Faculty of Engineering and Technology

Electrical and Computer Engineering Department

Students' names and numbers	Section and Instructor
Alaa Saleem-1200001	Dr. Yazan Abu Farha
Nidal Zabade-1200153	Sec.2 - M, W 10:05 - 11:25

ENCS3340-Artificial Intelligence (AI)

"E-mail Spam Classification using k-NN and MLP Classifiers"

July 15, 2023

Table of Contents

Confusion Matrix and Compression Performance Results

In Figure 1, the confusion matrix of the k-NN and MLP classifiers provides the classification results

Note: here k-NN classifier: k=3, Euclidean distance. MLP classifier: 2 hidden layers (10 neurons, 5 neurons).

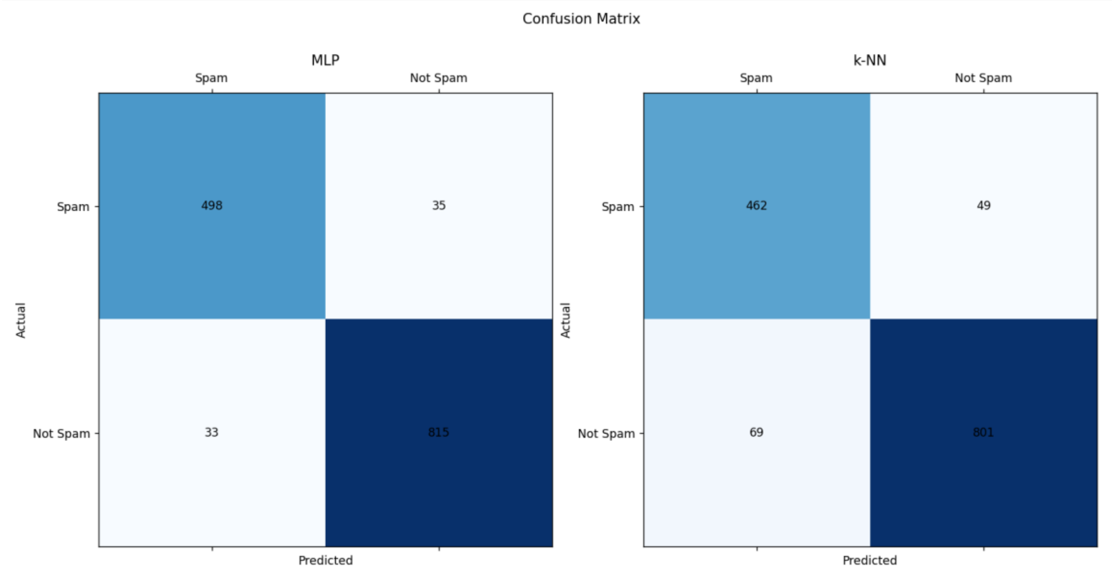

Figure 1: Confusion Matrix of k-NN and MLP classifiers

According to Figure 1 the classification results are as follows:

True Positives (TP) “Predicted positive, actually positive”

The MLP classifier correctly identified 498 emails as spam, the k-NN classifier correctly identified 462 emails as spam.

False Positive (FP) “Predicted positive, actually negative”

The MLP classifier incorrectly labeled 35 emails as spam when they were actually not spam, while the k-NN classifier incorrectly labeled 49 emails as spam when they were actually not spam.

True Negatives (TN) “Predicted negative, actually negative”

The MLP classifier accurately identified 815 emails as not spam, the k-NN classifier accurately identified 801 emails as not spam.

False Negatives (FN) “Predicted negative, actually positive”

For the MLP classifier there were 33 emails that were incorrectly classified as not spam, and 69 emails that were incorrectly classified as not spam for the k-NN classifier.

