CMSC 170: Introduction to Artificial intelligence

Week 05: Spam Filtering using a Naïve Bayes Classifier

John O-Neil V. Geronimo
Institute of Computer Science
University of the Philippines Los Baños

Content

I. Background
II. Implementing Spam Filter using Naïve Bayes Classifier

Content

I. Background
II. Implementing Spam Filter using Naïve Bayes Classifier

Background

A **Spam Filter** classifies messages (commonly emails) to remove incoming spam. This is the technology that prevents you from seeing spam in your inbox folder as much as possible; the spam instead goes to the Spam folder.

Background

A *common way* to implement a spam filter is to *represent all received emails as in a* ***bag-of-words*** and compute the probability of a message being spam given the words that make it up.

Background

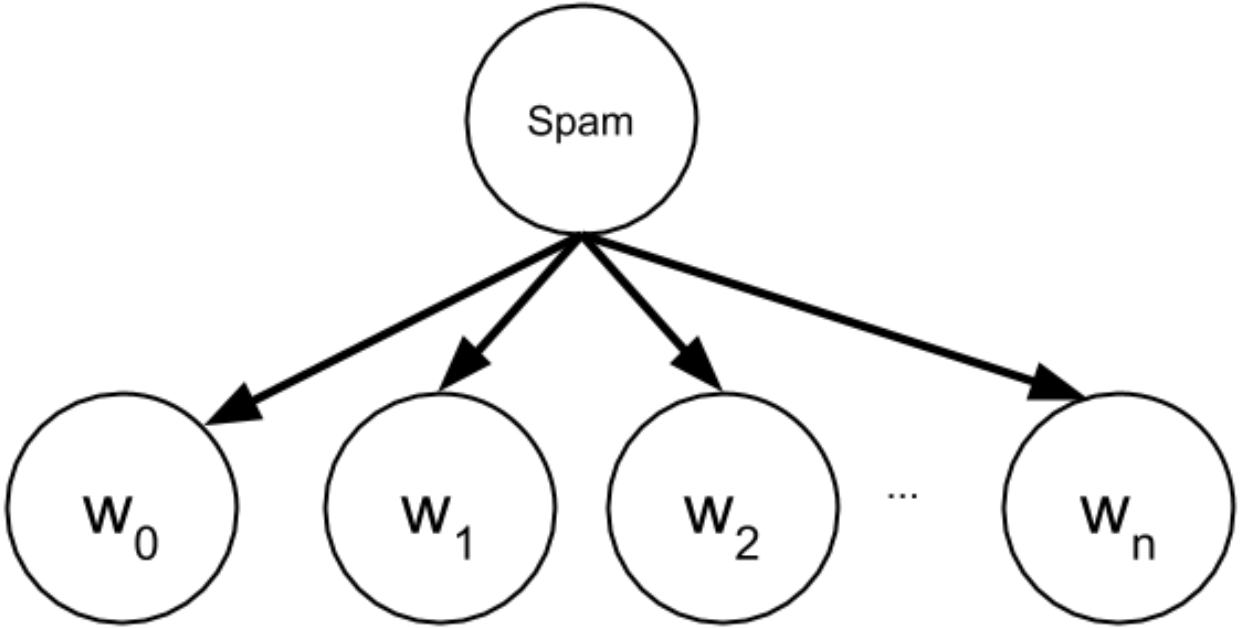
One of the ways to solve the spam filtering problem is via the **Naive Bayes classifier** technique. This technique is based on the Bayesian theorem which states that,

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Background

This theorem describes the ***probability of an event based on prior knowledge*** of conditions relating to the event. The naivety of the Naive Bayes classifier comes from its strong assumption that the conditions are ***independent*** of each other, that is, the features/attributes describing the event.

Background



Background

Naïve Bayes Theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Spam Filtering Using Naïve Bayes

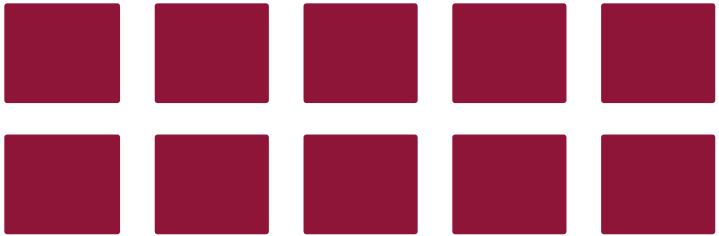
$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

Background

We are initially given a data set of both Spam and Ham messages. The data set is the set of messages, and the target labels indicate whether a message is Spam/Ham. Since we already know whether a message is Spam/Ham, the target labels are already given, thus making spam filtering a **supervised learning problem**.

