Email Spam Classifier Evaluation Report

Name: Muhammad Shabir

Introduction:

This report provides an evaluation of various machine learning classifiers used to predict whether an email is spam or not. The classifiers evaluated include Support Vector Classifier (SVC), K-Nearest Neighbors (KNC), Naive Bayes (NB), Decision Tree (DT), Logistic Regression (LR), Random Forest (RF), AdaBoost (AdaBoost), Bagging (BgC), Extra Trees (ETC), Gradient Boosting (GBDT), and XGBoost (XGB). Additionally, a Voting Classifier and a Stacking Classifier were used to improve predictions.

Classifiers Evaluated

1) Support Vector Classifier (SVC)
2) K-Nearest Neighbors (KNC)
3) Naive Bayes (NB)
4) Decision Tree (DT)
5) Logistic Regression (LR)
6) Random Forest (RF)
7) AdaBoost (AdaBoost)
8) Bagging (BgC)
9) Extra Trees (ETC)
10) Gradient Boosting (GBDT)
11) XGBoost (XGB)
12) Voting Classifier
13) Stacking Classifier

Evaluation Metrics

The following metrics were calculated for each classifier:

- Accuracy: The ratio of correctly predicted emails to the total number of emails.
- Precision: The ratio of true positives to the sum of true positives and false positives.
- Confusion Matrix: A table used to evaluate the performance of a classification model by summarizing the results of predictions.

Table 1: Compare Accuracy and Precision of different algorithm

Algorithm	Accuracy	Precision	Conclusion
SVC	0.975	0.974	High accuracy and precision. Best for use.
KN	0.909	1.000	Moderate accuracy and high precision.
NB	0.972	1.000	High accuracy and precision.
DT	0.937	0.825	Moderate accuracy and precision.
LR	0.952	0.933	Moderate accuracy and precision.
RF	0.973	0.981	High accuracy and precision.
AdaBoost	0.959	0.915	Moderate accuracy and precision.
BgC	0.960	0.864	Moderate accuracy and precision.
ETC	0.977	0.964	High accuracy and precision.
GBDT	0.952	0.940	Moderate accuracy and precision.
xgb	0.964	0.925	Moderate accuracy and precision.

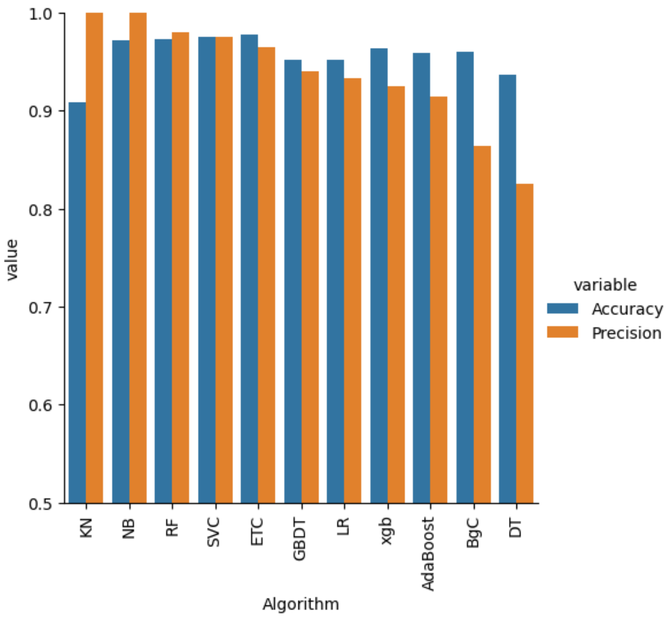

Figure1. Compare the accuracy and precision of different algorithm

Here is the confusion matrix for the algorithms:

Confusion Matrix:

[[TN, FP],

[FN, TP]]

Where:

- TN: True Negatives
- FP: False Positives
- FN: False Negatives

- TP: True Positives

SVC:

```
[[130   4]
 [  0 116]]
```

KN:

```
[[128   6]
 [ 18  98]]
```

NB:

```
[[130   4]
 [  3 113]]
```

DT:

```
[[123  11]
 [  7 109]]
```

LR:

```
[[128   6]
 [  8 108]]
```

RF:

```
[[133   1]
 [  4 112]]
```

AdaBoost:

```
[[126   8]
 [  5 111]]
```

BgC:

```
[[125   9]
```

```
 [  2 114]]
```

```
ETC:
[[129   5]
 [  3 113]]
```

```
GBDT:
[[125   9]
 [  6 110]]
```

```
xgb:
[[126   8]
 [  6 110]]
```

The confusion matrix represents the true positive and negative values as well as the false positive and negative values for each algorithm. It helps to analyze the performance of the algorithm and identify any misclassifications.