Figure 2 demonstrates the MLP classifier performance compared to the k-NN classifier in classifying spam emails. Note: here k-NN classifier: k=3, Euclidean distance. MLP classifier: 2 hidden layers (10 neurons, 5 neurons)

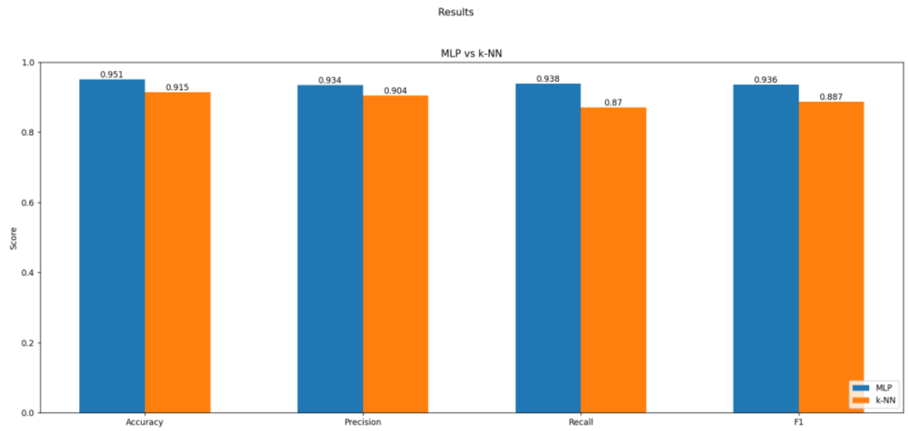

Figure 2: Performance of the k-NN and MLP classifiers

Based on the results shown in Figure 2, the performance of the MLP and k-NN classifiers in terms of accuracy, precision, recall, and F1-score can be summarized as follows:

Accuracy

$$Accuracy = \frac{\text{Total correct guesses}}{\text{Total guesses}} = \frac{TP + TN}{TP + TN + FP + FN}$$

The MLP classifier achieved an accuracy rate of 95.1%, indicating that it correctly classified 95.1% of the emails. On the other hand, the k-NN classifier achieved an accuracy rate of 91.5%, correctly classifying 91.5% of the emails.

Precision

$$Precision = \frac{\text{Correct positive guesses}}{\text{Total positive guesses}} = \frac{TP}{TP + FP}$$

The MLP classifier demonstrated a precision rate of 93.4%. This means that when the MLP classifier labeled an email as spam, it was correct 93.4% of the time. In comparison, the k-NN classifier achieved a precision rate of 90.4%, correctly labeling 90.4% of the spam emails.

Recall

$$Recall = \frac{\text{Correct positive guesses}}{\text{All positive labels}} = \frac{TP}{TP + FN}$$

The MLP classifier exhibited a recall rate of 93.8%. This indicates that the MLP classifier successfully identified a large proportion of the actual spam emails, capturing 93.8% of them. On the other hand, the k-NN classifier had a recall rate of 87%, identifying 87% of the actual spam emails.

F1-Score

$$F_1 = \frac{2 \cdot precision \cdot recall}{precision + recall} = \frac{2 \cdot TP}{2 \cdot TP + FP + FN}$$

The F1-score, which balances precision and recall, was 93.6% for the MLP classifier, while the k-NN classifier achieved an F1-score of 88.7%. The higher F1-score of the MLP classifier suggests a better balance between precision and recall compared to the k-NN classifier.

These results demonstrate that the MLP classifier outperformed the k-NN classifier in terms of accuracy, precision, recall, and F1-score. The MLP classifier achieved higher accuracy, precision, recall, and F1-score, indicating its effectiveness in accurately classifying spam emails.

Experiments

In order to enhance the performance of the k-NN classifier, additional experiments can be conducted. These experiments may involve varying the value of k, which represents the number of nearest neighbors considered during classification.

