College of Professional Studies
Northeastern University San Jose

MPS Analytics

Course: ALY6020: Predictive Analytics

Assignment:

Module 5 Project - Text Classification

Submitted on:

Feb 14, 2024

Submitted to:

Prof: BEHZAD AHMADI

Submitted by:

NIKSHITA RANGANATHAN

INTRODUCTION

Understanding the dataset:

This dataset is a collection of handwritten digit images designed for the development and evaluation of machine learning models. Each entry in the dataset corresponds to an image of a handwritten digit, with the first column representing the digit's label (0 through 9) and the subsequent columns containing pixel intensity values that describe the image. These pixel columns are selectively chosen or processed from potentially larger images to reduce dimensionality and computational complexity.

The dataset consists of thousands of rows, each representing a different handwritten digit image, with 45 features per image after preprocessing

The first column (label) indicates the digit that the image represents. This is the target variable we are trying to predict with our models.

Our objective with this dataset is to predict the handwritten digits accurately, thereby exploring the application of these models in educational settings, particularly in identifying students who may need assistance with motor skills development. Then we will summarize the findings from this comparative study and recommend the most suitable model for the school.

KNN

Initially, the dataset is divided into two segments: one comprises the input features, and the other contains the target variable, known as 'label'.
Then, the dataset is further divided into training and testing sets using a 80-20 split ratio.

In the process of preparing the dataset for model training and evaluation, an essential step involves feature scaling.

Feature scaling is a technique to normalize the range of features of data. We utilized the StandardScaler from Scikit-learn to standardize the features of the handwriting images before feeding them into our machine-learning models.

A **KNN model** is constructed. The model's metrics are calculated to assess its performance in detail.

The K-Nearest Neighbors (KNN) model's performance metrics (Figure 1), including an **accuracy of 62.4%, precision of 61.5%, recall of 62.4%, and F1-score of 61.6%**, present a moderate level of effectiveness in predicting handwritten digits.

Challenges with Using KNN for Handwriting Recognition:

- The choice of k is crucial. When k is too small, the model can be sensitive to noise in the dataset. When it is too large, it might include points from other classes, reducing accuracy**.**
- When dealing with data that has many dimensions, such as images, it becomes more challenging for KNN to clearly see the differences between instances. This issue can lower the accuracy of the model.
- KNN is computationally intensive because it requires calculating the distance between a test instance and all training instances to determine the nearest neighbors. This process can be slow and use up a lot of resources, especially with large datasets.
- Handwritten digits can vary greatly in style, size, and orientation. The KNN algorithm may find it difficult to handle such diversity because it makes decisions solely based on how close the data points are to each other, lacking the capability to recognize more complex patterns.

NEURAL NETWORK

We start by converting our target variable (the actual digit labels) into a format that's easier for the neural network to understand. This process, called one-hot encoding, turns each digit label into an array where the index corresponding to the digit is marked as 1, and all other indices are 0.

The model itself is structured in layers, beginning with input and learning layers, followed by a decision-making layer for digit classification.

The training process involves feeding it digit images alongside correct labels, allowing for iterative learning and adjustment based on prediction accuracy. This cycle of prediction and adjustment, spread over several epochs, enhances the model's proficiency in recognizing and classifying unseen digit images.

Our neural network's training progresses over 10 epochs, starting with an initial training accuracy of 57.3% and a validation accuracy of 63.63%, the model shows a steady increase in its ability to classify handwritten digits accurately.

By the end of the 10th epoch, the training accuracy reaches 69.38% (Figure 2), and the validation accuracy improves to 69.00%.

Challenges with Using Neural network for Handwriting Recognition:

- A common challenge in training neural networks is overfitting, where the model performs well on training data but poorly on new, unseen data. To ensure the model is effective, it is better to implement strategies such as adding dropout layers or adjusting the complexity of the model.
- Training neural networks, especially on large datasets, can be computationally intensive and time-consuming.
- Finding the optimal configuration for the model, including the number of layers, the number of neurons in each layer, and the learning rate, was a significant challenge. These parameters significantly impact model performance, and tuning them requires extensive experimentation.
- The variability in handwriting styles was another challenge. The model had to learn to interpret these variations accurately, a task that demanded training strategies.

COMPARING MODELS

- The neural network model achieves an accuracy of 0.690, compared to the KNN model's accuracy of 0.624. This suggests the neural network is better at correctly identifying handwritten digits.
- With a precision of 0.695, the neural network model is more precise in its predictions than the KNN model, which has a precision of 0.615. This indicates the neural network has fewer false positives.
- Recall rate of the neural network at 0.690 and KNN at 0.624. This shows the neural network is slightly better at capturing all relevant instances.
- The neural network's F1 score of 0.689 is higher than the KNN's 0.616, indicating a better balance between precision and recall in the neural network model.

CONCLUSION

Based on the outcomes of both models, a neural network model outperforms a KNN model in terms of accuracy and efficiency for handwriting recognition. Thus, I would recommend the school to consider using a neural network model for identifying students who may need help with their motor skills based on their handwriting. This model is likely to provide more consistent and accurate assessments, aiding in the early identification of students who could benefit from additional support in developing their motor skills.

REFERENCES

Allibhai, J. (2022, June 21). Building a Convolutional Neural Network (CNN) in Keras. *Medium*. https://towardsdatascience.com/building-a-convolutional-neural-network-cnn-in-keras-329fbbadc5f5

Zhang, Y. (2018, May 1). Number of parameters in dense and convolutional layers in neural networks. *Medium*. https://zhang-yang.medium.com/number-of-parameters-in-dense-and-convolutional-neural-networks-34b54c2ec349

APPENDIX

```
Accuracy of KNN model: 0.624
Precision of KNN model: 0.615
Recall of KNN model: 0.624
F1-Score of KNN model: 0.616
```

Figure 1 – KNN Model

```
263/263 [==============================] - 0s 1ms/step - loss: 0.8858 - accuracy: 0.6900
Accuracy of Neural network model: 0.690
```

```
263/263 [==============================] - 0s 844us/step
Precision of Neural network model: 0.695
Recall of Neural network model: 0.690
F1 Score of Neural network model: 0.689
```

Figure 2 – Neural network Model