Administrative corrections

- The prerequisite is apparently CS 1331.
- This will be changed to CS 3510, and my recommendation is to wait until you've had this class.
- Informally: probability, algorithms, programming
- If you don't have sufficient preparation, there's only so much we can do to help you...

Sentiment analysis

Pang and Lee define sentiment analysis broadly:

- Making a decision for a particular document
  - "is it positive or negative?"
  - "how positive is it?"
- Ordering a set of of texts
  - "rank these reviews by how positive they are"
- Giving a single label to an entire collection
  - "where on the scale between liberal and conservative do the writings of this author lie?"
- Categorizing the relationship between two enti- ties based on textual evidence
  - "does A approve of Bs actions?"

Sentiment analysis

Pang and Lee define sentiment analysis broadly:

- Making a decision for a particular document
  - **“is it positive or negative?”** (polarity classification)
  - “how positive is it?”
- Ordering a set of of texts
  - “rank these reviews by how positive they are”
- Giving a single label to an entire collection
  - “where on the scale between liberal and conservative do the writings of this author lie?”
- Categorizing the relationship between two enti- ties based on textual evidence
  - “does A approve of Bs actions?”

Sentiment analysis

Darin [ATL Urbanist] @atlurbanist · 11h
Great sunset tonight, behind the clouds in Downtown Atlanta #**atlanta** #sunset #weloveatl pic.twitter.com/JsMZBliIqw
2 3 View photo

NWS Atlanta @NWSAtlanta · 19h · ••• More
RT @LoznickaCBS46: Ugly skies over Buford, GA at this moment. Sent in from CBS46 viewer. cc: @NWSAtlanta #**Atlanta** pic.twitter.com/uzy2LUZnrC
3 View photo

Spam detection

spam

/spam/

noun

1. irrelevant or inappropriate messages sent on the Internet to a large number of recipients.
2. *trademark*
   a canned meat product made mainly from ham.

verb

1. send the same message indiscriminately to (large numbers of recipients) on the Internet.

Free in Atlanta @HelpAtlanta · 11m · ••• More

#**atlanta** How to raise credit score 28 points in 2 months? bit.ly/yWBeB5

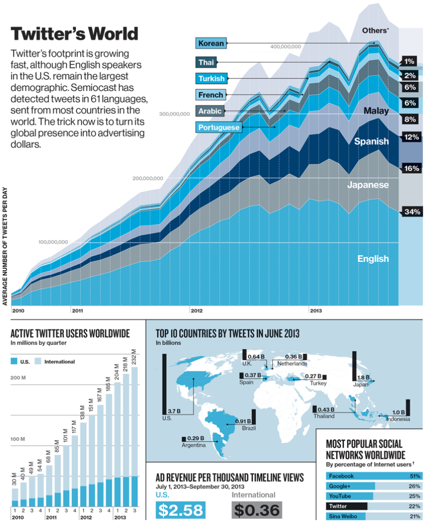
AtlBizChron @AtlBizChron · 19h · ••• More

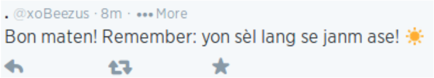
UPDATE: City did not err in demolition permit for historic building in King district bizj.us/116g2z #**Atlanta** #MLK

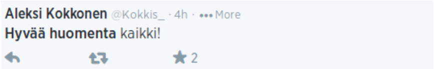
1

View summary

Language classification







The bag-of-words representation

$$\mathbf{w}_1 = \{\text{great}, \text{sunset}, \text{tonight}, \ldots\}$$

$$\mathbf{w}_2 = \{\text{ugly}, \text{skies}, \text{buford}, \ldots\}$$

The bag-of-words representation

$$\mathbf{w}_1 = \{\text{great}, \text{sunset}, \text{tonight}, \ldots\}$$

$$\mathbf{w}_2 = \{\text{ugly}, \text{skies}, \text{buford}, \ldots\}$$

	aardvark	abacus	...	behind	...	buford	...	clouds	...	great	...	ugly	...
$\mathbf{x}_1^T =$	0	0	$0\ldots0$	1	$0\ldots0$	0	$0\ldots0$	0	$0\ldots0$	1	$0\ldots0$	0	0 .
$\mathbf{x}_2^T =$	0	0	$0\ldots0$	0	$0\ldots0$	1	$0\ldots0$	0	$0\ldots0$	0	$0\ldots0$	1	0 .