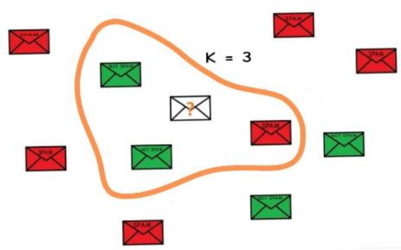

Figure 3: K-NN classifier k=3

By exploring different values of k, we can assess its impact on accuracy, precision, recall, and overall performance. Additionally, adjusting the test size, which determines the proportion of the dataset used for testing, can provide insights into the classifier's behavior under different data configurations. Through these experiments, we aim to determine if the k-NN classifier can be optimized to surpass the performance of the MLP classifier and achieve superior classification results

Increase k and decreasing the test size

Increasing K allows the k-NN classifier to consider more neighbors during classification. This can help in reducing the impact of noisy data points and improve the generalization of the classifier. (We will chose k where it is not too large to avoid over-filtering). By decreasing the test size to 0.15, you will have more training data available for both classifiers. This can potentially improve the performance of the k-NN classifier as it will have more data to learn from. (It’s important to note that reducing the test size also means we will have fewer instances for evaluation).

Example: K = 7, Test size = 0.15

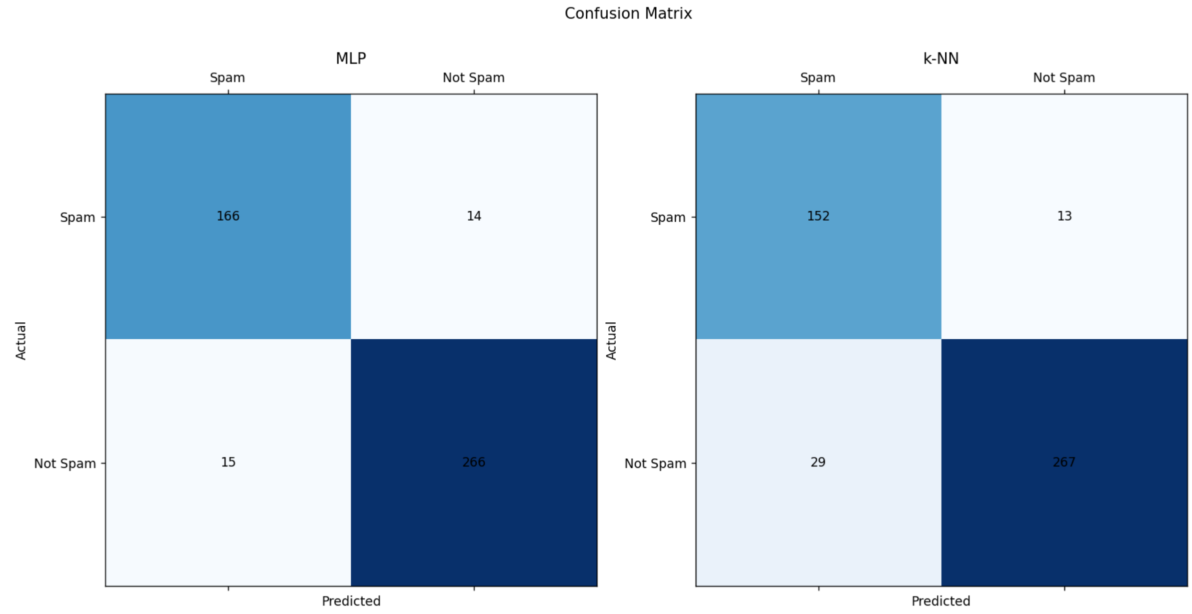

Figure 4: Confusion Matrix of k-NN and MLP classifiers

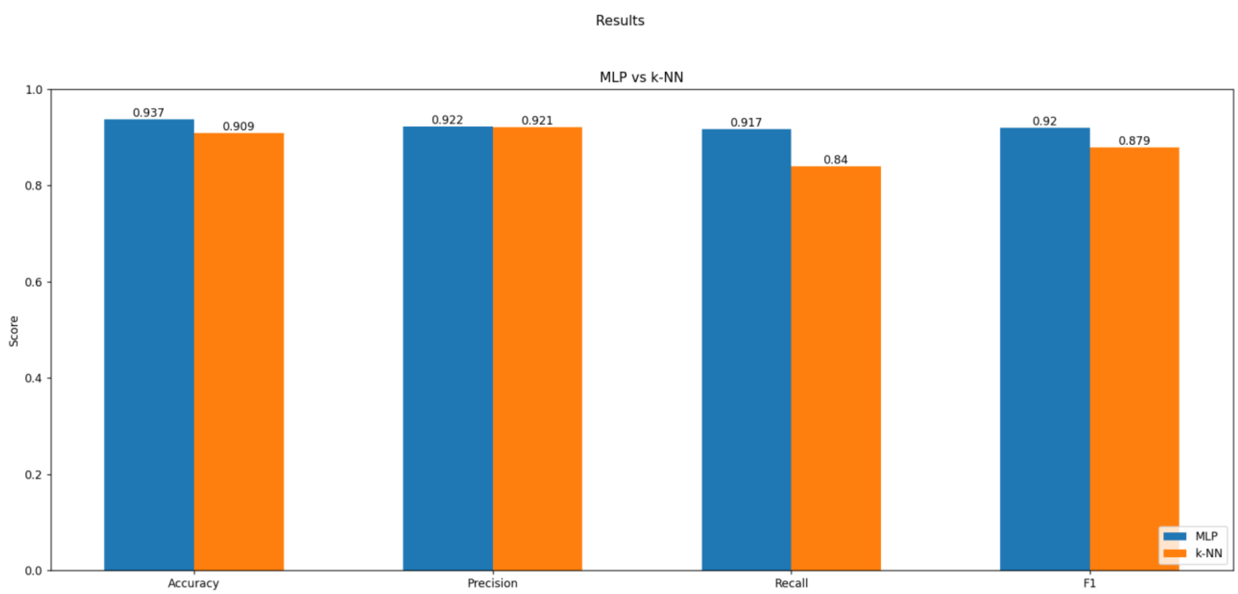

Figure 5: Performance of the k-NN and MLP classifiers

When the value of K is set to 7 and the test size is set to 0.15:

- The MLP classifier correctly identified 166 emails as spam, while the k-NN classifier correctly identified 152 emails as spam.

- The MLP classifier incorrectly labeled 14 emails as spam when they were actually not spam, whereas the k-NN classifier incorrectly labeled 13 emails as spam when they were actually not spam.

- The MLP classifier accurately identified 266 emails as not spam, and the k-NN classifier accurately identified 267 emails as not spam.

- The MLP classifier had 15 false negatives, incorrectly classifying 15 spam emails as not spam, while the k-NN classifier had 29 false negatives, incorrectly classifying 29 spam emails as not spam.

When K was increased to 7 and the test size was reduced to 0.15, the k-NN classifier showed improved performance compared to the previous scenario with K=3 and a test size of 0.3. Although the MLP classifier still outperformed the k-NN classifier, the new configuration demonstrated better results.

Increase k and keeping the test size the same (test size = 0.3)

Increasing K allows the k-NN classifier to consider more neighbors during classification. This can help in reducing the impact of noisy data points and improve the generalization of the classifier. (We will chose k where it is not too large to avoid over-filtering).

