

Rapport sur le Projet d'IA en Cours

Groupe: Nicolas Riedel / El Hadji Silly Keita / Esteban Dubois

Introduction

Le projet en cours vise à développer un modèle d'intelligence artificielle capable de différencier un chou-fleur d'un tournesol. L'objectif principal est d'utiliser les technologies d'apprentissage automatique, en particulier les bibliothèques Keras et TensorFlow en Python, pour former un modèle précis et efficace.

Méthodologie

Utilisation du Programme Python

Nous avons débuté en utilisant le programme Python fourni par le professeur. Initialement, nous avons rencontré des difficultés lors du lancement du code. Après une analyse approfondie et des ajustements, nous avons réussi à exécuter le code avec succès. Le programme utilise les bibliothèques Keras et TensorFlow pour la création et l'entraînement du modèle d'IA.

Entraînement du Modèle

Nous avons observé qu'il était nécessaire de réaliser environ une vingtaine d'époques pour que le modèle atteigne un niveau de précision satisfaisant. Ce processus d'entraînement est essentiel pour permettre au modèle de discerner de manière fiable les caractéristiques distinctives entre un chou-fleur et un tournesol.

```
from keras.optimizers import Adam

# Compiler le modèle
model_mobilenet.compile(optimizer=Adam(), loss='categorical_crossentropy',
metrics=['accuracy'])

# Entraîner le modèle
history = model_mobilenet.fit(
    X_train, y_train,
    epochs=10,
    batch_size=32,
    validation_data=(X_val, y_val)
)

# Évaluer le modèle sur l'ensemble de test
test_loss, test_accuracy = model_mobilenet.evaluate(X_test, y_test)
print(f"Test Loss: {test_loss:.4f}, Test Accuracy: {test_accuracy:.4f}")
```

Développement de l'Application Java

Suite à la création du modèle, nous avons entrepris le développement d'une application en Java. Cette application utilise le fichier de modèle (format pb) généré par l'entraînement du modèle en Python. L'objectif est de fournir une interface utilisateur permettant à l'utilisateur final de soumettre des images à analyser.

L'application Java a été conçue pour effectuer une analyse précise, exploitant les capacités du modèle IA préalablement entraîné. Les résultats obtenus ont démontré une validation d'environ 80%, indiquant une performance satisfaisante de l'application dans la distinction entre chou-fleur et tournesol.

```
URL modelUrl = ProjectIA.class.getResource("/mobilenet_with-preprocessing.pb");
String modelPath = new File(modelUrl.getPath()).getAbsolutePath();
Net net = Dnn.readNetFromTensorflow(modelPath);

// Lire et afficher l'image sélectionnée
ImageIcon originalIcon = new ImageIcon(selectedFile.getAbsolutePath());
Image originalImage = originalIcon.getImage();

int maxWidth = 700;
int maxHeight = 500;
Image scaledImage = originalImage.getScaledInstance(maxWidth, maxHeight,
Image.SCALE_SMOOTH);

ImageIcon scaledIcon = new ImageIcon(scaledImage);
imageLabel.setIcon(scaledIcon);
nameLabel.setText("Nom de l'image : " + selectedFile.getName());

// Charger et prétraiter l'image pour la comparaison
Mat inputImage = Imgcodecs.imread(selectedFile.getAbsolutePath());
Imgproc.resize(inputImage, inputImage, new Size(224, 224));
inputImage.convertTo(inputImage, CvType.CV_32F);
Scalar mean = new Scalar(127.5, 127.5, 127.5);
Core.subtract(inputImage, mean, inputImage);

// Faire la prédiction avec le modèle
Mat blob = Dnn.blobFromImage(inputImage);
net.setInput(blob);
Mat predictions = net.forward();

// Charger le fichier de labels
List<String> labels = loadLabels("/labels.txt");

// Interpréter les résultats en utilisant le fichier de labels
String predictionResult = interpretResults(predictions, labels);

// Afficher le résultat de la comparaison
resultLabel.setText("Résultat de la comparaison : " + predictionResult);
```

Conclusion

Dans l'ensemble, le projet a progressé de manière significative, surmontant les obstacles initiaux lors de l'utilisation du programme Python. L'application Java, développée pour tirer parti du modèle IA, représente un outil fonctionnel et performant. Le projet a été plutôt intéressant et novateur.