Spam or Not Spam
That is the question

Jakub Białek

4 maja 2006

Zwięzła definicja spamu

Spam to nadmiar informacji zbędnych dla odbiorcy przekazu.

Definicji poszerzona

Spam jest określany również:

- 'UCE' – unsolicited commercial email (niezamawiany komercyjny email)

Definicji poszerzona

Spam jest określany również:

- 'UCE' – unsolicited commercial email (niezamawiany komercyjny email)
- 'UBE' – unsolicited bulk email (niezamawiany wielokrotny email)

Definicji poszerzona

Spam jest określany również:

- 'UCE' – unsolicited commercial email (niezamawiany komercyjny email)
- 'UBE' – unsolicited bulk email (niezamawiany wielokrotny email)

Spam to takie informacje lub przesyłki, których odbiorca sobie nie zażyczył ani wcześniej na nie się nie zgodził. Najczęściej do niczego mu niepotrzebne, powodujące nieekonomiczne wykorzystanie użytych do przesyłki zasobów, często wywołujące irytację.

Standardowa definicji spamu

Elektroniczna wiadomość, jest spamem, jeżeli:

1. treść i kontekst wiadomości są niezależne od tożsamości odbiorcy, ponieważ ta sama treść może być skierowana do wielu innych potencjalnych odbiorców,

Standardowa definicji spamu

Elektroniczna wiadomość, jest spamem, jeżeli:

1. treść i kontekst wiadomości są niezależne od tożsamości odbiorcy, ponieważ ta sama treść może być skierowana do wielu innych potencjalnych odbiorców,
2. jej odbiorca nie wyraził uprzedniej, możliwej do weryfikacji, zamierzonej, wyraźnej i zawsze odwoływalnej zgody na otrzymanie tej wiadomości,

Standardowa definicji spamu

Elektroniczna wiadomość, jest spamem, jeżeli:

1. treść i kontekst wiadomości są niezależne od tożsamości odbiorcy, ponieważ ta sama treść może być skierowana do wielu innych potencjalnych odbiorców,
2. jej odbiorca nie wyraził uprzedniej, możliwej do weryfikacji, zamierzonej, wyraźnej i zawsze odwoływalnej zgody na otrzymanie tej wiadomości,
3. treść wiadomości daje odbiorcy podstawę do przypuszczeń, iż nadawca wskutek jej wysłania może odnieść korzyści nieproporcjonalne w stosunku do korzyści odbiorcy wynikających z jej odebrania.

Wybrane typy spamu

Wsród spamu można m.in. wyróżnić następujące typu:

- hoax – czyli fałszywka, plotka rozpowszechniana przez kogoś dla uciechy; głupi dowcip, który bardzo wielu ludziom zabiera dużo czasu,

Wybrane typy spamu

Wsród spamu można m.in. wyróżnić następujące typu:

- hoax – czyli fałszywka, plotka rozpowszechniana przez kogoś dla uciechy; głupi dowcip, który bardzo wielu ludziom zabiera dużo czasu,
- zombie

Wybrane typy spamu

Wsród spamu można m.in. wyróżnić następujące typu:

- hoax – czyli fałszywka, plotka rozpowszechniana przez kogoś dla uciechy; głupi dowcip, który bardzo wielu ludziom zabiera dużo czasu,
- zombie
- nigeryjski szwindel

Wybrane typy spamu

Wsród spamu można m.in. wyróżnić następujące typu:

- hoax – czyli fałszywka, plotka rozpowszechniana przez kogoś dla uciechy; głupi dowcip, który bardzo wielu ludziom zabiera dużo czasu,
- zombie
- nigeryjski szwindel
- joe-job

Trochę statystyk

Bill Gates otrzymuje 4 miliony wiadomości spamowych rocznie (11/2004), to:

Trochę statystyk

Bill Gates otrzymuje 4 miliony wiadomości spamowych rocznie (11/2004), to:

- 11 tys. wiadomości dziennie

Trochę statystyk

Bill Gates otrzymuje 4 miliony wiadomości spamowych rocznie (11/2004), to:

- 11 tys. wiadomości dziennie
- 456 wiadomości na godzinę

Trochę statystyk

Bill Gates otrzymuje 4 miliony wiadomości spamowych rocznie (11/2004), to:

- 11 tys. wiadomości dziennie
- 456 wiadomości na godzinę

Jef Poskanzer zarządzający domeną (i firmą) acme.com otrzymuje:

- o.k. 1 miliona wiadomości spamowych dziennie

Metody niestatystyczne walki ze spamem

Do zwalczania spamu stosowane są często następujące metody:

- black list (czarne listy),
- white list (białe listy),
- Vipul (znakowanie listów),
- Challenge-Response,
- filters that Fight Back (filtry rewanżujące się).