Example: K = 7, Test size = 0.3

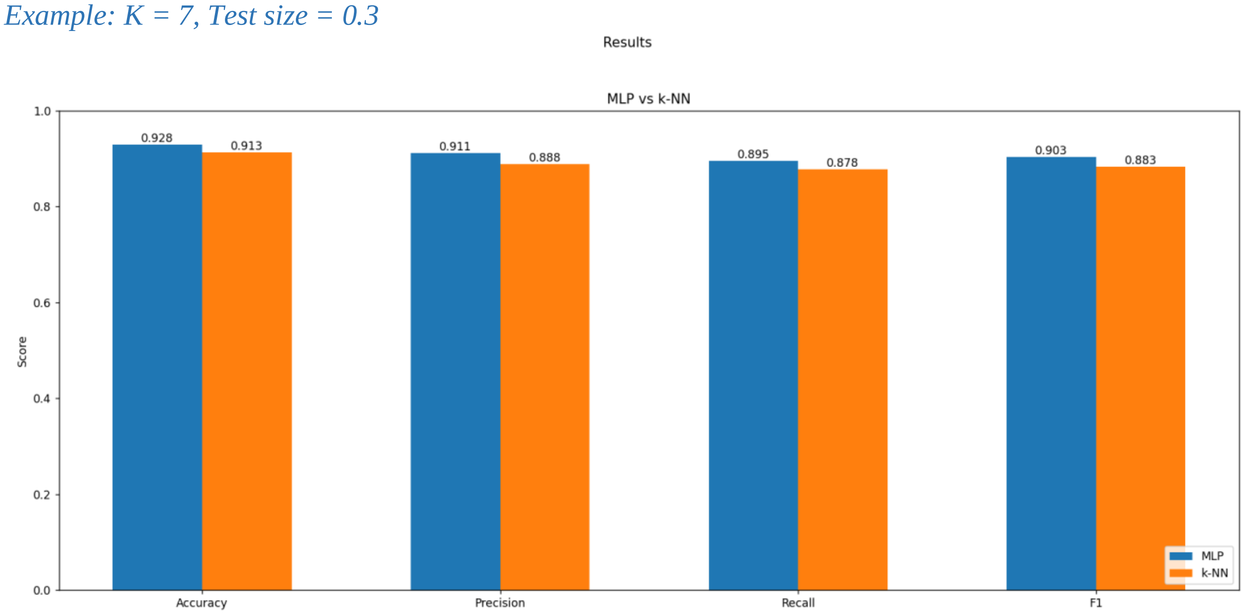

Figure 6: Confusion Matrix of k-NN and MLP classifiers

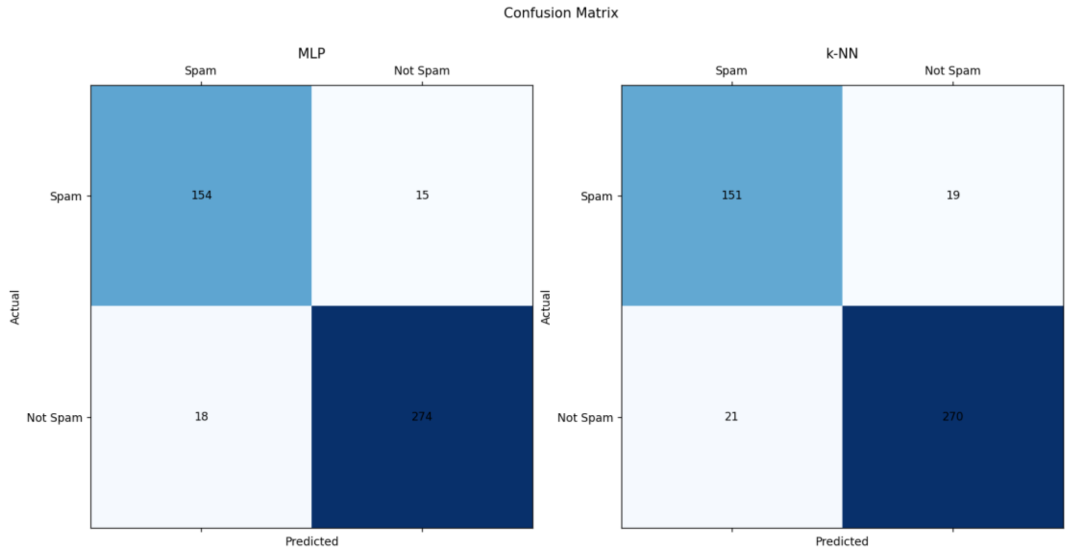

Figure 7: Performance of the k-NN and MLP classifiers

When the value of K is set to 7 and the test size is set to 0.3:

- The MLP classifier correctly identified 154 emails as spam, while the k-NN classifier correctly identified 151 emails as spam.

