Email Threat Detection Using Machine Learning

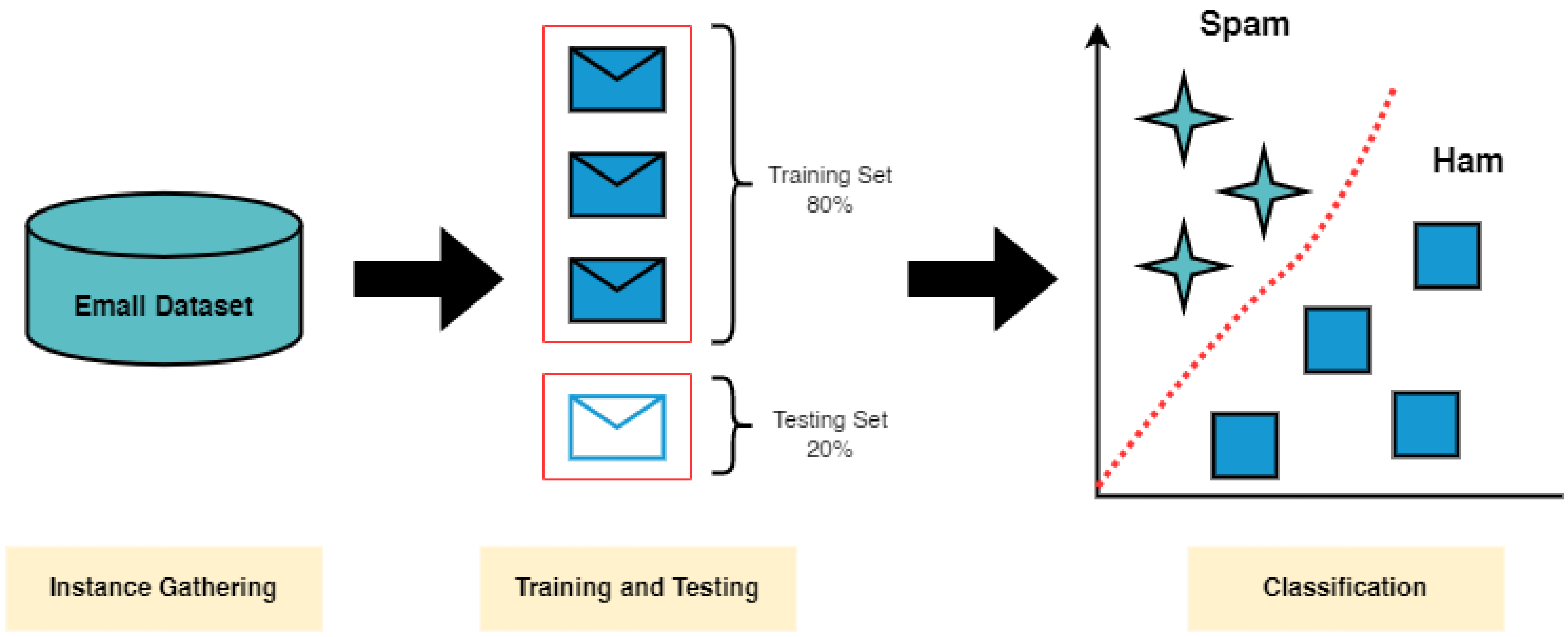
For CyBOK funded project:
Development of an active learning lesson plan and laboratory materials for AI for Security

