# Naive Bayesian vs. Keyword-Based Spam Filtering

by I. Androutsopoulos, J. Koutsias, K. V.Chandrinos and C. D. Spyropoulos

Adrian Scoică (as2270@cam.ac.uk)

February 12, 2013

イロト 不得下 イヨト イヨト

1/13



#### Problem Insights









#### Automatic Spam Detection

Is it a straightforward classification problem?

#### Automatic Spam Detection

#### Is it a straightforward classification problem?

4	Move to Inbox     More ~				
Get A	Free Experian Check - www.experian.co.uk - Get The UK's No. 1 Credit Checkl Free Experian® 30 Day Trial Now On.				
In case you want to buy Viagra from the Nigerian Prince					
*	Adrian Scoică ≺adriansc@rosedu.org> to Adrian ⊡				
	Hello,				
	I just wanted to warn you not to be tempted to buy Viagra from Nigerian princes. They are up to nothing but scams!!!				
	Hope this helps				
	Cheers, Adrian.				
	Click hara to Banlu or Forward				

Intention is more important than content...

#### Automatic Spam Detection

#### Is it a straightforward classification problem?

•	C More -	
Clocking In	Machines - www.ClockingSystem	s.co.uk - Clocking In Machines From Only £140 Free Del. Over £50 Ph: 0845 00
		Delete all spam messages now (messages t
🗆 🛧 💌	Adrian Scoicā	In case you want to buy Viagra from the Nigerian Prince - Hello, I just w
□ ☆ ∞	Buy_Vigara Today	Purchase Levtira & Vigara - ** Best Products! ** Vigara - 0.89\$ Levtira - 1.5
	Cotswold Outdoor News	Up to half price sale now on. Save on Rab, Salomon and more Cotsi
	ACM CareerNews	ACM CareerNews Alert for Tuesday, February 5, 2013 - February 5, 2013 -

... but language is a good predictor.

#### Building a Corpus

Authors compiled the publicly-available  $\ensuremath{\text{PU1}}$  corpus:

- 618 legitimate messages
- 481 spam messages
- Real emails

#### Building a Corpus

Authors compiled the publicly-available **PU1** corpus:

- 618 legitimate messages
- 481 spam messages

**Real** emails, but **encrypted** due to privacy issues:

```
From: spammer@spamcompany.comSubject: 1 \ 2 \ 3 \ 4To: spamtarget@provider.com5 \ 6 \ 7 \ 1 \ 2 \ 4 \ 8 \ 9 \ 3 \ 4Subject: Get rich now !\Rightarrow
Subject: Get rich now !
Click here to get rich ! Try it now !
```

 $\Rightarrow$  Paper focuses on **word-features** only.

Documents are modelled with vectors

 $\vec{x}_{Doc} = \langle x_1, x_2, x_3, \dots x_n \rangle$ 

Documents are modelled with vectors  

$$\vec{x}_{Doc} = \langle x_1, x_2, x_3, ... x_n \rangle$$
  
using only binary, word-only attributes  $x_i = \begin{cases} 1 & word_i \in Doc \\ 0 & otherwise \end{cases}$ 

Documents are modelled with vectors  $\vec{x}_{Doc} = \langle x_1, x_2, x_3, ... x_n \rangle$ using only <u>binary</u>, <u>word-only</u> attributes  $x_i = \begin{cases} 1 & word_i \in Doc \\ 0 & otherwise \end{cases}$ which have the highest <u>mutual information</u> with the class variable, *C* 

$$\sum_{x \in \{0,1\}, c \in \{\text{spam}, \text{legitimate}\}} P(X = x, C = c) \log \frac{P(X = x, C = c)}{P(X = x)P(C = c)}$$

Documents are modelled with vectors

 $\vec{x}_{Doc} = \langle x_1, x_2, x_3, ... x_n \rangle$ using only <u>binary</u>, <u>word-only</u> attributes  $x_i = \begin{cases} 1 & word_i \in Doc \\ 0 & otherwise \end{cases}$ which have the highest <u>mutual information</u> with the class variable, *C* 

$$\sum_{x \in \{0,1\}, c \in \{\text{spam}, \text{legitimate}\}} P(X = x, C = c) \log \frac{P(X = x, C = c)}{P(X = x)P(C = c)}$$

Using the Naive Bayes assumption, we can compute:

$$P(spam | \vec{x}_{Doc}) = \frac{P(spam) * \prod_{i=1}^{n} P(X_i = x_i | spam)}{\sum_{c \in \{spam, legitimate\}} P(C = c) \prod_{i=1}^{n} P(X_i = x_i | C = c)}$$

In spam filtering, **precision** if more important than **recall**.