Metody statystyczne i 'uczące się'

Lepsze rezultaty w walce z spamem można uzyskać stosując metody, które 'uczą się' m.in.:

- decision trees (drzewa decyzyjne),

Metody statystyczne i 'uczące się'

Lepsze rezultaty w walce z spamem można uzyskać stosując metody, które 'uczą się' m.in.:

- decision trees (drzewa decyzyjne),
- neural network (sieci neuronowe),

Metody statystyczne i 'uczące się'

Lepsze rezultaty w walce z spamem można uzyskać stosując metody, które 'uczą się' m.in.:

- decision trees (drzewa decyzyjne),
- neural network (sieci neuronowe),
- k nearest neighbors (k najbliższych sąsiadów),

Metody statystyczne i 'uczące się'

Lepsze rezultaty w walce z spamem można uzyskać stosując metody, które 'uczą się' m.in.:

- decision trees (drzewa decyzyjne),
- neural network (sieci neuronowe),
- k nearest neighbors (k najbliższych sąsiadów),
- naive bayes,

Metody statystyczne i 'uczące się'

Lepsze rezultaty w walce z spamem można uzyskać stosując metody, które 'uczą się' m.in.:

- decision trees (drzewa decyzyjne),
- neural network (sieci neuronowe),
- k nearest neighbors (k najbliższych sąsiadów),
- naive bayes,
- support vector machines.

Twierdzenie Bayesa

Twierdzenie

$$P(T|X) = \frac{P(X|T)P(T)}{P(X)} = \frac{P(X|T)P(T)}{P(X|T)P(T)+P(X|T^c)P(T^c)}$$

X – obserwacja,
T – jedna z hipotez,
$P(X)$ – obserwowane prawdopodobieństwo X,
$P(X|T)$ – prawdopodobieństwo, że X nastąpi według teorii T,
$P(T)$ – prawdopodobieństwo, że teoria T jest prawdziwa,
$P(T|X)$ – prawdopodobieństwo, że teoria T jest prawdziwa, jeśli zaobserwowano X.

Dowód twierdzenia Bayesa

Niech X będzie pewnym zdarzeniem, T pewną teorią. Załóżmy, że $P(X) \neq 0$ i $P(T) \neq 0$, mamy:

- $P(X \cap T) = P(T)P(X|T)$

Dowód twierdzenia Bayesa

Niech X będzie pewnym zdarzeniem, T pewną teorią. Załóżmy, że $P(X) \neq 0$ i $P(T) \neq 0$, mamy:

- $P(X \cap T) = P(T)P(X|T)$
- $P(X \cap T) = P(X)P(T|X)$

Dowód twierdzenia Bayesa

Niech X będzie pewnym zdarzeniem, T pewną teorią. Załóżmy, że $P(X) \neq 0$ i $P(T) \neq 0$, mamy:

- $P(X \cap T) = P(T)P(X|T)$
- $P(X \cap T) = P(X)P(T|X)$
- $P(T)P(X|T) = P(X)P(T|X)$

Dowód twierdzenia Bayesa

Niech X będzie pewnym zdarzeniem, T pewną teorią. Załóżmy, że $P(X) \neq 0$ i $P(T) \neq 0$, mamy:

- $P(X \cap T) = P(T)P(X|T)$
- $P(X \cap T) = P(X)P(T|X)$
- $P(T)P(X|T) = P(X)P(T|X)$
- $P(T|X) = \frac{P(T)P(X|T)}{P(X)}$

Dowód twierdzenia Bayesa

Niech X będzie pewnym zdarzeniem, T pewną teorią. Załóżmy, że $P(X) \neq 0$ i $P(T) \neq 0$, mamy:

- $P(X \cap T) = P(T)P(X|T)$
- $P(X \cap T) = P(X)P(T|X)$
- $P(T)P(X|T) = P(X)P(T|X)$
- $P(T|X) = \frac{P(T)P(X|T)}{P(X)}$
- $P(X) = P(X \cap T) + P(X \cap T^c)$

Przykład

Weźmy dwie równie prawdopodobne hipotezy: T_1, T_2, $P(T_i) = \frac{1}{2}$. Zbieramy dane – zaobserwowaliśmy X. Jeśli zaobserwowanie X jest bardziej prawdopodobne dla T_1, $P(X|T_1) > P(X|T_2)$, to T_1 jest (teraz, pod dokonaniu badania) bardziej prawdopodobne.