Dr Hossein Abroshan

Senior Lecturer in Cyber Security

Anglia Ruskin University, Cambridge, UK

Spam detection using ML

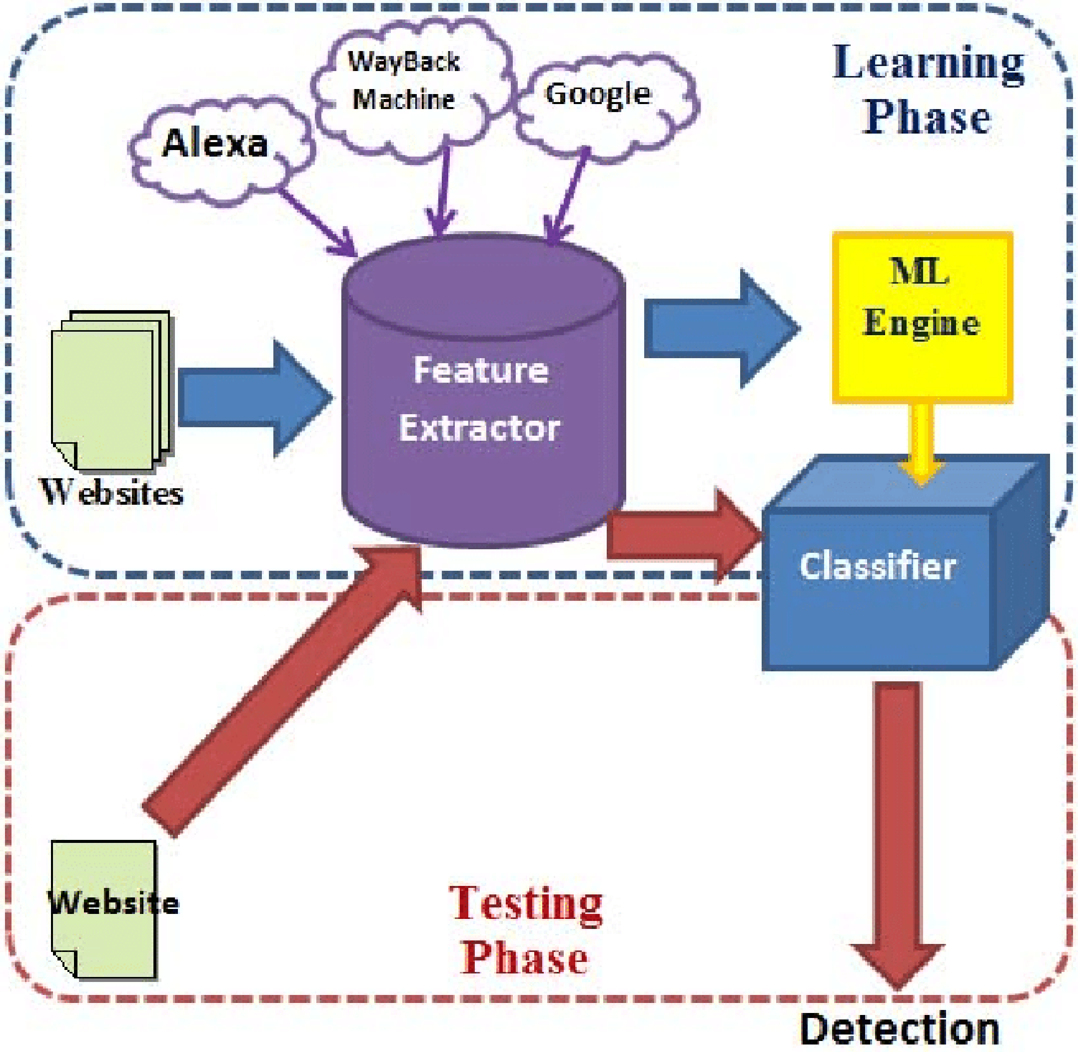


Source: Machine Learning Techniques for Spam Detection in Email

Phishing detection using ML

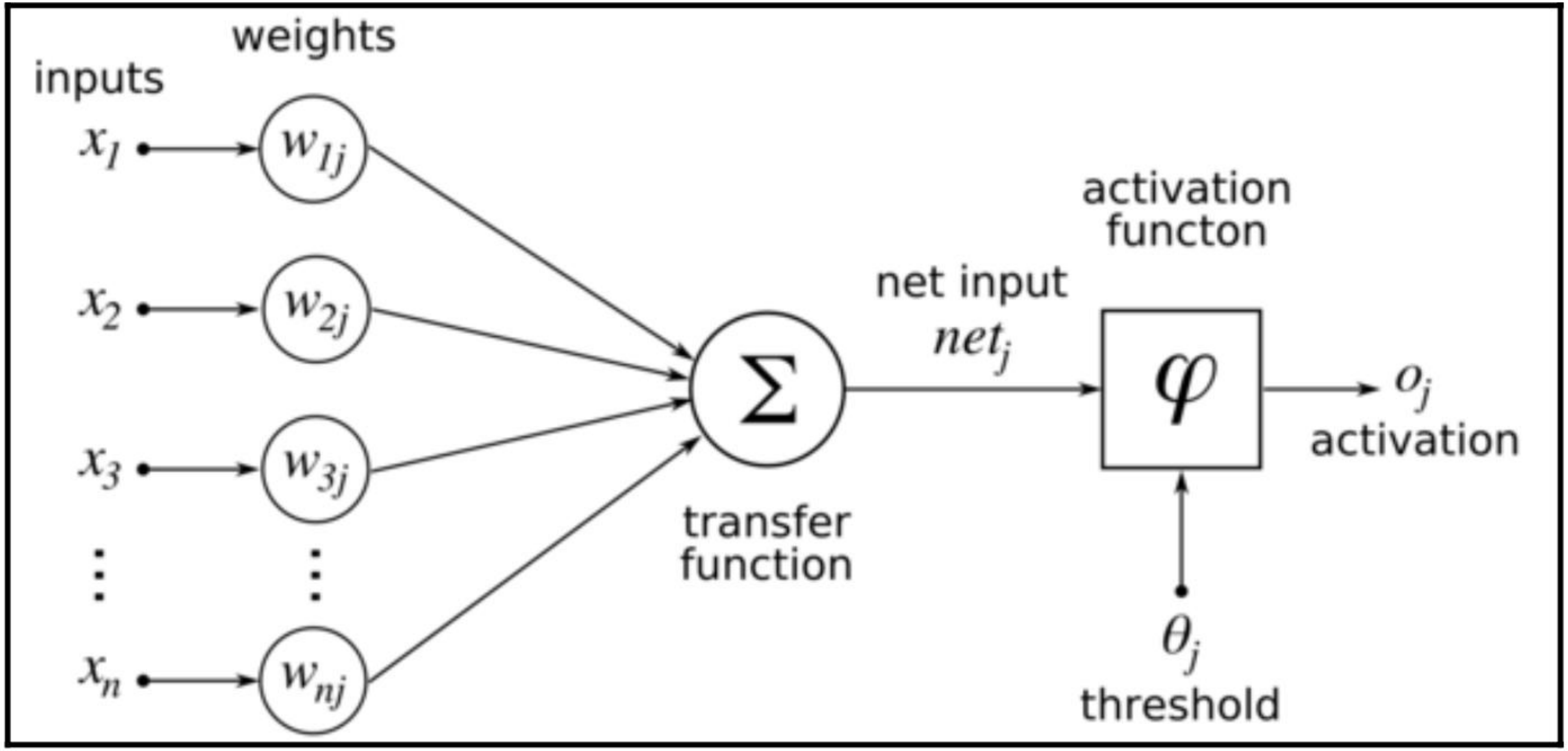
- ❖ Detect and filter phishing emails
- ❖ Detect phishing websites
- ❖ Detect phishing domains/links

For example:
Phishing website detector =>



Source: [An entice resistant automatic phishing detection]

Perceptron



Source: Performance Analysis of Open Source Machine Learning Frameworks for ...