- The MLP classifier incorrectly labeled 15 emails as spam when they were actually not spam, whereas the k-NN classifier incorrectly labeled 19 emails as spam when they were actually not spam.

- The MLP classifier accurately identified 274 emails as not spam, and the k-NN classifier accurately identified 270 emails as not spam.

- The MLP classifier had 18 false negatives, incorrectly classifying 18 spam emails as not spam, while the k-NN classifier had 21 false negatives, incorrectly classifying 21 spam emails as not spam.

When comparing the two cases, with K=3 and a test size of 0.3, the k-NN classifier exhibited slightly better performance than in the case with K=7 and a test size of 0.3. However, it is crucial to highlight that the MLP classifier still outperformed the k-NN classifier in terms of accurately identifying spam emails and achieving higher accuracy, precision, and recall metrics.

Decrease k and decreasing the test size

Decreasing K in the k-NN classifier allows for considering fewer neighbors during classification. This approach can be beneficial for capturing finer structures and more localized patterns in the data. By reducing the value of K, the classifier becomes more sensitive to individual data points.

Example: K = 2, Test size = 0.15

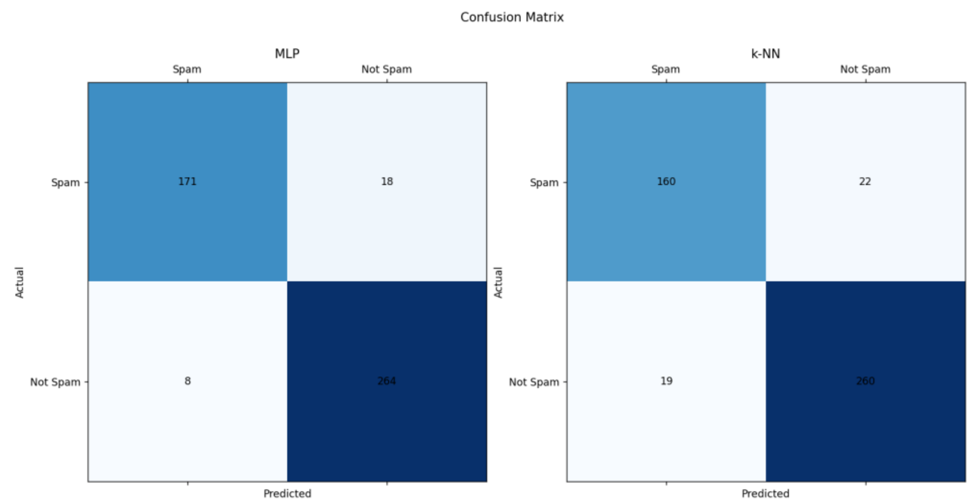

Figure 8: Confusion Matrix of k-NN and MLP classifiers

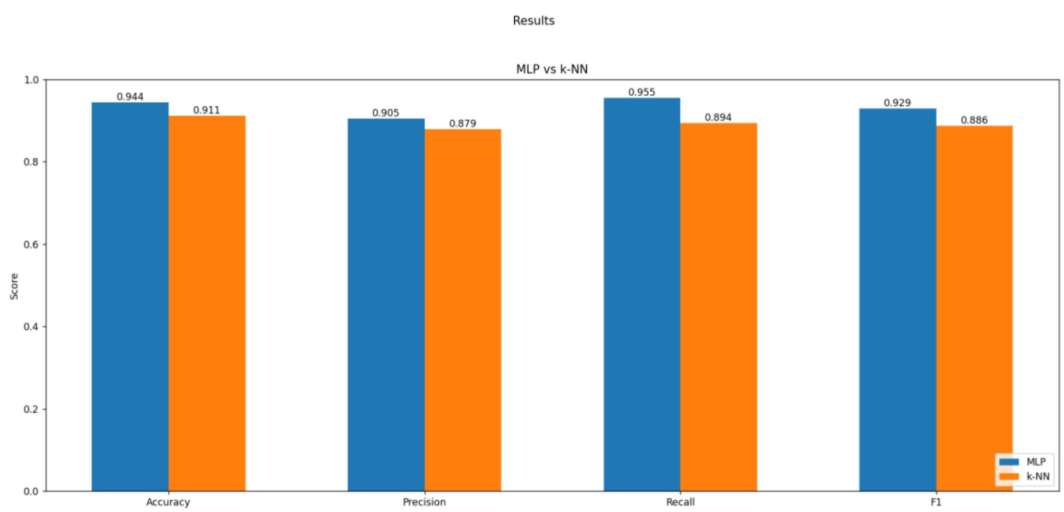

Figure 9: Performance of the k-NN and MLP classifiers

When the value of K is set to 2 and the test size is set to 0.15:

- The MLP classifier correctly identified 171 emails as spam, while the k-NN classifier correctly identified 160 emails as spam.
- The MLP classifier incorrectly labeled 18 emails as spam when they were actually not spam, whereas the k-NN classifier incorrectly labeled 22 emails as spam when they were actually not spam.
- The MLP classifier accurately identified 264 emails as not spam, and the k-NN classifier accurately identified 260 emails as not spam.
- The MLP classifier had 8 false negatives, incorrectly classifying 8 spam emails as not spam, while the k-NN classifier had 19 false negatives, incorrectly classifying 19 spam emails as not spam.