Blocking a legitimate message is  $\lambda$  times more expensive than letting a spam message pass, so when **do** we block?

In spam filtering, **precision** if more important than **recall**.

Blocking a legitimate message is  $\lambda$  times more expensive than letting a spam message pass, so when **do** we block?  $\frac{P(spam | \vec{x}_{Doc})}{P(legitimate | \vec{x}_{Doc})} > \lambda \Leftrightarrow P(spam | \vec{x}_{Doc}) > t = \frac{\lambda}{\lambda + 1}$ 

The paper uses threshold *t* to analyze three scenarios:

$$t = 0.999$$
  $t = 0.9$   $t = 0.5$ 

In spam filtering, **precision** if more important than **recall**.

Blocking a legitimate message is  $\lambda$  times more expensive than letting a spam message pass, so when **do** we block?  $\frac{P(spam | \vec{x}_{Doc})}{P(legitimate | \vec{x}_{Doc})} > \lambda \Leftrightarrow P(spam | \vec{x}_{Doc}) > t = \frac{\lambda}{\lambda + 1}$ 

The paper uses threshold t to analyze three scenarios:

t = 0.999		t = 0.9		t = 0.5
cautious	$\leftarrow$	average	$\rightarrow$	aggressive

How do we evaluate performance?

In spam filtering, **precision** if more important than **recall**.

Blocking a legitimate message is  $\lambda$  times more expensive than letting a spam message pass, so when **do** we block?  $\frac{P(spam | \vec{x}_{Doc})}{P(legitimate | \vec{x}_{Doc})} > \lambda \Leftrightarrow P(spam | \vec{x}_{Doc}) > t = \frac{\lambda}{\lambda + 1}$ 

The paper uses threshold t to analyze three scenarios:

t = 0.999 t = 0.9 t = 0.5cautious  $\leftarrow$  average  $\rightarrow$  aggressive

How do we evaluate performance?  $WErr = \frac{\lambda n_{L \to S} + n_{S \to L}}{\lambda N_{L} + N_{S}}$ Comparing to the baseline (no filter):  $TCR = \frac{WErr^{baseline}}{WErr} = \frac{N_{S}}{\lambda n_{L \to S} + n_{S \to L}}$ 

# Results

Filter used	t	No. attr.	TCR
(a) NB (bare)	0.5	50	4.90
(b) NB (stop-list)		50	4.95
(c) NB (lemmatizer)		100	4.29
(d) NB (stop-list + lemmatizer)		100	4.53
Keyword patterns		-	2.01

# Results

Filter used	t	No. attr.	TCR
(a) NB (bare)	0.5	50	4.90
(b) NB (stop-list)		50	4.95
(c) NB (lemmatizer)		100	4.29
(d) NB (stop-list + lemmatizer)		100	4.53
Keyword patterns		-	2.01
(a) NB (bare)	0.9	100	2.20
(b) NB (stop-list)		150	2.28
(c) NB (lemmatizer)		100	2.83
(d) NB (stop-list + lemmatizer)		100	2.56
Keyword patterns		-	1.40

# Results

Filter used	t	No. attr.	TCR	
(a) NB (bare)	0.5	50	4.90	-
(b) NB (stop-list)		50	4.95	
(c) NB (lemmatizer)		100	4.29	
(d) NB (stop-list + lemmatizer)		100	4.53	
Keyword patterns		-	2.01	
(a) NB (bare)	0.9	100	2.20	-
(b) NB (stop-list)		150	2.28	
(c) NB (lemmatizer)		100	2.83	
(d) NB (stop-list + lemmatizer)		100	2.56	
Keyword patterns		-	1.40	
(a) NB (bare)	0.999	700	0.15	-
(b) NB (stop-list)		700	0.15	
(c) NB (lemmatizer)		50	0.11	
(d) NB (stop-list + lemmatizer)		600	0.11	
Keyword patterns	•		0.04	æ

8/13

#### Results

How many attributes do we need?



*TCR* of the filters for t = 0.5 ( $\lambda = 1$ )

#### Results

How many attributes do we need?



#### Results

How many attributes do we need?



*TCR* of the filters for t = 0.999 ( $\lambda = 999$ )

#### Results

How much training data is enough?



3

#### Conclusions

Paper key points and contributions:

- Introduces cost-sensitive evaluation.
- Proves effectiveness of automatic spam filtering.
- Proves stop-lists don't improve performance with MI attribute selection.
- Shows classifiers are trainable even with small amounts of data.

Thank you!

イロト 不得下 イヨト イヨト 二日