Spam detection

Email	Buy	Sex	Spam or Ham?
1	1	0	H
2	0	1	H
3	0	0	H
4	1	1	S

$$y = B + S;$$

Email	B	S	2B + 3S	Spam or Ham?
1	1	0	2	H
2	0	1	3	H
3	0	0	0	H
4	1	1	5	S

$$y = 2B + 3S;$$

Perceptron

$$y = 2B + 3S;$$

$$y = w_1 x_1 + w_2 x_2 + \ldots + w_n x_n;$$

$$y = \sum w_i x_i$$

Just like how human nerouns work

$$w_1 x_1 + w_2 x_2 + \ldots + w_n x_n \geq \theta \rightarrow y = +1;$$
$$w_1 x_1 + w_2 x_2 + \ldots + w_n x_n < \theta \rightarrow y = -1;$$

Threshold

Perceptron learning process

- Initializing the weights to a predefined value (usually equal to *0*)
- Calculating the output value, y_i, for each corresponding training sample, x_i
- Updating the weights on the basis of the distance between the expected output value (that is, the y value associated with the original class label of the corresponding input data, x_i) and the predicted value (the y_i value estimated by the Perceptron)

Expected value

Input

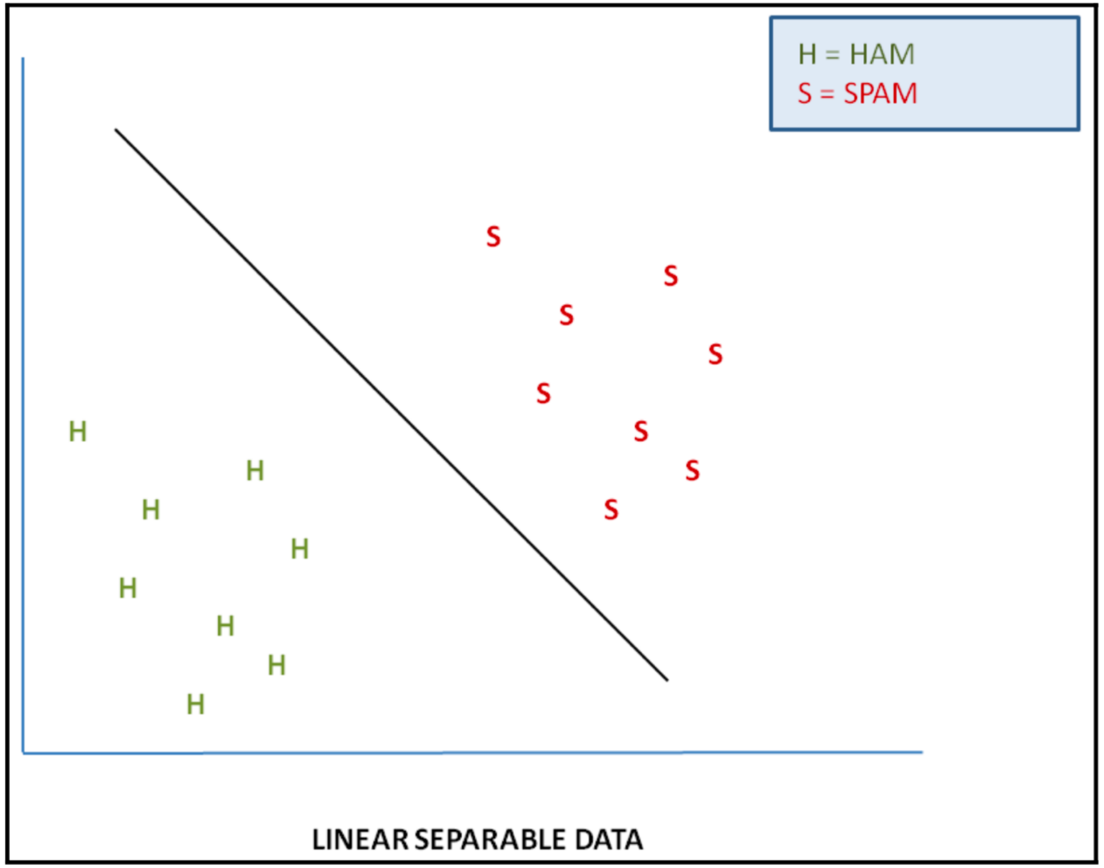
Deviation

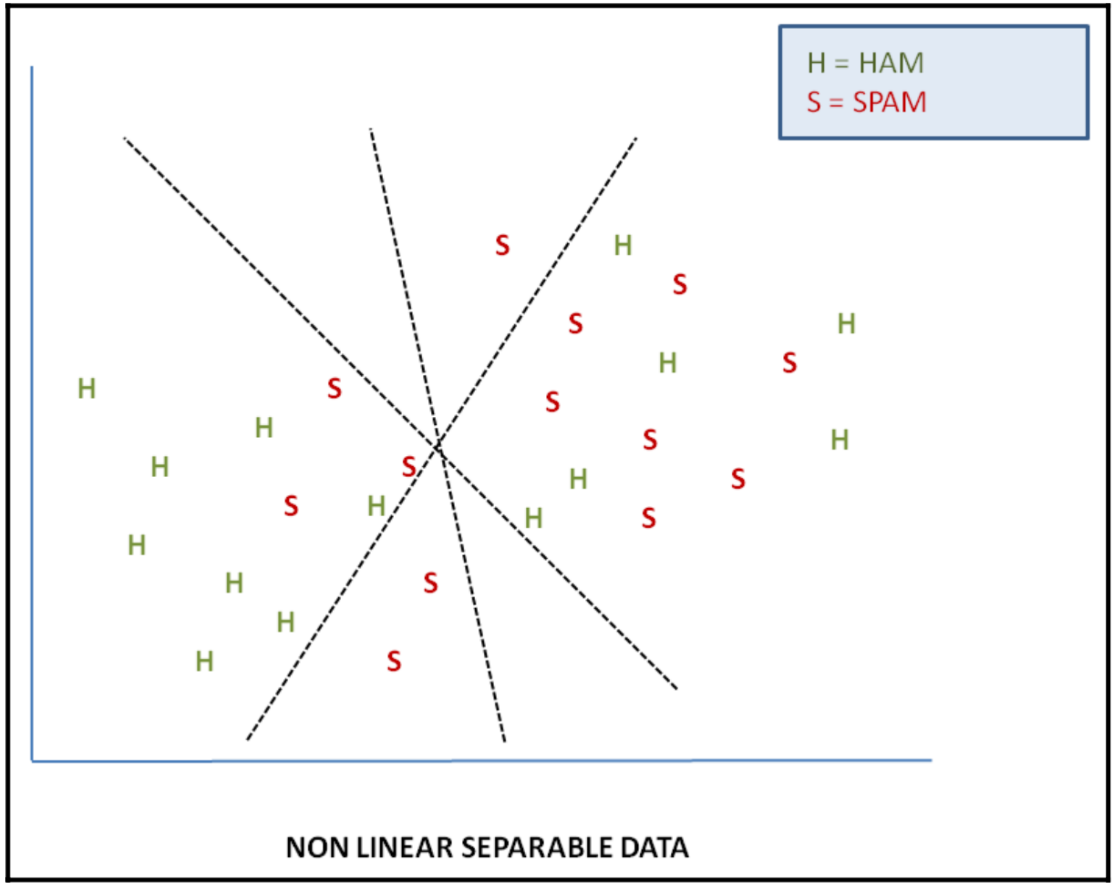
$$\Delta w_i = \lambda(y - y_i)x_i;$$

Constant
Learning rate

Predicted value

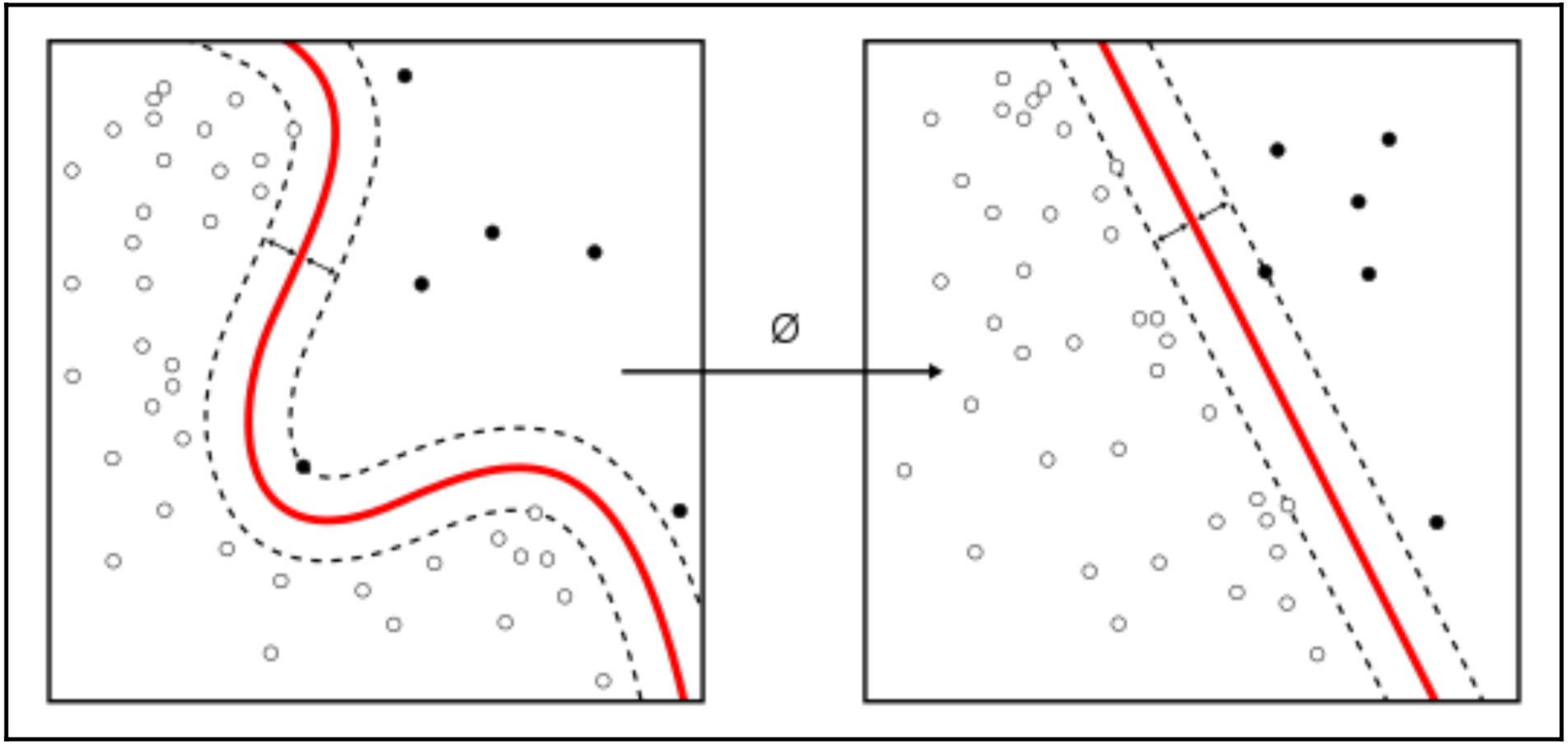
Pros and Cons





Source: [Hands-On Artificial Intelligence for Cybersecurity](book)

Using SVM



The SVM doesn't have some limitations of Perceptron

Source: [Wikimedia]