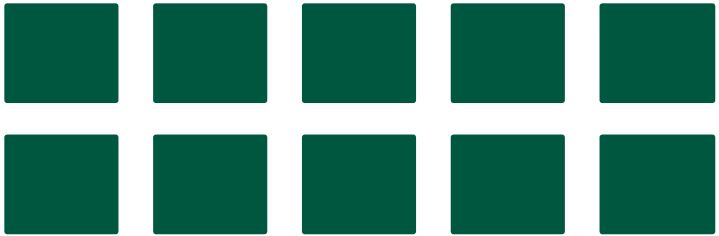
Background

Spam Bag-of-Words



Index	Word	Frequency

Ham Bag-of-Words



Index	Word	Frequency

Formula:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

How do we compute for P(Spam)?

$$P(Spam) = \frac{count(Spam)}{count(Spam \cup Ham)}$$

$$P(Ham) = \frac{count(Ham)}{count(Spam \cup Ham)}$$

$$= 1 - P(Spam)$$

Formula:

$$\boldsymbol{P(Spam|message)} = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\,(Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$

How do we compute for P(message|Spam)?

$$message = w_0 w_1 w_2 \ldots w_n$$

$$P(message|Spam) = P(w_0 w_1 w_2 \ldots w_n|Spam)$$
$$= P(w_0|Spam)P(w_1|Spam) \ldots P(w_0|Spam)$$

Formula:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\ (Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$
- $P(message|Spam) = P(w_0|Spam) \ldots P(w_n|Spam)$

How do we compute for P(w|Spam)?

$$P(w_n|Spam) = \frac{P(w, Spam)}{P(Spam)}$$

$$= \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$$

Formula:

$$\boldsymbol{P(Spam|message)} = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\ (Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$
- $P(message\,|Spam)$
  $= P(w_0|Spam) \dots P(w_n|Spam)$
- $P(w_n|Spam)$
  $= \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$

How do we compute for P(message|Ham)?

$$message = w_0 w_1 w_2 \dots w_n$$

$$P(message|Ham) = P(w_0 w_1 w_2 \dots w_n|Ham)$$
$$= P(w_0|Ham)P(w_1|Ham) \dots P(w_0|Ham)$$

Formula:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\ (Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$
- $P(message\,|Spam)$
  $= P(w_0|Spam) \dots P(w_n|Spam)$
- $P(w_n|Spam)$
  $= \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$
- $P(message\,|Ham)$
  $= P(w_0|Ham) \dots P(w_n|Ham)$

How do we compute for P(w|Ham)?

$$P(w_n|Ham) = \frac{P(w, Ham)}{P(Ham)}$$

$$= \frac{count(w\ in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)}$$

Formula:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\ (Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$
- $P(message|Spam)$
  $= P(w_0|Spam) \dots P(w_n|Spam)$
- $P(w_n|Spam)$
  $= \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$
- $P(message|Ham)$
  $= P(w_0|Ham) \dots P(w_n|Ham)$
- $P(w_n|Ham)$
  $= \frac{count(w\ in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)}$

How do we compute for P(message)?

$$P(message) = P(message|Spam)P(Spam) + P(message|Ham)P(Ham)$$

Formula:

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

- $P(Spam) = \frac{count\ (Spam)}{count(Spam \cup Ham)}$
- $P(Ham) = 1 - P(Spam)$
- $P(message|Spam) = P(w_0|Spam) \dots P(w_n|Spam)$
- $P(w_n|Spam) = \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$
- $P(message|Ham) = P(w_0|Ham) \dots P(w_n|Ham)$
- $P(w_n|Ham) = \frac{count(w\ in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)}$
- $P(message) = P(message|Spam)P(Spam) + P(message|Ham)P(Ham)$

