

**Department of Electrical & Computer Engineering**

**Second Semester (2022/2023)**

**ENCS3340, ARTIFICIAL INTELLIGENCE**

**Project # 2**

**Prepared by:**

**Hidaya Mustafa 1201910**

**Adam Nassan 1202076**

**Instructor: Dr. Ismail Khater**

**Dr. Yazan Abufarha**

**Date:15/7/2023**

**Section: 3,4**

# **Procedure:**

## **Project description :**

The project is to build a spam filter using two different machine learning models: k-nearest neighbors (KNN) and multilayer perceptron (MLP).

The KNN model is a simple and efficient algorithm that works by finding the k most similar training examples to a new example and predicting the label of the new example based on the labels of the k nearest neighbors.

The MLP model is a more complex algorithm that uses artificial neural networks to learn the relationship between the features of the training examples and their labels.

The project evaluated using the following metrics: accuracy, precision, recall, and F1-score. Accuracy is the fraction of test examples that are correctly classified. Precision is the fraction of predicted positive examples that are actually positive. Recall is the fraction of actual positive examples that are correctly classified. F1-score is a measure of both precision and recall.

In addition to that we find the conflict matrix is a useful tool for understanding the relationship between different features in a dataset. It help me to identify potential problems in your data and to make better decisions about how to model it.

The project implemented in Python using the NumPy, SciPy, and Scikit-learn libraries. The project evaluated on a dataset of spam and ham emails.

The results of the project are to build a spam filter that achieves high accuracy, precision, recall, and F1-score. The project also provide insights into the performance of the two machine learning models.

# **Result :**

## **K-NN :**

The table we provided shows the confusion matrix for the spam filter. The confusion matrix is a table that shows how many instances were correctly classified and how many were incorrectly classified. The rows of the table correspond to the actual values, and the columns correspond to the predicted values.

In this case, the actual values are whether an email is spam (positive) or ham (negative). The predicted values are what the spam filter predicted the email to be.

The table shows that the spam filter correctly classified 478 spam emails and 782 ham emails. However, it also incorrectly classified 46 ham emails as spam and 75 spam emails as ham.

The overall accuracy of the spam filter is 0.9102, which means that it correctly classified 91.02% of the emails. However, the precision of the spam filter is 0.8719, which means that it correctly classified 87.19% of the spam emails. The recall of the spam filter is 0.8644, which means that it correctly classified 86.44% of the ham emails.

The precision and recall of the spam filter are both important metrics. Precision measures how accurate the spam filter is at predicting spam emails, while recall measures how complete the spam filter is at predicting spam emails.

In this case, the spam filter has a high accuracy, but it has a lower precision and recall. This suggests that the spam filter is good at classifying emails as spam or ham, but it is not as good at distinguishing between spam emails and ham emails.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Actual values** | | |
| **Predicted values** |  | **Positive** | **Negative** |
| **Positive** | TP = 478 | FP = 46 |
| **Negative** | FN = 75 | TN = 782 |

The first thing we note is that the accuracy of the model increases as the number of nearest neighbors increases. This is because the model is able to learn more about the relationship between the features of the spam and ham emails and their labels as it considers more neighbors.

However, the precision and recall of the model decrease as the number of nearest neighbors increases. This is because the model becomes more conservative in its predictions as it considers more neighbors. It is more likely to predict a spam email as ham, and vice versa.

The F1-score is a measure of both precision and recall, so it takes both into account. As a result, the F1-score of the model also decreases as the number of nearest neighbors increases.

Overall, the results suggest that the best number of nearest neighbors for this model is 3. This gives the model a good balance of accuracy, precision, and recall.

**\*\*\*\* 1st-Nearest Neighbor Results \*\*\*\***

**Accuracy: 0.9102099927588704**

**Precision: 0.9097472924187726**

**Recall: 0.8719723183391004**

**F1: 0.8904593639575972**

**\*\*\*\* 3rd-Nearest Neighbor Results \*\*\*\***

**Accuracy: 0.9123823316437364**

**Precision: 0.9122137404580153**

**Recall: 0.864376130198915**

**F1: 0.8876508820798514**

**\*\*\*\* 10th-Nearest Neighbor Results \*\*\*\***

**Accuracy: 0.9080376538740044**

**Precision: 0.9288793103448276**

**Recall: 0.820952380952381**

**F1: 0.8715874620829119**

## **MLP :**

The results we provided show that the MLP model is able to achieve a high accuracy, precision, and recall for spam filtering. The accuracy of the model is 0.943519188993483, which means that it correctly classified 94.35% of the emails. The precision of the model is 0.9545454545454546, which means that it correctly classified 95.45% of the spam emails. The recall of the model is 0.9083044982698962, which means that it correctly classified 90.83% of the ham emails.

The F1-score is a measure of both precision and recall, so it takes both into account. As a result, the F1-score of the model is also high, at 0.9308510638297872.

The results also show that the MLP model is able to achieve a high accuracy, precision, and recall even when the number of false positives and false negatives is increased. For example, when the number of false positives is increased to 34 and the number of false negatives is increased to 50, the accuracy of the model is still 0.9471397538015931, the precision of the model is still 0.9396887159533074, and the recall of the model is still 0.92.

These results suggest that the MLP model is a good choice for spam filtering. It is able to achieve a high accuracy, precision, and recall even when the number of false positives and false negatives is increased.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Actual values** | | |
| **Predicted values** |  | **Positive** | **Negative** |
| **Positive** | 503 | 794 |
| **Negative** | 34 | 50 |

**\*\*\*\* MLP Results \*\*\*\***

**Accuracy: 0.9471397538015931**

**Precision: 0.9396887159533074**

**Recall: 0.92**

**F1: 0.929740134744947**

**\*\*\*\* MLP Results \*\*\*\***

**Accuracy: 0.939174511223751**

**Precision: 0.9366852886405959**

**Recall: 0.9095840867992767**

**F1: 0.9229357798165139**

**\*\*\*\* MLP Results \*\*\*\***

**Accuracy: 0.943519188993483**

**Precision: 0.9545454545454546**

**Recall: 0.9083044982698962**

**F1: 0.9308510638297872**

**The K-NN** model is a simple and efficient algorithm that works by finding the k most similar training examples to a new example and predicting the label of the new example based on the labels of the k nearest neighbors.

The advantages of the KNN model are that it is simple to understand and implement, and it is very efficient. The KNN model can also be used to classify data that is not linearly separable.

The disadvantages of the KNN model are that it can be sensitive to the choice of k, and it can be computationally expensive to find the k nearest neighbors for a new example.

**The MLP** model is a more complex algorithm that uses artificial neural networks to learn the relationship between the features of the training examples and their labels.

The advantages of the MLP model are that it can achieve a higher accuracy than the KNN model, and it is not as sensitive to the choice of hyperparameters. The MLP model can also be used to classify data that is not linearly separable.

The disadvantages of the MLP model are that it is more complex to understand and implement, and it can be computationally expensive to train.

**To improve the KNN and MLP models**

The KNN model can be improved by using a better method for choosing k. One way to do this is to use cross-validation to evaluate different values of k. The KNN model can also be improved by using a more efficient algorithm for finding the k nearest neighbors.

The MLP model can be improved by using a better set of hyperparameters. One way to do this is to use grid search to search for the best set of hyperparameters. The MLP model can also be improved by using a more powerful neural network architecture.

The KNN and MLP models are both effective spam filtering models. The KNN model is simpler to understand and implement, but the MLP model can achieve a higher accuracy. Both models can be improved by using a better method for choosing hyperparameters or by using a more powerful neural network architecture.