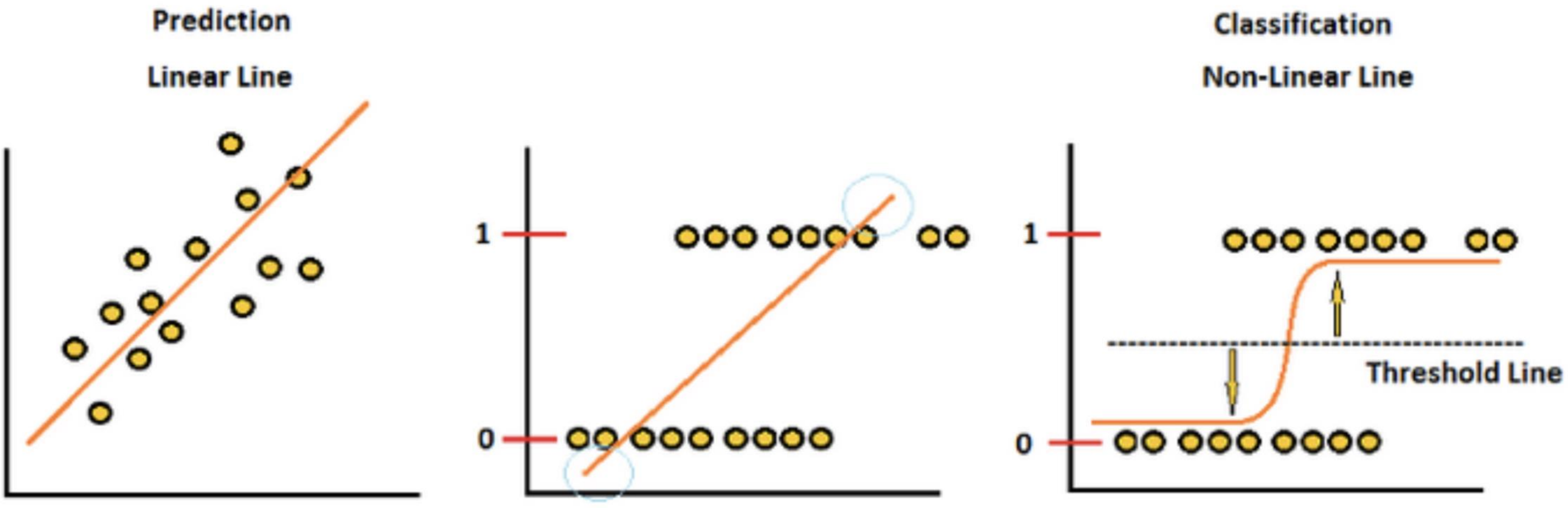
Spam detection/filtering approaches

We can use SVM to also detect image based spam emails

Spam filtering approches:

- **Content-based filtering**: The approach consists of trying to identify the suspect keywords that are most commonly used in textual spam messages even within images; to this end, pattern recognition techniques leveraging optical character recognition (**OCR**) technology are implemented in order to extract text from images (this is the solution that SpamAssassin adopts).
- **Non content-based filtering**: In this case, we try to identify specific features of spam images (such as color features and so on), on the grounds that spam images, being computer-generated, show different characteristics compared to natural images; for the extraction of the features, we make use of advanced recognition techniques based on NNs and **deep learning** (**DL**).

Linear Regression vs Logistic Regression



Constant

Coefficient

$$y = b_0 + b_1 * x_1$$

Dependent variable (DV)

Independent variable (IV)

