized and quality-controlled. For image processing, a fundamental design decision is the dimensionality of the input: whether images are monochromatic (e.g., binary masks or grayscale) or multi-channel (e.g., three-channel RGB or YCbCr, or more rarely, four-channel). Data quality significantly impacts the learned model weights; thus, preprocessing may be necessary to mitigate the effects of low-quality images (e.g., blurred or noisy inputs). For more advanced use cases, low-quality data may be intentionally included or synthetically degraded (cf. data augmentation) during training, to improve the model's robustness to noise and artifacts at inference time. It is also important to ensure variability in training images that reflects the deployment environment, and to restrict model input to the relevant region of interest (ROI) extracted from the full image.

The selection of model architecture, including the number and size of hidden layers, along with hyperparameters (such as learning rate, regularization strength, and activation function type) plays a crucial role in the effectiveness of the training process and the model's generalization ability. To assess model performance, data should be divided into training, validation, and test sets, which helps detect underfitting and overfitting and supports optimal hyperparameter selection.

The face comparison problem (or identity verification based on facial images) is one of the fundamental topics in biometrics and pattern recognition. The goal of the task (in a simplified form) is to determine whether two given images depict the same individual. Formally, this is a binary classification problem, where for each image pair (I_1, I_2) a label $y \in \{0, 1\}$ must be assigned, with $y = 1$ indicating identity match.

Modern approaches to face recognition rely on deep neural networks trained using supervised or self-supervised learning. A dominant methodology involves deep models that map input images to a latent feature space (embedding space), where faces of the same individual are represented by vectors that are close according to a selected distance metric, such as Euclidean or cosine distance.

Typical architectures used in this context include Siamese networks and triplet networks, which learn similarity functions rather than direct classifiers. In the Siamese network case, the model takes image pairs (I_1, I_2) as input and minimizes the contrastive loss:

$$\mathcal{L}_{\text{contrastive}} = y \cdot \|f(I_1) - f(I_2)\|^2 + (1 - y) \cdot \max(0, m - \|f(I_1) - f(I_2)\|)^2 \ ,$$

where $f(\cdot)$ is the feature extraction function implemented as a deep convolutional network, and m is a margin that separates different classes. Alternatively, a triplet loss can be used, which operates on triplets of images: (anchor, positive, negative).

Once training is complete, comparing two images is reduced to computing the distance between their respective embedding vectors. A decision threshold separates the “same person” and “different person” classes, and it may be determined empirically using a validation set.

The effectiveness of face recognition systems depends on several factors, including input data quality, lighting conditions, face orientation, and the presence of occlusions (e.g., masks or glasses). High-performance models (e.g., FaceNet, ArcFace, AdaFace) can achieve human-level or superhuman accuracy on benchmark datasets (e.g., CelebA, LFW, MegaFace). More compact models are suitable in scenarios with limited computational resources or restricted training data availability. This assignment focuses on evaluating an MLP-based classifier for the identity verification task.

3 Tasks

Using the provided source code file and the CelebA dataset (or the streaming version), complete the following face verification tasks:

1. Train an MLP model on a random subset of 10, 100, 500, 1000, and 5000 image pairs, and evaluate the identity verification performance on a disjoint subset of 200 image pairs (ensuring no training images are reused for evaluation). Using the metrics introduced in Assignment 4, analyze the impact of training set size on classification performance (40 points).

2. Investigate the influence of the learning rate (`lr`) on model performance, using a fixed training set size (1000 pairs) and a fixed number of epochs. Analyze at least 5 different learning rate values (20 points).

3. Investigate the influence of the number of training epochs on model performance, using a fixed training set size (1000 pairs) and a fixed learning rate. Analyze at least 5 different epoch counts (20 points).

4. Based on the experiments, examine the relationship between training set size, learning rate, and number of epochs. Determine the best-performing parameter set and consider strategies such as adaptive learning rate schedules or early stopping based on changes in the loss function (20 points).

5. During evaluation, apply additional perturbations to the test images (e.g., Gaussian noise, blurring, brightness adjustment) and compare the identity verification performance between perturbed and unperturbed data. Modify the system to mitigate the impact of such perturbations (e.g., by augmenting the training data, modifying the network architecture, changing the loss function, or applying preprocessing techniques) (up to 25 bonus points).

For each task, prepare a report that includes a theoretical description of the method, hyperparameter settings, implemented modifications, results and interpretation, supplementary materials, and brief descriptions of the libraries used in implementation. In the report summary, describe any implementation challenges encountered during the assignment.

Submit the report to the instructor at least 24 hours before the assignment deadline.

4 References

- Introduction to MLP
- Introduction to PyTorch
- FaceNet
- M. Kim et al., AdaFace: Quality Adaptive Margin for Face Recognition
- Albumentations