1.Notebook Flow Description:

The notebook begins with

1. data cleaning, where HTML tags, punctuation, dates, and email addresses are removed from the text data.
2. Text processing :
   - **a.** where tokenization is performed to split the text into individual words so this should be the first step before stating processing so we can work on a single word
   - **b.** secondly we apply stop word removal to eliminate common English words that may not carry significant meaning
   - **c.** After that, part-of-speech (POS) tagging is applied to identify the grammatical parts of each word , it is put before lemmatization to facilitate the classification as some words
     Are written the same way but in the context they are either verb or noun
   - **d.** Lemmatization is then performed to reduce words to their base or dictionary form

3. Model accuracy function to apply the accuracy techniques
4. Applying word embedding techniques
   We used Word2vec , doc2Vec , TF-IDF and bag of words
   Then pass every one of them to once for logistic regression
   And same way for decision tree
5. Finally evaluating the model accuracies by calling the function in step 4

2.Data Preprocessing & Features Extraction:

First off we applied data cleaning in order to removes unnecessary elements like HTML tags, punctuation, dates, and email addresses to improve the quality of text data.

2 .Tokenization: Splits text into tokens , so each later process focuses only on a single word and it should be added first.

3 .Stop word removal: to eliminates common English words that do not contribute much to the meaning of the text like (prepositions , articles , pronouns , conjunctions)

4 .Post of word tagging : to classify each word , it is put before lemmatization to improve the accuracy of lemmatization which word is a noun and which is a verb

5 .Lemmatization: to reduces words to their base or dictionary form

We didn’t need stemming because lemmatization is more generalized and more accurate

6 . we applied word embedding to extracts meaningful features from text data, enabling ML models to learn patterns and relationships , word2vec and doc2vec are easy to train so we choose them over the others like Bert which is based on transformers and it will take time to

train for non neural network models BoW and TF-idf were simple to train and code so we choose them

3.Data Splitting:

The data is split into training and testing sets with a 60-40 ratio , randomly

4.Model Training:

Logistic Regression, Decision Tree classifiers are chosen for training. Text embedding techniques like Word2Vec and Doc2Vec for neural network choice and bag of words as well as TF-IDF for non neural network

5.Model Evaluation:

A . Accuracy test : Accuracy measures the overall correctness of the model's predictions, which is important in binary classification tasks like spam detection.

B. precision test :focuses specifically on the quality of positive predictions to tell us how many of the emails classified as spam are actually spam

6.Dominant models :

From the table we chooses Word2Vec and Bag of words as a dominant models

Because they produced the highest accuracy values among the other models on classifying the spam emails

	model	accuracy_test	precision_test
0	Logistic Regression (w2v)	0.984045	0.984869
1	Logistic Regression (TF-IDF)	0.969815	0.997041
2	Logistic Regression (BoW)	0.994825	0.994580
3	Logistic Regression (Doc2Vec)	0.962915	0.960674
4	Decision Tree (Word2Vec)	0.959034	0.921773
5	Decision Tree (Doc2Vec)	0.873221	0.826531
6	Decision Tree (TF-IDF)	0.978008	0.956407
7	Decision Tree (BoW)	0.979301	0.960212