a.k.a. **Log Odds**

or **Logit**

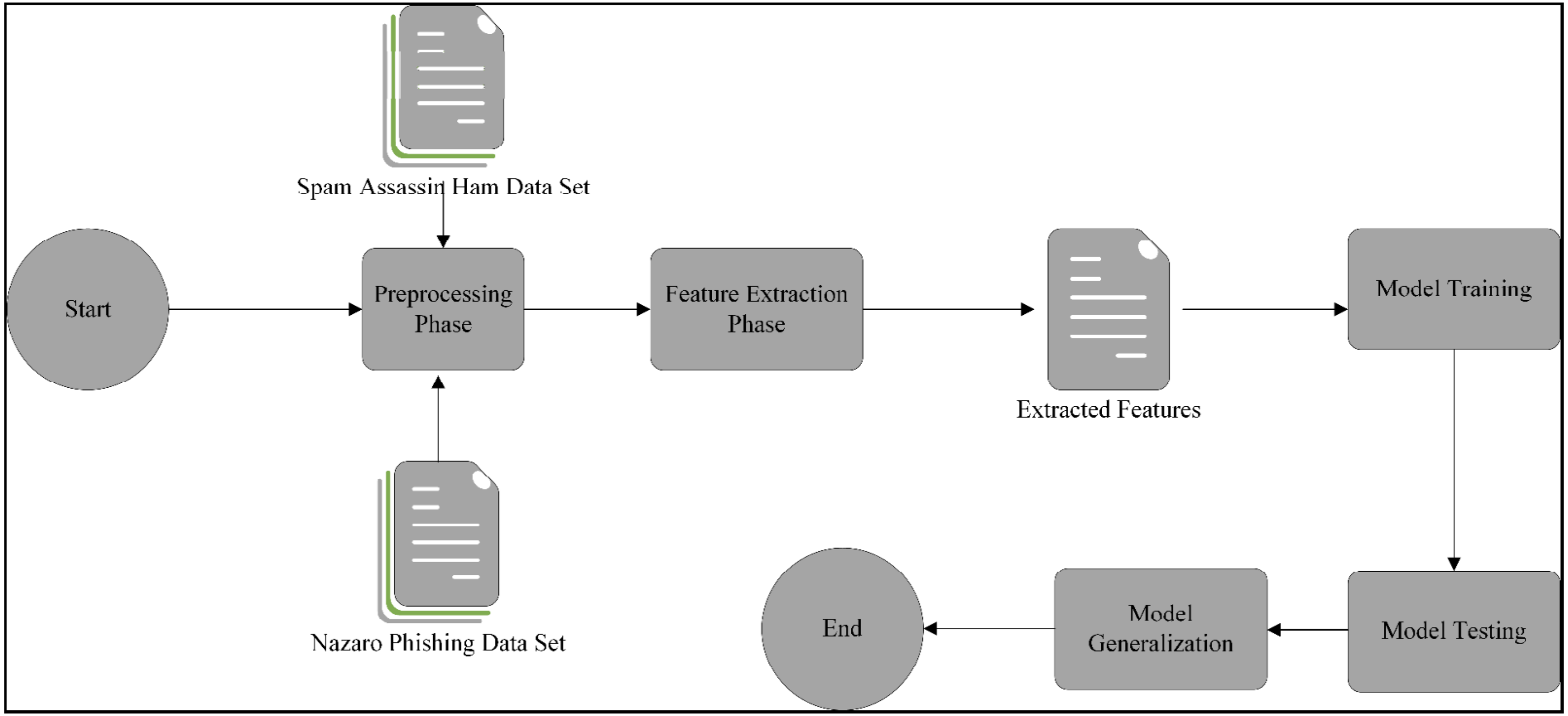
Intercept

$$\boxed{\log\left(\frac{P}{1-P}\right)} = \beta_0 + \beta_1 X$$

Source: [Machine Learning Classification Algorithms with Codes]()

Phishing email detection using Logistic Regression

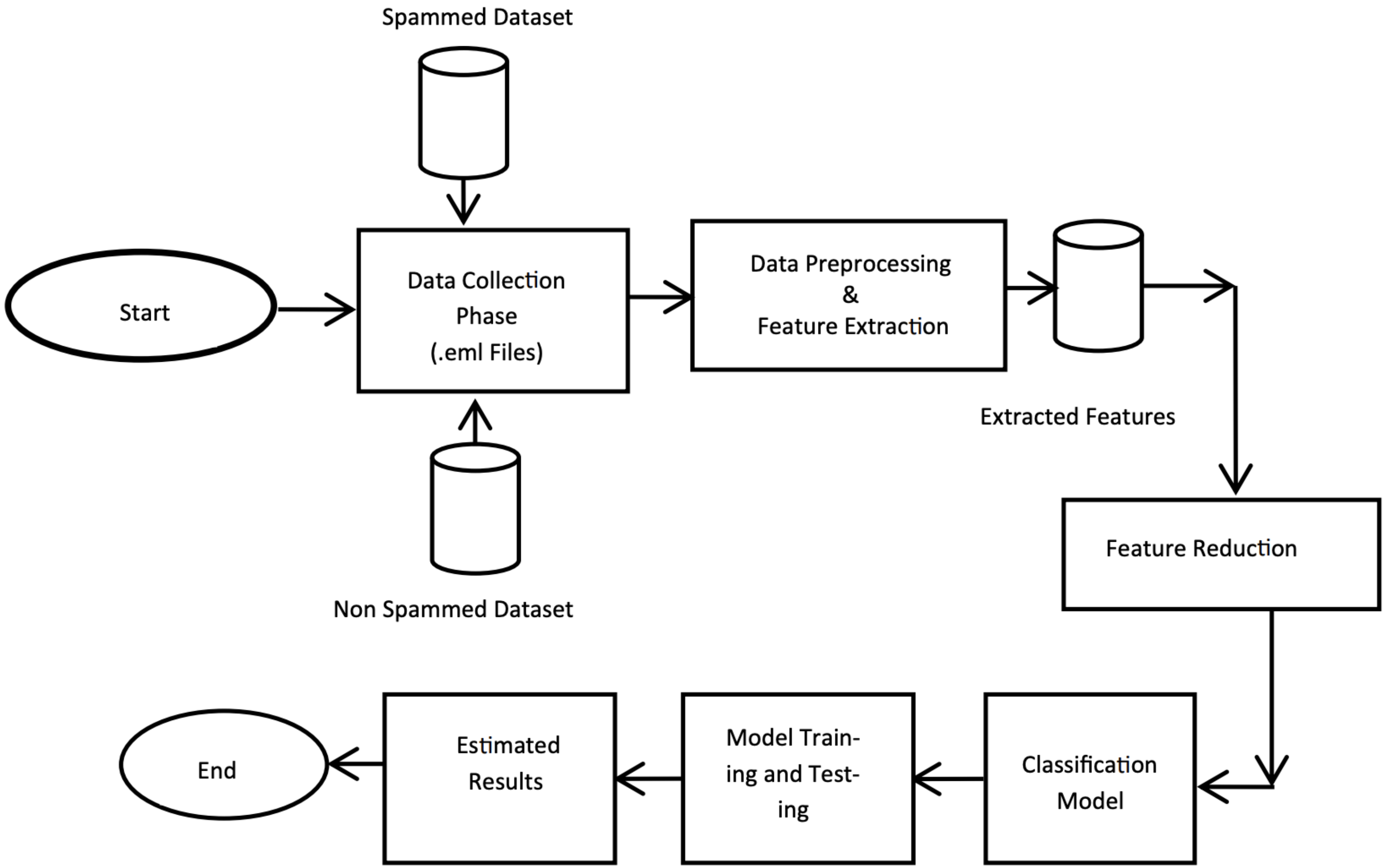
Example: A Phishing email detection solution



Source: [AN INTELLIGENT CLASSIFICATION MODEL FOR PHISHING EMAIL DETECTION]

Phishing email detection using Logistic Regression

Anti-Phishing feature selection



Source: [Feature Selection for Email Phishing Detection Using Machine Learning]()