Klasy

W celu klasyfikacji wiadomości, wprowadzamy dwie grupy (klasy):

- S – dla oznaczenia spamu,

Klasy

W celu klasyfikacji wiadomości, wprowadzamy dwie grupy (klasy):

- S – dla oznaczenia spamu,
- L – dla oznaczenia prawidłowej wiadomości email.

Klasy

W celu klasyfikacji wiadomości, wprowadzamy dwie grupy (klasy):

- S – dla oznaczenia spamu,
- L – dla oznaczenia prawidłowej wiadomości email.

Interesuje nas wyznaczenie, do której z grup należy dana wiadomość (email). Tak, więc korzystając z twierdzenia Bayesa mamy:

Klasyfikacja

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)} = \frac{P(x|c)P(c)}{P(x|S)P(S)+P(x|L)P(L)}$$

gdzie $c \in \{L, S\}$.

Opis klasyfikacji

Zwykle znane są wartości dla $P(c)$ (prawdopodobieństwo, że losowa wiadomość jest z tej klasy) oraz $P(c|x)$ (np. możemy założyć, że słowo BUY nigdy nie wystąpi w prawidłowej wiadomości email, wtedy prawdopodobieństwo otrzymanie wiadomości zawierającej słowo BUY z kategorii L jest równe zero). Toteż korzystając z wcześniejszego wzoru można wyznaczyć prawdopodobieństwo z jakim dana wiadomość należy do jednej z klas L, S.

Reguła klasyfikacji

Korzystając z wcześniejszego wzoru, daną wiadomość klasyfikujemy według następującej zasady:

- jeśli $P(S|x) > P(L|x)$ (czyli prawdopodobieństwo a-posteriori, że x jest spamem jest większe niż prawdopodobieństwo a-posteriori, że x nie jest spamem) wtedy x klasyfikujemy jako spam (S),

Reguła klasyfikacji

Korzystając z wcześniejszego wzoru, daną wiadomość klasyfikujemy według następującej zasady:

- jeśli $P(S|x) > P(L|x)$ (czyli prawdopodobieństwo a-posteriori, że x jest spamem jest większe niż prawdopodobieństwo a-posteriori, że x nie jest spamem) wtedy x klasyfikujemy jako spam (S),
- w przeciwnym wypadku klasyfikujemy x jako prawidłową wiadomość (L).

Reguła klasyfikacji

Korzystając z wcześniejszego wzoru, daną wiadomość klasyfikujemy według następującej zasady:

- jeśli $P(S|x) > P(L|x)$ (czyli prawdopodobieństwo a-posteriori, że x jest spamem jest większe niż prawdopodobieństwo a-posteriori, że x nie jest spamem) wtedy x klasyfikujemy jako spam (S),
- w przeciwnym wypadku klasyfikujemy x jako prawidłową wiadomość (L).

Powyższa reguła jest znana jako MAP (maximum a-posteriori probability).

Przekształcenie reguły

Używając twierdzenia Bayesa, przekształcamy wcześniejszą reguła klasyfikacji do postaci:

Jeśli $\frac{P(x|S)}{P(x|L)} > \frac{P(L)}{P(S)}$ sklasyfikuj x jako spam, wpp x sklasyfikuj jako prawidłowa wiadomość (nie spam).

Dla wygody oznaczmy przez $\Lambda(x)$ wyrażenie: $\frac{P(x|S)}{P(x|L)}$.

Dalsze oznaczenia

Niech $\mathcal{L}(c_1, c_2)$ oznacza koszt (ryzyko) nieprawidłowego sklasyfikowania obiektu klasy c_1 jako należący do klasy c_2.

Dalsze oznaczenia

Niech $\mathcal{L}(c_1, c_2)$ oznacza koszt (ryzyko) nieprawidłowego sklasyfikowania obiektu klasy c_1 jako należący do klasy c_2. W naszym przypadku oczywiście $\mathcal{L}(S, S) = \mathcal{L}(L, L) = 0$.

Dalsze oznaczenia

Niech $\mathcal{L}(c_1, c_2)$ oznacza koszt (ryzyko) nieprawidłowego sklasyfikowania obiektu klasy c_1 jako należący do klasy c_2. W naszym przypadku oczywiście $\mathcal{L}(S, S) = \mathcal{L}(L, L) = 0$. Wtedy spodziewane ryzyko zaklasyfikowania wiadomości x do klasy c wynosi:

$$R(c|x) = \mathcal{L}(S, c)P(S|x) + \mathcal{L}(L, c)P(L|x)$$

Zasada klasyfikacji Bayesa

Oczywiście chcemy, aby nasza klasyfikacja miała małe spodziewane ryzyko dla dowolnej wiadomości, więc naturalnie jest użyć następującej reguły:

Zasada klasyfikacji Bayesa

Jeśli $R(S|x) < R(L(x))$ sklasyfikuj x jako spam, wpp sklasyfikuj jako prawidłową wiadomość.