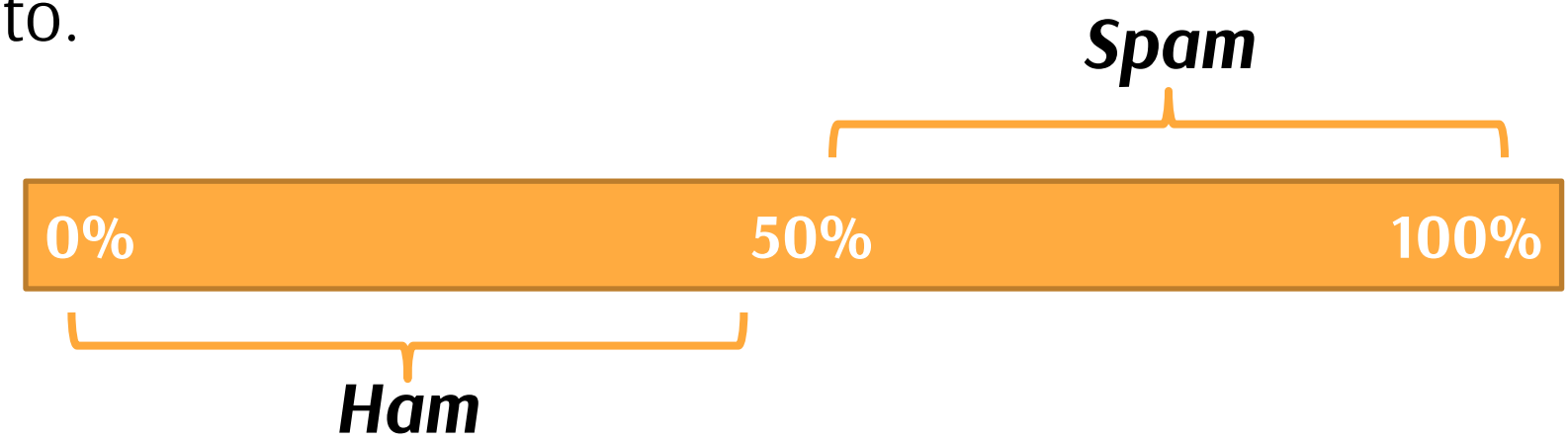
Putting it all together,

$$P(S|m) = \frac{P(w_0|S)P(w_1|S) \dots P(w_n|S)P(S)}{P(w_0|S)P(w_1|S) \dots P(w_n|S)P(S) + P(w_0|H)P(w_1|H) \dots P(w_n|H)P(H)}$$

where m = message, S = Spam, and H = Ham

Background

Lastly, a **threshold** must be set in order to determine which among the two (2) classes does a given message fall into.



Content

I. Background
II. Implementing Spam Filter using Naïve Bayes Classifier

Spam Filter

Spam Dataset

- You will love this product!
- I used this cream and it removed my acne!!!
- Buy now!

Ham Dataset

- Lucky I am in love with my bestfriend
- You are still the one that I love
- Could I love you anymore?

Spam Filter

Spam Bag-of-Words

Index	Word	Frequency
0	you	1
1	will	1
2	love	1
3	this	2
4	product	1
5	i	1
6	used	1
7	cream	1

Index	Word	Frequency
8	and	1
9	it	1
10	removed	1
11	my	1
12	acne	1
13	buy	1
14	now	1

TNOW:

16

DS:

15

Spam Filter

Ham Bag-of-Words

Index	Word	Frequency
0	lucky	1
1	i	3
2	am	1
3	in	1
4	love	3
5	with	1
6	my	1
7	bestfriend	1

Index	Word	Frequency
8	you	2
9	are	1
10	still	1
11	the	1
12	one	1
13	that	1
14	could	1
15	anymore	1

TNOW:

21

DS:

16

Spam Filter

Is this message ham or spam?

I love you.

Spam Filter

Compute for P(Spam):

Since we have three emails in the spam dataset and a total of six messages for both spam and ham dataset

$$P(Spam) = \frac{count(Spam)}{count(Spam \cup Ham)}$$

$$P(Spam) = \frac{3}{6} = \mathbf{0.5}$$

Spam Filter

Compute for P(Ham):

Since we have three emails in the ham dataset and a total of six messages for both spam and ham dataset

$$P(Ham) = \frac{count(Ham)}{count(Spam \cup Ham)}$$

$$P(Ham) = \frac{3}{6} = \mathbf{0.5}$$

Spam Filter

Compute for P(message|Spam):

Remember that we can compute P(message|spam) using this formula:

$$P(message|Spam) = P(w_0|Spam)P(w_1|Spam) \dots P(w_n|Spam)$$

P(w_n|Spam) can be computed using this formula:

$$P(w_n|Spam) = \frac{count(w\ in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$$