Phishing email detection using Logistic Regression

Example of extracted features

Feature	**Description**	**Data Type**	**Information Gain**
HTML Body	Checks if the email body contains HTML content.	Number {0,1}	0.681
Hexadecimal URLs	The number of URLs consisting of hexadecimal characters in the email.	Number	0.652
Domains Count	The number of domains in the URLs that exists in the email.	Number	0.652
TextLinkDifference	The number of URLs whose label is different from its anchor in the email.	Number	0.649
Dots Count	The maximum number of dots that exist in a URL in the email.	Number	0.497
Email Contains Account Term	Checks if the email contains the term "Account"	Number {0,1}	0.493
Email Contains Dear Term	Checks if the email contains the term "Dear"	Number {0,1}	0.375
Images as URL	The number of image URLs.	Number	0.298
IP URLs	The number of URLs whose domain is specified as an IP address.	Number	0.297

Source: AN INTELLIGENT CLASSIFICATION MODEL FOR PHISHING EMAIL DETECTION

SPAM Detection Features

Example of extracted features (Attachment and URL)

Now, it's your turn Identify some features for phishing or spam detection.

Spam Attachments Features					
Habul Dataset			Botnet Dataset		
Rank	Category	Feature	Rank	Category	Feature
1	Subject	Number of capitalized words	1	Subject	Min of the compression ratio for the bz2 compressor
2	Subject	Sum of all the character lengths of words	2	Subject	Min of the compression ratio for the zlib compressor
3	Subject	Number of words containing letters and numbers	3	Subject	Min of character diversity of each word
4	Subject	Max of ratio of digit characters to all characters of each word	4	Subject	Min of the compression ratio for the lzw compressor
5	Header	Hour of day when email was sent	5	Subject	Max of the character lengths of words
		(a)			(b)

Spam URLs Features					
1	URL	The number of all URLs in an email	1	Header	Day of week when email was sent
2	URL	The number of unique URLs in an email	2	Payload	Number of characters
3	Payload	Number of words containing letters and numbers	3	Payload	Sum of all the character lengths of words
4	Payload	Min of the compression ratio for the bz2 compressor	4	Header	Minute of hour when email was sent
5	Payload	Number of words containing only letters	5	Header	Hour of day when email was sent
		(c)			(d)

Source: [improving Knowledge Based Spam Detection Methods: The Effect of Malicious Related Features in Imbalance Data Distribution]()

Spam Email Detection Using Deep Learning Techniques (example study)

Methodology

Source: [Research Paper]

Spam Email Detection Using Deep Learning Techniques (example study)

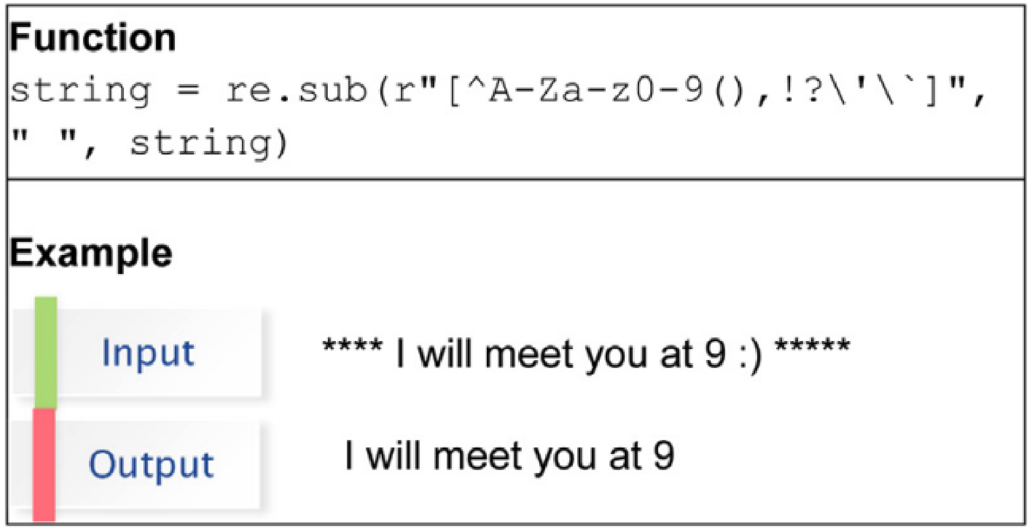
Dataset: open source Spambase data set from the UCI machine learning repository

Data set	SPAM	HAM	Total Samples
Dev data	2000	3000	5000
Hold out test data	113	113	226

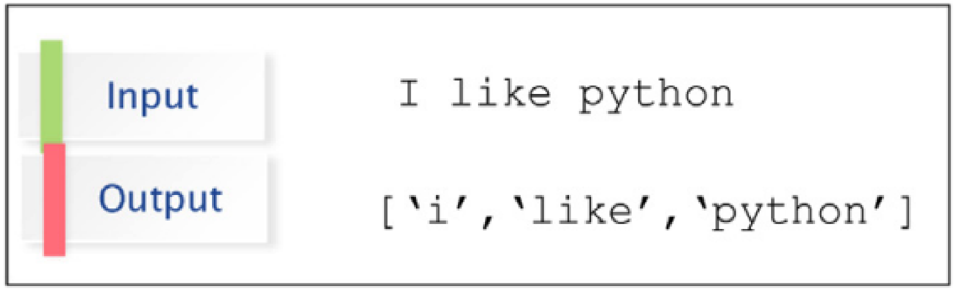
Source: Research Paper

Spam Email Detection Using Deep Learning Techniques (example study)

Examples of features: word counts, stopword counts, punctuation counts, and uniqueness factors, etc.



