Text Classification
A Get-To-Know Introduction

Weiwei Cheng
University of Marburg / Deutsche Bank
www.chengweiwei.com
Advanced Search

Use this form to automatically construct your query. (Alternatively, you can type search operators directly into the search box.)

Find tweets based on...

Words
- All of these words
- Any of these words
- None of these words

Attitudes
- With positive attitude :) □
- With negative attitude :( □
- Asking a question ? □

try it at http://search.twitter.com/advanced
Formally...

Training (or learning):
Input: a set of $m$ labeled documents $(x_1, y_1), \ldots, (x_m, y_m)$
Output: a learned classifier $f: x \rightarrow y$

Testing (or predicting):
Input: a document $x$
Output: a class $y$ from some fixed set of labels $y_1, \ldots, y_K$
Why we need computers?

If we have *enough* brains, we can solve *all* problems!

Wrong!
Can human do this?

Human brain has limited a capacity.
Can you remember all the books from the library?

Is the computer only for our “dirty work”?  

Human perception is fuzzy.  
Human intuition is (more than often) wrong.  
Human reasoning is by nature not robust against illusion.  
You like to watch magic, don’t you?
Examples
Weiwei Cheng got a double bachelor's degree in Computer Science and Business Administration at Zhengzhou University, China. After graduation, he went to Germany and studied from 2005 to 2007 in Data and Knowledge Engineering Department at Otto-von-Guericke University Magdeburg. In 2006 he was awarded the Outstanding Foreign Student Scholarship by the Ministry of Education and Cultural Affairs of Saxony-Anhalt. He received his Master of Science degree with highest honor in November 2007 and was awarded as the Best Graduate of Year in October 2008. Since December 2007, he worked in Knowledge Engineering & Bioinformatics Lab at University of Marburg as a research assistant and Ph.D. candidate under the supervision of Prof. Eyke Hüllermeier.
Representing document with bag-of-words

A lady was picking through the frozen turkeys at the supermarket, but couldn’t find one big enough for her family. She asked a stock boy, “Do these turkeys get any bigger?” The stock boy replied, “No madam, they’re dead.”

... ignores the ordering of the words.
Multinomial Naïve Bayes classifier

- A probabilistic learning method;
- Based on Bayes’ theorem;
- State-of-the-art;
- Simple, in terms of implementation and use.

Great and simple idea with fancy name
Some basic probability

<table>
<thead>
<tr>
<th>Notation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P(A)$</td>
<td>$P($someone works in DB$)$</td>
</tr>
<tr>
<td>$P(B)$</td>
<td>$P($someone is smart$)$</td>
</tr>
<tr>
<td>$P(A&amp;B)$</td>
<td>$P($someone works in DB and she/he is smart$)$</td>
</tr>
<tr>
<td>$P(A\cup B)$</td>
<td>$P($someone works in DB or she/he is smart$)$</td>
</tr>
<tr>
<td>$P(A</td>
<td>B)$</td>
</tr>
</tbody>
</table>

If $A$ and $B$ are independent, e.g.,

A: I wear yellow shoes today,
B: Today it rains,

then $P(A\&B) = P(B)P(A)$. 
Bayes' theorem

The chance that a smart guy works in DB.

The chance that a guy in DB is smart.

The chance someone works in DB.

The chance someone is smart.

\[ P(A|B) = \frac{P(B|A)P(A)}{P(B)} \]

A: someone works in DB
B: someone is smart
There is a drug test. It will correctly identify a drug user as testing positive 99% of the time, and will correctly identify a non-user as testing negative 99% of the time. Let's assume some company decides to test its employees for drug use, and it is known that 0.5% of the population actually use the drug. What is the probability that, given a positive drug test, an employee is actually a drug user?

\[
P(D|+) = \frac{P(+|D)P(D)}{P(+)} = \frac{P(+|D)P(D)}{P(+|D)P(D) + P(+|N)P(N)} = \frac{0.99 \times 0.005}{0.99 \times 0.005 + 0.01 \times 0.995} = 0.3322.
\]

~ 33%
We want to know which class can lead to the highest \( P(y|x) \).

By Bayes’ theorem

\[
P(y|x) = \frac{P(x|y)P(y)}{P(x)} \quad \propto P(x|y)P(y)
\]

With the bag-of-words representation

\[
P(y)P(x|y) = P(y)P(x_1|y)P(x_2|y) \ldots P(x_n|y)
\]

To choose a class that maximize this.

The probability of class \( y \), e.g., the chance an email being spam.

The probability of word \( x_2 \) found in class \( y \), e.g., the chance of word “pharmacy” seen in a spam email.

The probability of the email’s class (spam/not spam) given its text.
Take-away message

Text classification finds the patterns in documents, reduces the human effort.

How do we represent the text in the computer?
*bag-of-words*

Which classifier?
*Naive Bayes. Key idea: Bayes’ theorem*
Accuracy

Number of labeled documents

Banko & Brill, 2001
That is all from my side 😊
Thank you all for being here!

Time for discussion!

"I think you should be more explicit here in step two."

www.chengweiwei.com