Powyższa zasada minimalizuje całościowe spodziewane ryzyko (średnie ryzyko).

Ostateczna wersja zasady klasyfikacji

Ostatecznie możemy zapisać naszą zasada klasyfikacji jako:

$$\Lambda(x) \underset{L}{\overset{S}{\gtrless}} \lambda \frac{P(L)}{P(S)}$$

gdzie $\lambda = \frac{\mathcal{L}(L,S)}{\mathcal{L}(S,L)}$ oznacza jak niebezpieczne jest sklasyfikowanie prawidłowej wiadomości jako spam.

Ostateczna wersja zasady klasyfikacji

Ostatecznie możemy zapisać naszą zasada klasyfikacji jako:

$$\Lambda(x) \underset{L}{\overset{S}{\gtrless}} \lambda \frac{P(L)}{P(S)}$$

gdzie $\lambda = \frac{\mathcal{L}(L,S)}{\mathcal{L}(S,L)}$ oznacza jak niebezpieczne jest sklasyfikowanie prawidłowej wiadomości jako spam.
Aby filtr miał sens interesuje nas, aby takie sytuacje w ogóle nie zachodziły lub zdarzały się bardzo rzadko, odwrotna sytuacja jest mniej groźna i dopuszczalne jest jej wystąpienie (wzięcie spamu za prawidłową wiadomość).

Praktyczny algorytm

Aby skonstruować klasyfikator Bayesowski musimy móc wyznaczyć prawdopodobieństwa: $P(x|c)$ i $P(c)$ dla każdych x i c. Dokładna wartość tych prawdopodobieństw nie jest znana, lecz można jest przybliżać na podstawie danych treningowych.

Praktyczny algorytm

Aby skonstruować klasyfikator Bayesowski musimy móc wyznaczyć prawdopodobieństwa: $P(x|c)$ i $P(c)$ dla każdych x i c. Dokładna wartość tych prawdopodobieństw nie jest znana, lecz można jest przybliżać na podstawie danych treningowych.
Dla przykładu:

- $P(S)$ możemy przybliżać przez iloraz liczby wiadomości będących spamem, do liczby wszystkich wiadomości

Praktyczny algorytm c.d...

Trudniejsze do przybliżenia jest $P(x|c)$, które zależy od tego, jak wybierzemy wektor cech x dla wiadomości m. Weźmy najprostszy przypadek, gdy wektor cech jest pojedyńczym binarnym atrybutem, oznaczającym wystąpienie słowa „w” w wiadomości.

Praktyczny algorytm c.d...

Trudniejsze do przybliżenia jest $P(x|c)$, które zależy od tego, jak wybierzemy wektor cech x dla wiadomości m. Weźmy najprostszy przypadek, gdy wektor cech jest pojedyńczym binarnym atrybutem, oznaczającym wystąpienie słowa „w" w wiadomości.
Wtedy wektor cech x_w wynosi:

- 1 jeśli słowo w występuje w wiadomości,
- 0 jeśli słowo w nie występuje w wiadomości.

Praktyczny algorytm c.d...

Trudniejsze do przybliżenia jest $P(x|c)$, które zależy od tego, jak wybierzemy wektor cech x dla wiadomości m. Weźmy najprostszy przypadek, gdy wektor cech jest pojedyńczym binarnym atrybutem, oznaczającym wystąpienie słowa „w" w wiadomości.
Wtedy wektor cech x_w wynosi:

- 1 jeśli słowo w występuje w wiadomości,
- 0 jeśli słowo w nie występuje w wiadomości.

W takim przypadku przybliżamy prawdopodobieństwo:

$$P(x_w = 1|S) \approx \frac{\#\text{treningowych próbek spamu zawierających słowo } w}{\#\text{wszystkich treningowych próbek spamu}}$$

Praktyczny algorytm c.d...

Podsumowując mamy następujący algorytm:

Praktyczny algorytm c.d...

Podsumowując mamy następujący algorytm:

- trening
  1. oblicz przybliżenie dla $P(c)$, $P(x_w = 1|c)$, $P(x_w = 0|c)$ dla $c \in \{S, L\}$ na podstawie danych treningowych,

Praktyczny algorytm c.d...