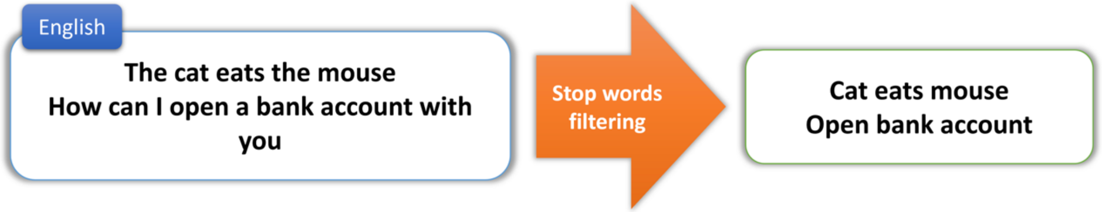
Email Cleaning



Email Tokenization

```
For each word in words:
  IF word not in stop_words and word_freq[word] >= 7:
   Email_words.append(word)
```

Stop word Removal



Source: [Research Paper 1](), [Research Paper 2](), [Stopword Removal]()

Spam Email Detection Using Deep Learning Techniques (example study)

Model	Accuracy	F1 Score
KNN	0.9310	0.9081
NB	0.9540	0.9408
BiLSTM	0.9650	0.9556
Bert Base Cased	**0.9730**	**0.9696**

Training Result on Training Data (Using Different Algorithms)

Source: [Research Paper]

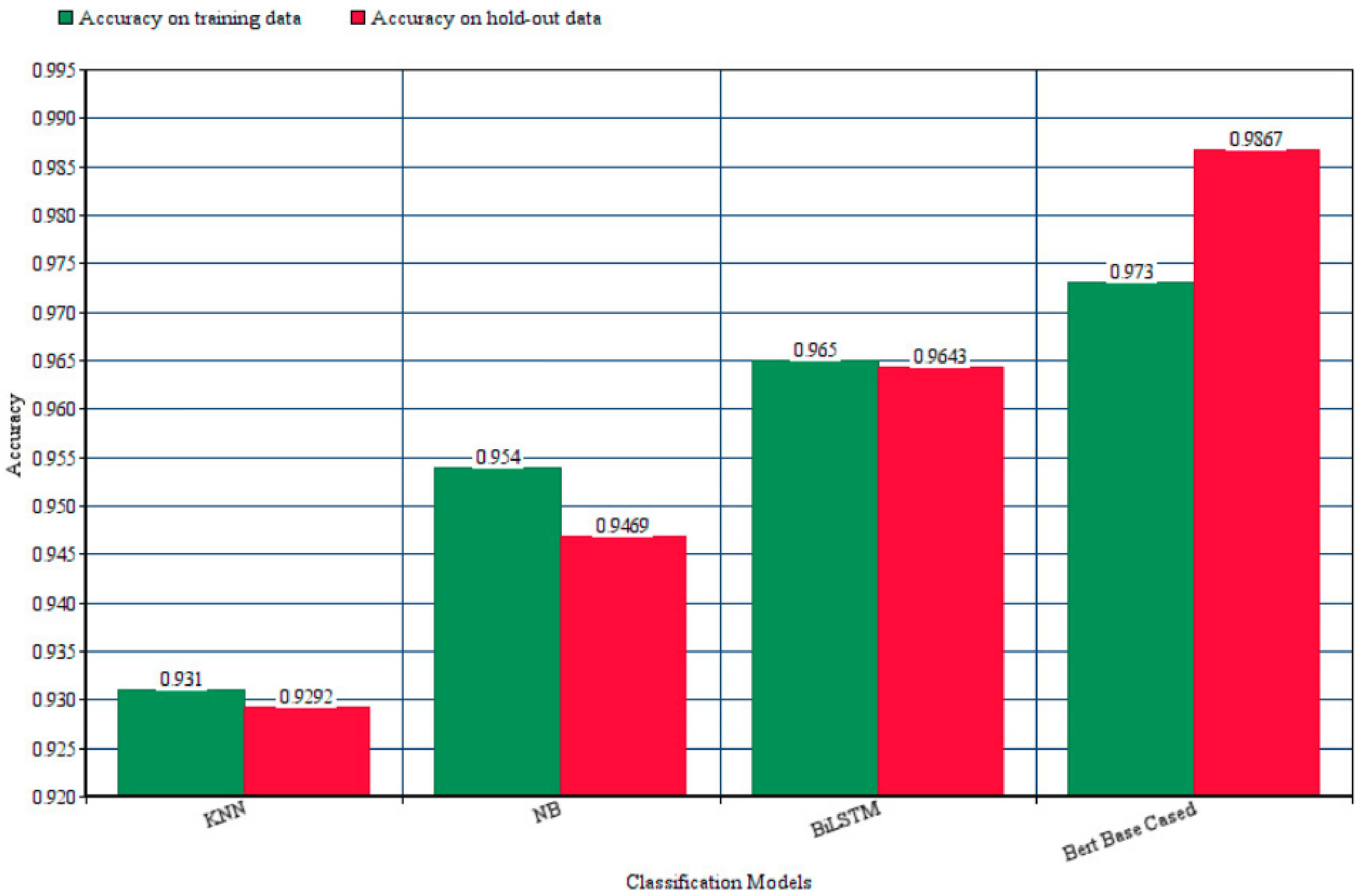
Spam Email Detection Using Deep Learning Techniques (example study)

Model	Accuracy	F1 Score
KNN	0.9292	0.9081
NB	0.9469	0.9459
BiLSTM	0.9643	0.9600
Bert Base Cased	**0.9867**	**0.9866**

Testing Result on Test Data (Using Different Algorithms)

Source: [Research Paper]

Spam Email Detection Using Deep Learning Techniques (example study)



Accuracy Comparison of Algorithms

Source: [Research Paper]