

МІНІСТЕРСТВО ОСВІТИ І НАУКИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»
Кафедра систем штучного інтелекту



Лабораторна робота № 3
З курсу «Обробка зображень методами штучного інтелекту»

Виконав:
ст.гр. КН-408
Петров Кирило

Перевірив:
Пелешко Д.

Львів – 2022

Класифікація зображень. Застосування нейромереж для пошуку подібних зображень.

Мета: набути практичних навиків у розв’язанні задачі пошуку подібних зображень на прикладі організації CNN класифікації.

Завдання

12. Побудувати CNN на основі **LeNet-5** для класифікації зображень на основі датасету fashion-mnist. Зробити налаштування моделі для досягнення необхідної точності. На базі Siamese networks побудувати систему для пошуку подібних зображень в датасеті fashion-mnist. Візуалізувати отримані результати t-SNE.

Виконання

Нейронна мережа **LeNet-5** складається з 5 шарів: перші 3 – згорткові, а останні 2 – лінійні.

```
keras.layers.Conv2D(6, kernel_size=5, strides=1, activation='tanh', input_shape=x_train[0].shape, padding='same'), #C1
keras.layers.AveragePooling2D(), #S2
keras.layers.Conv2D(16, kernel_size=5, strides=1, activation='tanh', padding='valid'), #C3
keras.layers.AveragePooling2D(), #S4
keras.layers.Flatten(), #Flatten
keras.layers.Dense(120, activation='tanh'), #C5
keras.layers.Dense(84, activation='tanh'), #F6
keras.layers.Dense(10, activation='softmax') #Output layer
])
```

Далі будемо сіамські мережі, вони влаштовані так, що беруться дві LeNet-5 мережі з окремими входами, спільним виходом та вагами. На виході кожна мережа продукує вектори передбачень, які можна порівнювати між собою для визначення їх схожості.

```
img_a_in = Input(shape=x_train.shape[1:], name='ImageA_Input')
img_b_in = Input(shape=x_train.shape[1:], name='ImageB_Input')
img_a_feat = lenet_5(img_a_in)
img_b_feat = lenet_5(img_b_in)
combined_features = concatenate([img_a_feat, img_b_feat], name='merge_features')
combined_features = Dense(16, activation='linear')(combined_features)
```

```

combined_features = BatchNormalization()(combined_features)
combined_features = Activation('relu')(combined_features)
combined_features = Dense(4, activation='linear')(combined_features)
combined_features = BatchNormalization()(combined_features)
combined_features = Activation('relu')(combined_features)
combined_features = Dense(1, activation='sigmoid')(combined_features)
similarity_model = Model(inputs=[img_a_in, img_b_in], outputs=[combined_features], name='Similarity_Model')

```

На початку випадково ініціалізовані мережі не дали результату:

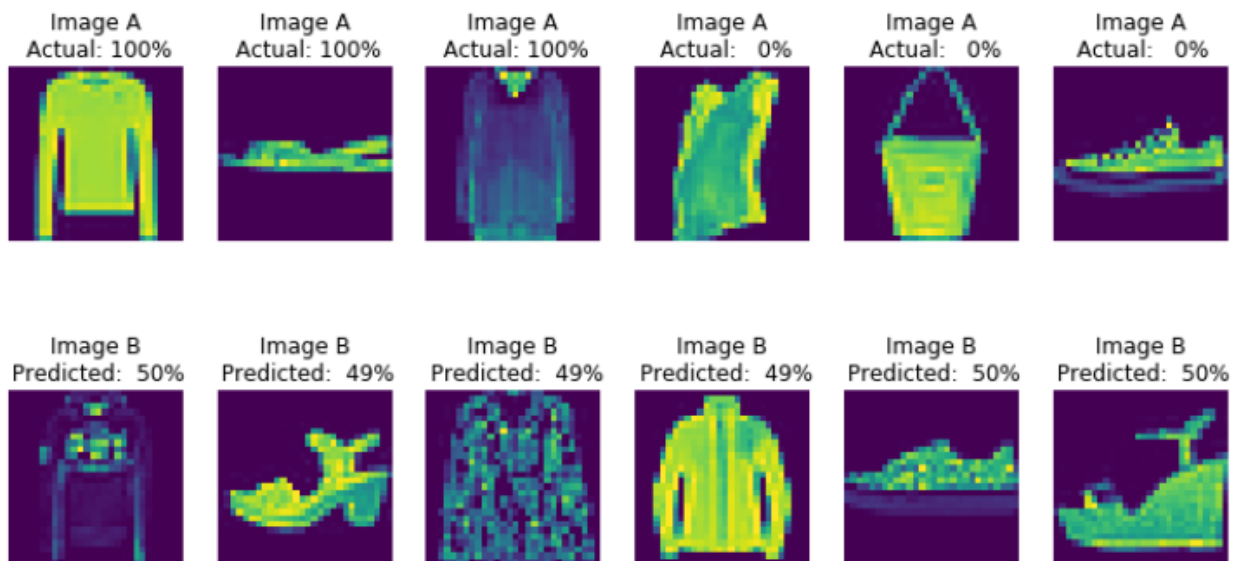
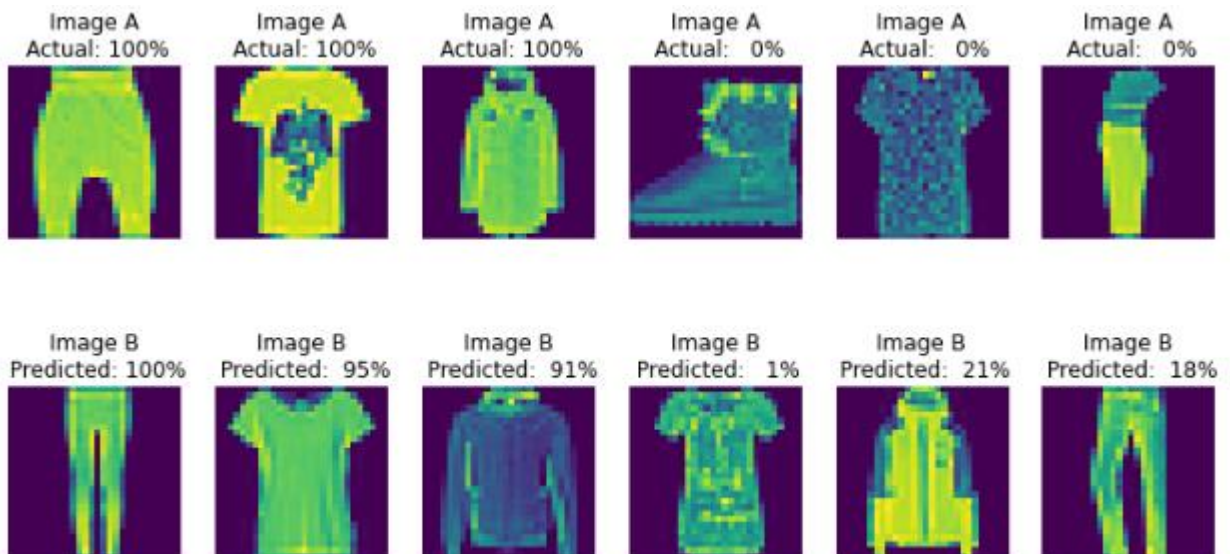