Podsumowując mamy następujący algorytm:

- trening
  1. oblicz przybliżenie dla $P(c)$, $P(x_w = 1|c)$, $P(x_w = 0|c)$ dla $c \in \{S, L\}$ na podstawie danych treningowych,
  2. oblicz $P(c|x_w = 0)$, $P(c|x_w = 1)$ używając twierdzenia Bayesa,

Praktyczny algorytm c.d...

Podsumowując mamy następujący algorytm:

- trening
  1. oblicz przybliżenie dla $P(c)$, $P(x_w = 1|c)$, $P(x_w = 0|c)$ dla $c \in \{S, L\}$ na podstawie danych treningowych,
  2. oblicz $P(c|x_w = 0)$, $P(c|x_w = 1)$ używając twierdzenia Bayesa,
  3. oblicz $\Lambda(x_w)$ dla $x_w = 0, 1$, oblicz $\lambda \frac{P(L)}{P(S)}$, zapamiętaj te 3 wartości.

Praktyczny algorytm c.d...

Podsumowując mamy następujący algorytm:

- trening
  1. oblicz przybliżenie dla $P(c)$, $P(x_w = 1|c)$, $P(x_w = 0|c)$ dla $c \in \{S, L\}$ na podstawie danych treningowych,
  2. oblicz $P(c|x_w = 0)$, $P(c|x_w = 1)$ używając twierdzenia Bayesa,
  3. oblicz $\Lambda(x_w)$ dla $x_w = 0, 1$, oblicz $\lambda \frac{P(L)}{P(S)}$, zapamiętaj te 3 wartości.
- klasyfikacja
  1. dla wiadomości m wyznacz wektor cech x_w, odnajdź zapisaną wartość dla $\Lambda(x_w)$ i użyj zasady klasyfikacji do zdecydowania, do której kategorii należy wiadomość m.

Ulepszenie algorytmu

Taka klasyfikacja nie będzie dobra, gdyż opiera się
na występowaniu lub nie tylko jednego słowa w wiadomości.

Ulepszenie algorytmu

Taka klasyfikacja nie będzie dobra, gdyż opiera się na występowaniu lub nie tylko jednego słowa w wiadomości. Aby polepszyć algorytm, wektor cech powinien zawierać więcej danych (atrybutów). Wybierzmy kilka słów: $w_1, w_2, \ldots, w_m$ i zdefiniujmy wektor cech dla wiadomości m jako:

$$x = (x_1, x_2, \ldots, x_m)$$

gdzie x_i jest równe 1, jeśli słowo w_i występuje w wiadomości, wpp 0.

Ulepszenie algorytmu c.d...

Jeśli wykorzystamy poprzednie algorytm, będziemy musieli obliczyć i zapamiętać wartości $\Lambda(x)$ dla każdego możliwego x (a tych jest 2^m), jest to niepraktyczne.

Ulepszenie algorytmu c.d...

Jeśli wykorzystamy poprzednie algorytm, będziemy musieli obliczyć i zapamiętać wartości $\Lambda(x)$ dla każdego możliwego x (a tych jest 2^m), jest to niepraktyczne.
Toteż zakładamy, że składowe wektora x są niezależne w każdej klasie, czyli wystąpienie słowa w_i w wiadomości, nie wpływa na prawdopodobieństwa wystąpienia innych słów.
Nie jest to dobre założenie, lecz umożliwia obliczenie wymaganych prawdopodobieństw bez konieczności pamiętania dużej liczby danych:

$$P(x|c) = \prod_{i=1}^{m} P(x_i|c) \qquad \Lambda(x) = \prod_{i=1}^{m} \Lambda_i(x_i)$$

Naiwna klasyfikacja Bayesowska

Słowo „naiwna” wynika z przyjętego założenia o niezależności składowych wektora cech. W praktyce algorytm spisuje się dobrze i jest jednym z najpopularniejszych rozwiązań używanych w filtrach antyspamowych.

Naiwna klasyfikacja Bayesowska

Słowo „naiwna” wynika z przyjętego założenia o niezależności składowych wektora cech. W praktyce algorytm spisuje się dobrze i jest jednym z najpopularniejszych rozwiązań używanych w filtrach antyspamowych.
Podsumowując algorytm wygląda następująco:

Naiwna klasyfikacja Bayesowska

Słowo „naiwna" wynika z przyjętego założenia o niezależności składowych wektora cech. W praktyce algorytm spisuje się dobrze i jest jednym z najpopularniejszych rozwiązań używanych w filtrach antyspamowych.
Podsumowując algorytm wygląda następująco:

- trening
  1. dla każdego w_i oblicz i zapamiętaj $\Lambda_i(x_i)$ dla $x_i = 0, 1$, oblicz i zapamiętaj $\lambda \frac{P(L)}{P(S)}$.

