Spoken and Natural Language Understanding Practical Assignment 2

Submitted by Jiahui Dai

April 26, 2025

Accompanying material: Assignment_2.ipynb

1 SMS Spam Detection

Naive Bayes Classifier is used to detect spam in this task [**raschka2014naive**].

1.1 Data Preprocessing Steps

Steps:

1. Tokenisation: breaking down a text corpus into individual tokens
2. Stop words removal: to remove words that are relatively common and uninformative
3. Stemming and Lemmatization:
   (a) Stemming: the transformation a word into its root form
   (b) Lemmatization: to obtain the grammatically correct forms of words
4. Data Splitting: Random 80-20 data split into training and test data

1.2 Experimental Design and Methods

1. Split the data into training and test data
2. Building of two different vectorisors based on its features:
   (a) **Bag-of-Words** (BoW)
   BoW treats each document as a collection of words, disregarding grammar and word order but keeping track of word frequency.
   (b) **TF-IDF**
   TF-IDF (Term Frequency-Inverse Document Frequency) is used to evaluate how important a word is to a document in a collection or corpus. It helps to weigh terms based on their frequency within a document and their rarity across all documents.

$$TF(t,d) = \frac{\text{\# terms of } t \text{ appears in document } d}{\text{Total \# of terms in document } d} \tag{1}$$

$$IDF(t) = \log\left(\frac{\text{Total \# of documents}}{\text{\# of documents containing } t}\right) \tag{2}$$

$$TF - IDF(t,d) = TF(t,d) \cdot IDF(t) \tag{3}$$

Term Frequency (TF) measures how frequently a term occurs in a document.

Inverse Document Frequency (IDF) measures the importance of a term across all documents in the corpus.

TF-IDF is the product of TF and IDF. It means that

- If a term appears frequently in a document but also in many documents, the TF will be high but the IDF will be low, lowering the overall score.
- If a term appears frequently in one document but is rare across all documents, the score will be high, emphasizing the importance of that term for the document.

3. Trained using Multinomial Naive Bayes form scikit-learn
4. Compare performance on the same train/test split.

1.3 Hyperparameters

Laplace smoothing is used to find tune the model to obtain the best/optimal evaluation metrics. (Default: $\alpha = 1.0$)

1.4 Evaluation Metric [google_accuracy]

- TP: A spam message is correctly classified as spam
- TN: A ham (non-spam) message is correctly classified as ham
- FP: A ham message is incorrectly classfied as spam
- FN: A spam message is incorrectly classfied as ham

Accuracy is the proportion of all classifications that were correct, whether positive or negative.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

Recall or **true positive rate (TPR)** is the proportion of all actual positives that were classified correctly as positives.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5}$$

Precision is the proportion of all the model's positive classifications that are actually positive.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{6}$$

F_β **Score (F1 Score)** provides a balanced measure of a model's precision (P) and recall (R).

$$F_\beta = (1 + \beta^2)\frac{P \cdot R}{\beta^2 \cdot P + R} \tag{7}$$

1.5 Findings

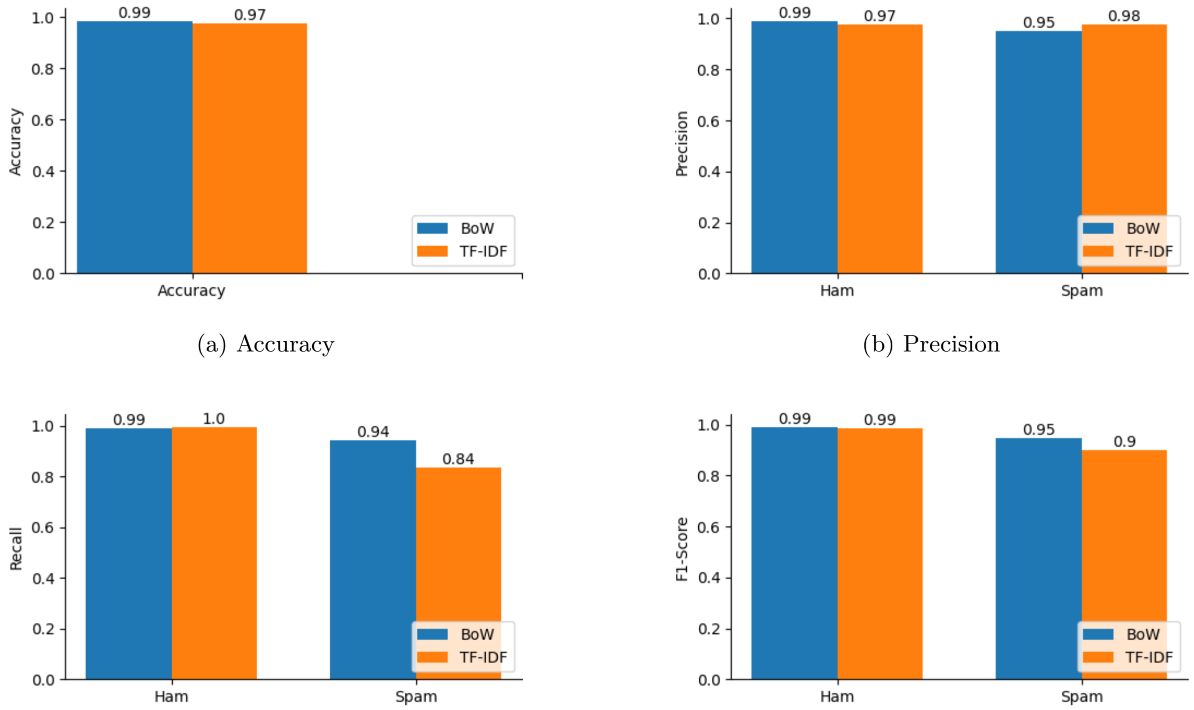
Performance metrics is shown in Figure 1

Accuracy In Figure 1a, both models have high accuracies, which suggests that both models are able to correctly predict well and accurately the labels for the test data, whether 'Ham' or 'Spam'.

Precision In Figure 1b, both models have high precision for both 'Ham' and 'Spam' labels, which suggests that both models are able to correctly well and predict precisely the labels for the test data, on both 'Ham' and 'Spam'.

Recall In Figure 1c, both models have 1.0 score for recall on 'Ham' and 'Spam', which suggests that both models were able to correctly identify label 'Ham' and 'Spam' test data. TF-IDF model still have lower recall for 'Spam' compared to BoW model, suggesting that it is not as recallative as BoW model.

F1 Score In Figure 1d, both models have high F1-score on both 'Ham' and 'Spam', which suggests a good balance between precision and recall for 'Ham' and 'Spam'.



(a) Accuracy

(b) Precision

(c) Recall

(d) F1 Score

Figure 1: Performance Metrics between BoW and TF-IDF models

Higher score on 'Ham' compared to 'Spam' There is a higher score on 'Ham' than 'Spam' is highly due to the imbalanced data of 'Ham' and 'Spam'. There is a total of 4516 or approximately 87.4% 'Ham' labelled documents and 653 or approximately 12.6% 'Spam' labelled documents.

Overall better performance of BoW Model The BoW model has a higher corresponding metric score compared to TF-IDF model, with slightly better overall identification of 'Spam' messages. The TF-IDF model has a slightly lower precision and lower recall for 'Spam' category (which also lowers the F1 score).

1.6 Drawbacks

- Manual vectorization is slower than optimized libraries.
- No deep contextual understanding (can't catch subtle spam phrasing).

1.7 Potential Improvements

- Use higher n-grams for better context (Currently only unigram)

2 Search Engine

3 Additional Experiments