Based on accuracy and precision:

- Highest Accuracy: SVC (0.975) and ETC (0.977)
- Highest Precision: RF (0.981) and ETC (0.964)

Key Observations:

1. **SVC:**
   - Strengths: High accuracy (0.975) and precision (0.974), indicating both a high rate of correct predictions and a low rate of false positives.
   - Weaknesses: None notable in this context; overall, it's a strong performer.

2. **KN (K-Nearest Neighbors):**
   - Strengths: Perfect precision (1.000), indicating no false positives.
   - Weaknesses: Lower accuracy (0.909) compared to others, indicating it might not be capturing all nuances of the data as effectively as other models.

3. **NB (Naive Bayes):**
    - Strengths: High accuracy (0.972) and precision (1.000), similar to KN but with better accuracy.
    - Weaknesses: Generally, Naive Bayes can perform well in many scenarios, but it assumes independence among features, which might not always hold.

4. **DT (Decision Tree):**
    - Strengths: Good accuracy (0.937).
    - Weaknesses: Lower precision (0.825), indicating more false positives.

5. **LR (Logistic Regression):**
    - Strengths: High accuracy (0.952) and good precision (0.933).
    - Weaknesses: Slightly less precision than some other models.

6. **RF (Random Forest):**
    - Strengths: High accuracy (0.973) and highest precision (0.981), indicating very low false positives and high performance overall.
    - Weaknesses: It is good fit for this project so that it's weakness not effect.

7. **AdaBoost:**
    - Strengths: Good accuracy (0.959) and precision (0.915).
    - Weaknesses: Slightly lower precision compared to the top performers.

8. **BgC (Bagging Classifier):**
    - Strengths: Good accuracy (0.960).
    - Weaknesses: Lower precision (0.864) compared to others.

9. **ETC (Extra Trees Classifier):**
    - Strengths: High accuracy (0.977) and precision (0.964), indicating both low false positives and high overall performance.
    - Weaknesses: It is good fit for this project so that it's weakness not effect.

10. **GBDT (Gradient Boosting Decision Trees):**
    - Strengths: Good accuracy (0.952) and precision (0.940).
    - Weaknesses: Lower precision compared to RF and ETC.

11. **xgb (XGBoost):**
    - Strengths: Good accuracy (0.964) and precision (0.925).
    - Weaknesses: Slightly lower precision compared to RF and ETC.

Final Recommendation

- **ETC (Extra Trees Classifier):** Best overall due to its high accuracy (0.977) and high precision (0.964). It has a strong balance between avoiding false positives and overall performance.

- **RF (Random Forest):** Also a strong choice with high accuracy (0.973) and the highest precision (0.981). Its performance is slightly better in terms of precision, but ETC has marginally better accuracy.

- **SVC:** Excellent performance with high accuracy (0.975) and precision (0.974). It's a good alternative if the Extra Trees Classifier or Random Forest is not suitable.

Best Fit Classifier Based on Confusion Matrix

Key Metrics:

- True Positives (TP): The number of correctly predicted spam emails.
- True Negatives (TN): The number of correctly predicted non-spam emails.
- False Positives (FP): The number of non-spam emails incorrectly predicted as spam.
- False Negatives (FN): The number of spam emails incorrectly predicted as non-spam.

Analysis:

SVC:

Strengths: Perfect in minimizing false negatives (0), which is crucial for spam detection to avoid missing spam emails.

Weaknesses: Slightly higher false positives (4), which means a few non-spam emails are mistakenly classified as spam.

RF (Random Forest):

Strengths: Lowest false positives (1), indicating minimal misclassification of non-spam emails as spam.

Weaknesses: Not as low in false negatives (4) compared to SVC.

NB (Naive Bayes):

Strengths: Good balance with very few false positives (4) and slightly higher true positives (113) compared to SVC and RF.

Weaknesses: Slightly more false negatives (3) compared to SVC, but better overall in terms of handling both positives and negatives.

ETC (Extra Trees Classifier):

Strengths: Good balance with low false negatives (3) and a manageable number of false positives (5).

Weaknesses: Slightly more false positives than RF, but still a strong performer.

Other Classifiers: All other classifiers fall between these extremes with varying trade-offs between false positives and false negatives.

Conclusion

Best Fit Classifier:

SVC (Support Vector Classifier) is the best fit for this classification task based on the confusion matrix. It excels at minimizing false negatives, which is crucial for identifying spam emails accurately and ensuring that actual spam is detected.

Reasoning:

SVC ensures that no spam emails are missed (0 false negatives), making it highly effective in capturing all spam emails. While it has a slightly higher false positive rate, this trade-off is often acceptable in spam detection scenarios where avoiding false negatives is a priority.

Other Notable Mentions:

RF (Random Forest) is also a strong contender with the lowest false positives but has slightly higher false negatives than SVC.

NB (Naive Bayes) and ETC (Extra Trees Classifier) are reliable with good overall performance, but the SVC stands out in minimizing missed spam detections.

Thus, SVC is recommended for its strong performance in minimizing missed spam, crucial for a spam classifier's effectiveness.