Naiwna klasyfikacja Bayesowska

Słowo „naiwna" wynika z przyjętego założenia o niezależności składowych wektora cech. W praktyce algorytm spisuje się dobrze i jest jednym z najpopularniejszych rozwiązań używanych w filtrach antyspamowych.
Podsumowując algorytm wygląda następująco:

- trening
  1. dla każdego w_i oblicz i zapamiętaj $\Lambda_i(x_i)$ dla $x_i = 0, 1$, oblicz i zapamiętaj $\lambda \frac{P(L)}{P(S)}$.
- klasyfikacja
  1. wyznacz x, oblicz $\Lambda(x)$ przez pomnożenie zapamiętanych wartości $\Lambda_i(x_i)$, użyj zasady klasyfikacji.

Jaki słowa wybrać?

Ciągle należy zdecydować, jakie słowa wybrać do wektora cech. Istnieje kilka rozwiązań:

Jaki słowa wybrać?

Ciągle należy zdecydować, jakie słowa wybrać do wektora cech. Istnieje kilka rozwiązań:

- wszystkie słowa występujące w treningowych wiadomościach,

Jaki słowa wybrać?

Ciągle należy zdecydować, jakie słowa wybrać do wektora cech. Istnieje kilka rozwiązań:

- wszystkie słowa występujące w treningowych wiadomościach,
- jeśli liczba słów jest zbyt duża można:

Jaki słowa wybrać?

Ciągle należy zdecydować, jakie słowa wybrać do wektora cech. Istnieje kilka rozwiązań:

- wszystkie słowa występujące w treningowych wiadomościach,
- jeśli liczba słów jest zbyt duża można:
  - odrzucić najpopularniejsze słowa,

Jaki słowa wybrać?

Ciągle należy zdecydować, jakie słowa wybrać do wektora cech. Istnieje kilka rozwiązań:

- wszystkie słowa występujące w treningowych wiadomościach,
- jeśli liczba słów jest zbyt duża można:
  - odrzucić najpopularniejsze słowa,
  - odrzucić najrzadsze słowa

Wybrane metody walki

Zostaną porównane ze sobą następujące metody walki ze spamem:

- naiwna klasyfikacja Bayesa,
- k najbliższych sąsiadów,
- artificial neural networks (multilayer perceptron),
- support vector machine classification (metoda wektorów nośnych).

Kilka słów o metodach

K nearest neighbors – aby móc sparametryzować liczbę wiadomości poprawnych uznanych za spam, została wprowadzona nastepująca zasada klasyfikacji (zasada l/k):

Jeśli l lub więcej wiadomości spośród k sąsiadów x jest spamem, to sklasyfikuj x jako spam, wpp sklasyfikuj jako poprawną wiadomość.

Kilka słów o metodach

K nearest neighbors – aby móc sparametryzować liczbę wiadomości poprawnych uznanych za spam, została wprowadzona nastepująca zasada klasyfikacji (zasada l/k):

Jeśli l lub więcej wiadomości spośród k sąsiadów x jest spamem, to sklasyfikuj x jako spam, wpp sklasyfikuj jako poprawną wiadomość.

Support vector machines – działa według tej samej idei co perceptron, różnica polega na tym, że nie szukamy jakieś (dowolnej) hiperprzestrzenii dzielącej (separującej), lecz bardzo specyficznej optymalnej płaszczyzny podziału w wielowymiarowej przestrzeni przekształconych zmiennych (maximal margin separating hyperplane); metoda uwzględnia skomplikowane, nieliniowe zależności między badanym cechami.