Рис. 1. Результати передбачень випадково ініціалізованих сіамських мереж.

Після тренування протягом 25 епох було отримано значно кращі результати:



Використання t-SNE показало, що сіамська мережа добре виокремила ознаки штанів, кросівок, сумок, чобіт, сандалей, кросівок (які є візуально відрізними та легшими для розпізнання) та гірше футболки, пуловерів, суконь, плащів, сорочок (які візуально схожі між собою).

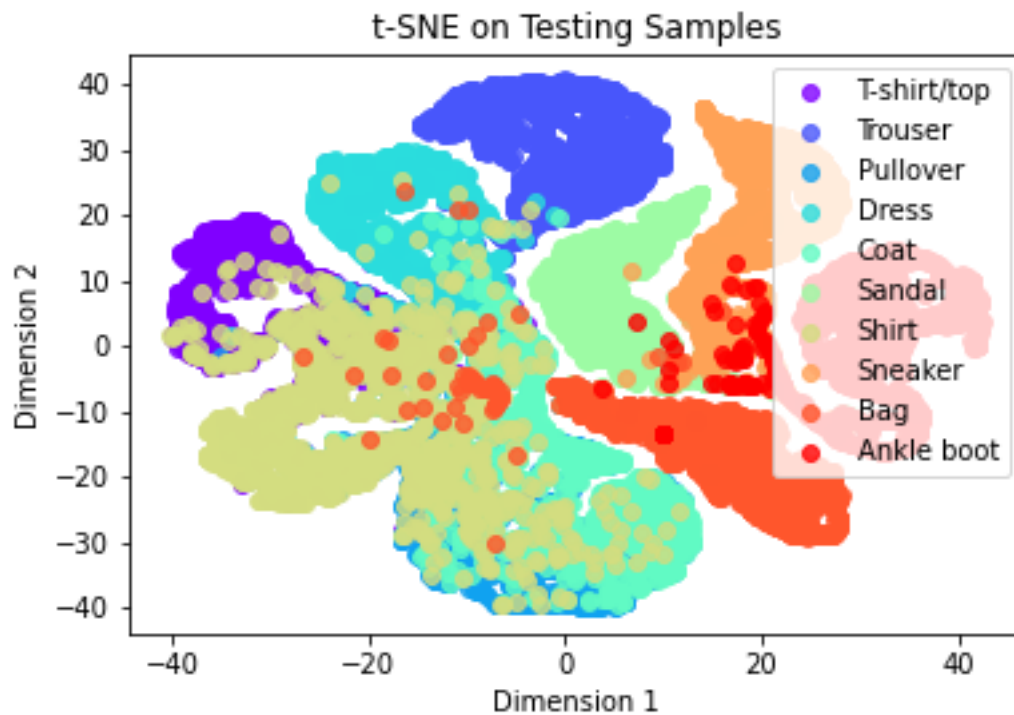


Рис. 2. Результати роботи мереж створені за допомогою t-SNE.

Висновок: у ході роботи я набув практичних навиків у розв’язанні задачі пошуку подібних зображень на прикладі організації CNN класифікації.