Spam Filter

Compute for P(message|Spam):

$$P(\text{"}i\text{"}|Spam) = \frac{count(\text{"}i\text{" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$$

$$P(\text{"}love\text{"}|Spam) = \frac{count(\text{"}love\text{" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$$

$$P(\text{"}you\text{"}|Spam) = \frac{count(\text{"}you\text{" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)}$$

Spam Filter

Spam Bag-of-Words

Index	Word	Frequency
0	you	1
1	will	1
2	love	1
3	this	2
4	product	1
5	i	1
6	used	1
7	cream	1

Index	Word	Frequency
8	and	1
9	it	1
10	removed	1
11	my	1
12	acne	1
13	buy	1
14	now	1

TNOW:

16

DS:

15

Spam Filter

Compute for P(message|Spam):

$$P(\text{"i"}|Spam) = \frac{count(\text{"i" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)} = \frac{1}{16}$$

$$P(\text{"love"}|Spam) = \frac{count(\text{"love" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)} = \frac{1}{16}$$

$$P(\text{"you"}|Spam) = \frac{count(\text{"you" } in\ Spam)}{count(total\ no\ of\ words\ in\ Spam)} = \frac{1}{16}$$

Spam Filter

Compute for P(message|Spam):

$$P(message|Spam) = P(w_0|Spam)P(w_1|Spam) \dots P(w_n|Spam)$$

$$= P("i"|Spam)P("love"|Spam)P("you"|Spam)$$

$$= \frac{1}{16} * \frac{1}{16} * \frac{1}{16}$$

$$= \mathbf{\frac{1}{4096}}$$

Spam Filter

Compute for P(message|Ham):

Remember that we can compute P(message|Ham) using this formula:

$$P(message|Ham) = P(w_0|Ham)P(w_1|Ham) \dots P(w_n|Ham)$$

P(w_n|Ham) can be computed using this formula:

$$P(w_n|Ham) = \frac{count(w\ in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)}$$

Spam Filter

Compute for P(message|Ham):

$$P("i"|Ham) = \frac{count("i" \ in \ Ham)}{count(total \ no \ of \ words \ in \ Ham)}$$

$$P("love"|Ham) = \frac{count("love" \ in \ Ham)}{count(total \ no \ of \ words \ in \ Ham)}$$

$$P("you"|Ham) = \frac{count("you" \ in \ Ham)}{count(total \ no \ of \ words \ in \ Ham)}$$

Spam Filter

Ham Bag-of-Words

Index	Word	Frequency
0	lucky	1
1	i	3
2	am	1
3	in	1
4	love	3
5	with	1
6	my	1
7	bestfriend	1

Index	Word	Frequency
8	you	2
9	are	1
10	still	1
11	the	1
12	one	1
13	that	1
14	could	1
15	anymore	1

TNOW:

21

DS:

16

Spam Filter

Compute for P(message|Spam):

$$P(\text{"i"}|Ham) = \frac{count(\text{"i" } in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)} = \frac{3}{21}$$

$$P(\text{"love"}|Ham) = \frac{count(\text{"love" } in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)} = \frac{3}{21}$$

$$P(\text{"you"}|Ham) = \frac{count(\text{"you" } in\ Ham)}{count(total\ no\ of\ words\ in\ Ham)} = \frac{2}{21}$$

Spam Filter

Compute for P(message|Ham):

$$P(message|Ham) = P(w_0|Ham)P(w_1|Ham) \dots P(w_n|Ham)$$

$$= P("i"|Ham)P("love"|Ham)P("you"|Ham)$$

$$= \frac{3}{21} * \frac{3}{21} * \frac{2}{21}$$

$$= \mathbf{\frac{2}{1029}}$$

Spam Filter

Compute for P(Spam|message):

$$P(Spam|message) = \frac{P(message|Spam)P(Spam)}{P(message)}$$

$$= \frac{P(message|Spam)P(Spam)}{P(message|Spam)P(Spam) + P(message|Ham)P(Ham)}$$

$$= \frac{\frac{1}{4096}(0.5)}{\frac{1}{4096}(0.5) + \frac{2}{1029}(0.5)} = 0.1115931027 = 11.16\%$$

Spam Filter

Since P(Spam|message) = 11.16%

The message “I love you.”
is classified as a **ham** message!

Content

I. Background
II. Implementing Spam Filter using Naïve Bayes Classifier

Keep safe!