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,
- zbiór jest podzielony na 10 część (9 zostanie wykorzystanych jako treningowe a 1 jako testowa),

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,
- zbiór jest podzielony na 10 część (9 zostanie wykorzystanych jako treningowe a 1 jako testowa),
- wiadomości w zbiorze podlegały wcześniejszemu przetworzeniu:

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,
- zbiór jest podzielony na 10 część (9 zostanie wykorzystanych jako treningowe a 1 jako testowa),
- wiadomości w zbiorze podlegały wcześniejszemu przetworzeniu:
  - odrzucono załączniki, znaczniki HTML i nagłówek listu (poza tematem),

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,
- zbiór jest podzielony na 10 część (9 zostanie wykorzystanych jako treningowe a 1 jako testowa),
- wiadomości w zbiorze podlegały wcześniejszemu przetworzeniu:
  - odrzucono załączniki, znaczniki HTML i nagłówek listu (poza tematem),
  - słowa zostały zakodowane jako liczby,

Wykorzystane dane

- wykorzystane dane pochodzą ze zbioru PU1, który utworzył Ion Androutsopoulos,
- zbiór zawiera 1099 wiadomości, z czego 481 jest spamem,
- zbiór jest podzielony na 10 część (9 zostanie wykorzystanych jako treningowe a 1 jako testowa),
- wiadomości w zbiorze podlegały wcześniejszemu przetworzeniu:
  - odrzucono załączniki, znaczniki HTML i nagłówek listu (poza tematem),
  - słowa zostały zakodowane jako liczby,
- każdy wektor cech zawiera 21700 atrybutów (tyle różnych słów jest w wiadomościach).

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),
2. każde słowo zostaje zastąpione formą bazową (tematem, bez końcówek),

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),
2. każde słowo zostaje zastąpione formą bazową (tematem, bez końcówek),
3. "stop-list", 100 najpopularniejszych słów zostaje usuniętych z każdej wiadomości,

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),
2. każde słowo zostaje zastąpione formą bazową (tematem, bez końcówek),
3. "stop-list", 100 najpopularniejszych słów zostaje usuniętych z każdej wiadomości,
4. połączenie dwóch poprzednich wersji.

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),
2. każde słowo zostaje zastąpione formą bazową (tematem, bez końcówek),
3. "stop-list", 100 najpopularniejszych słów zostaje usuniętych z każdej wiadomości,
4. połączenie dwóch poprzednich wersji.

Testy wstępne wykazały, że najlepsze rezultaty dają dane w wersji 4 i na nich tylko dalsze testy są przeprowadzane.

Wersje danych

Cały zbiór zawiera cztery wersje każdej wiadomości:

1. oryginalną (słowa są zakodowane przez liczby),
2. każde słowo zostaje zastąpione formą bazową (tematem, bez końcówek),
3. "stop-list", 100 najpopularniejszych słów zostaje usuniętych z każdej wiadomości,
4. połączenie dwóch poprzednich wersji.

Testy wstępne wykazały, że najlepsze rezultaty dają dane w wersji 4 i na nich tylko dalsze testy są przeprowadzane. Należy zauważyć, że ten zbiór nie odzwierciedla rzeczywistej sytuacji, gdyż najbardziej miarodajne wyznaczniki spamu: nagłówek i znaczniki HTML nie są brane pod uwagę. Można więc uznać, że przy ich uwzględnieniu wyniki byłby lepsze.

Oznaczenia

- $N = 1099$ oznacza liczbę wszystkich wiadomości,
- $N_{S \to L}$ liczbę wiadomości będących spamem uznanych za „niespam”,
- $N_{L \to S}$ liczbę wiadomości poprawnych uznanych za spam,
- $N_S = 481$ liczba wiadomości będących spamem,
- $N_L = 618$ liczba prawidłowych wiadomości („niespam”).

Miary efektowności

Współczynnik błędu określamy następująco:

$$E = \frac{N_{S\to L} + N_{L\to S}}{N}$$

precyzje:

$$P = 1 - E$$

legitimate fallout:

$$F_L = \frac{N_{L\to S}}{N_L}$$

spam fallout:

$$F_S = \frac{N_{S\to L}}{N_S}$$

Miary efektywności c.d...

Należy zauważyć, że błąd i precyzja powinny być względne do przypadku gdy klasyfikacja nie występuje. Bez klasyfikacji mamy trywialnie zagwarantowane, że $\frac{N_L}{N}$ w naszym przypadku będzie zawsze większe od 50%.

Miary efektywności c.d...

Należy zauważyć, że błąd i precyzja powinny być względne do przypadku gdy klasyfikacja nie występuje. Bez klasyfikacji mamy trywialnie zagwarantowane, że $\frac{N_L}{N}$ w naszym przypadku będzie zawsze większe od 50%.
Tak więc wprowadzamy miarę wzrostu precyzji:

$$G = \frac{P}{N_L/N} = \frac{N - N_{S \to L} - N_{L \to S}}{N_L}$$

Jakość metod

Wcześniej przedstawione metody na opisanych danych, osiągły następujące wyniki:

Metoda	$N_{L\to S}$	$N_{S\to L}$	P	F_L	F_S	G
Naive Bayes ($\lambda = 1$)	0	138	87.4%	0.0%	28.7%	1.56
k-NN ($k = 51$)	68	33	90.8%	11.0%	6.9%	1.61
Perceptron	8	8	98.5%	1.3%	1.7%	1.75
SVM	10	11	98.1%	1.6%	2.3%	1.74

Jakość metod c.d..