For the scenario with K=2 and a test size of 0.15, the k-NN classifier showed slightly improved performance compared to the K=3 and test size=0.3 scenario. However, it is important to note that the MLP classifier still outperformed the k-NN classifier in terms of correctly identifying spam emails and achieving higher accuracy, precision, and recall.

Therefore, while adjusting the values of K and test size can influence the performance of the k-NN classifier. The MLP classifier, with its 2 hidden layers (10 neurons, 5 neurons), consistently outperformed the k-NN classifier in the given scenarios. This can be attributed to its ability to learn complex non-linear decision boundaries, extract meaningful feature representations, and optimize its performance through backpropagation. These factors contribute to the MLP's superior classification accuracy compared to the k-NN classifier.

Improvements

To improve the performance of the tested models. Firstly, increasing the training size can potentially enhance accuracy and generalization. With a larger dataset, the models can learn from more diverse examples, leading to improved performance. Secondly, optimizing the architecture of the MLP classifier, such as adjusting the number of neurons and hidden layers, can enhance its ability to capture complex patterns in the data. Experimenting with different configurations can help improve the performance. Finally, fine-tuning the hyper parameters of both models, such as learning rate, distance metric in the case of k-NN, can lead to improved performance. By carefully considering these factors and experimenting with different settings, the performance of the models can be enhanced in classifying spam emails.

References

[1] https://link.springer.com/chapter/10.1007/978-981-15-7106-0_18 visited at 22: 17 PM on 14/7/2023

[2] https://towardsdatascience.com/spam-email-classifier-with-knn-from-scratch-python-6e68eeb50a9e visited at 23: 08 PM on 14/7/2023

[3]https://www.researchgate.net/publication/369465003_Email_Spam_Classification_using_KNN_and_Naive_Bayes visited at 23: 15 PM on 14/7/2023

[4] https://www.enjoyalgorithms.com/blog/email-spam-and-non-spam-filtering-using-machine-learning visited at 23: 37 PM on 14/7/2023

[5] Machine Learning slides visited