Aby zmniejszyć (do zera) liczbę wiadomości poprawnych uznawanych za spam, zmodyfikowano odpowiednie współczynniki. Jako, że perceptron nie daje możliwości takich ustawień, nie jest on dalej testowany.

Metoda	$N_{L\to S}$	$N_{S\to L}$	P	F_L	F_S	G
Naive Bayes ($\lambda = 8$)	0	140	87.3%	0.0%	29.1%	1.55
l/k-NN ($k = 51, l = 35$)	0	337	69.3%	0.0%	70.0%	1.23
SVM soft margin(0.3)	0	101	90.8%	0.0%	21.0%	1.61

Połączona klasyfikacja

Niech f i g oznaczają dwa filtry spamowe, oba o bardzo mały prawdopodobieństwie wzięcia wiadomości prawidłowej za spam. Można stworzyć filtr o lepszej precyzji stosując poniższą regułę klasyfikacji:

Połączona klasyfikacja

Niech f i g oznaczają dwa filtry spamowe, oba o bardzo mały prawdopodobieństwie wzięcia wiadomości prawidłowej za spam. Można stworzyć filtr o lepszej precyzji stosując poniższą regułę klasyfikacji:

Sklasyfikuj wiadomość x jako spam jeśli f lub g sklasyfikowało x jako spam, wpp (jeśli $f(x) = g(x) = L$) sklasyfikuj x jako poprawną wiadomość.

Połączona klasyfikacja

Niech f i g oznaczają dwa filtry spamowe, oba o bardzo mały prawdopodobieństwie wzięcia wiadomości prawidłowej za spam. Można stworzyć filtr o lepszej precyzji stosując poniższą regułę klasyfikacji:

Sklasyfikuj wiadomość x jako spam jeśli f lub g sklasyfikowało x jako spam, wpp (jeśli $f(x) = g(x) = L$) sklasyfikuj x jako poprawną wiadomość.

Metoda	$N_{L\to S}$	$N_{S\to L}$	P	F_L	F_S	G
N.B $\cup$ SVM s.m.	0	61	94.4%	0.0%	12.7%	1.68

Połączona klasyfikacja c.d...

Niech h będzie klasyfikatorem z wysoką precyzją (np. perceptron lub SVM), możemy użyć następującej klasyfikacji:

Jeśli $f(x) = g(x) = c$ klasyfikujemy x do klasy c, jeśli klasyfikatory f i g dają inne wyniki, nie klasyfikujemy natychmiastowo x jako spam, lecz wykorzystujemy h jako klasyfikator (z racji jego wysokiej precyzji) i klasyfikujemy x według niego.

Połączona klasyfikacja c.d...

Niech h będzie klasyfikatorem z wysoką precyzją (np. perceptron lub SVM), możemy użyć następującej klasyfikacji:

Jeśli $f(x) = g(x) = c$ klasyfikujemy x do klasy c, jeśli klasyfikatory f i g dają inne wyniki, nie klasyfikujemy natychmiastowo x jako spam, lecz wykorzystujemy h jako klasyfikator (z racji jego wysokiej precyzji) i klasyfikujemy x według niego.

Metoda	$N_{L\to S}$	$N_{S\to L}$	P	F_L	F_S	G
2-z-3	0	62	94.4%	0.0%	12.9%	1.68

Podsumowanie

Naiwna klasyfikacja Bayesa mimo naiwnego założenia o niezależności poszczególnych słów od siebie, jest jedną z najlepszych metod filtracji spamu. Obecnie jest szeroko wykorzystywana w wielu produktach, z racji wysokiej skuteczności. Często też jest jedną z linii obrony przeciw spamowi, współpracującą z innymi zabezpieczeniami antyspamowymi.

Bibliografia

- Machine Learning Techniques in Spam Filtering, Konstantin Tretyakov, Institute of Computer Science, University of Tartu.
- Spam or Not Spam? That is the question, Ravi Kiran and Indriyati Atmosukarto.
- A Neural Network Classifier for Junk E-Mail, Ian Stuart, Sung-Hyuk Cha, and Charles Tappert.
- http://nospam-pl.net/

Koniec

Dziękuję za uwagę!