The bag-of-words representation

$\mathbf{w}_1 = \{\text{great}, \text{sunset}, \text{tonight}, \ldots\}$ $\mathbf{w}_2 = \{\text{ugly}, \text{skies}, \text{buford}, \ldots\}$

	aardvark	abacus	...	behind	...	buford	...	clouds	...	great	...	ugly	...
$\mathbf{x}_1^T =$	0	0	0 . . . 0	1	0 . . . 0	0	0 . . . 0	0	0 . . . 0	1	0 . . . 0	0	0 .
$\mathbf{x}_2^T =$	0	0	0 . . . 0	0	0 . . . 0	1	0 . . . 0	0	0 . . . 0	0	0 . . . 0	1	0 .

$$\mathbf{x}_1 = \{\text{great} : 1, \text{sunset} : 1, \text{tonight} : 1, \ldots\}$$
$$\mathbf{x}_2 = \{\text{ugly} : 1, \text{skies} : 1, \text{buford} : 1, \ldots\}$$

Feature functions

Suppose $y \in \mathcal{Y} = \{\text{pos}, \text{neg}\}$. Then,

$$\mathbf{f}(\mathbf{x}, y = \text{pos}) = [\mathbf{x}^\top, \mathbf{0}^\top]^\top$$
$$\mathbf{f}(\mathbf{x}, y = \text{neg}) = [\mathbf{0}^\top, \mathbf{x}^\top]^\top$$

Feature functions

Suppose $y \in \mathcal{Y} = \{\text{pos}, \text{neg}, \text{neut}\}$. Then,

$$\begin{aligned}
\mathbf{f}(\mathbf{x}, y = \text{pos}) &= [\mathbf{x}^\top, \mathbf{0}^\top, \mathbf{0}^\top \quad]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neg}) &= [\mathbf{0}^\top, \mathbf{x}^\top, \mathbf{0}^\top \quad]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neut}) &= [\mathbf{0}^\top, \mathbf{0}^\top, \mathbf{x}^\top \quad]^\top
\end{aligned}$$

Feature functions

Suppose $y \in \mathcal{Y} = \{\text{pos}, \text{neg}, \text{neut}\}$. Then,

$$
\begin{aligned}
\mathbf{f}(\mathbf{x}, y = \text{pos}) =& [\mathbf{x}^\top, \mathbf{0}^\top, \mathbf{0}^\top \quad]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neg}) =& [\mathbf{0}^\top, \mathbf{x}^\top, \mathbf{0}^\top \quad]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neut}) =& [\mathbf{0}^\top, \mathbf{0}^\top, \mathbf{x}^\top \quad]^\top
\end{aligned}
$$

The feature vector is composed of individual feature functions, e.g.:

$$
\begin{aligned}
f_{176}(\mathbf{x}, y) :=& x_{176} \times \delta(y = \text{pos}) \\
=& \delta(\text{great} \in \mathbf{w} \wedge y = \text{pos}) \\
f_{177}(\mathbf{x}, y) :=& x_{177} \times \delta(y = \text{pos}) \\
f_{10176}(\mathbf{x}, y) :=& x_{176} \times \delta(y = \text{neg}) \ldots
\end{aligned}
$$

Feature functions

Suppose $y \in \mathcal{Y} = \{\text{pos}, \text{neg}, \text{neut}\}$. Then,

$$\begin{aligned}
\mathbf{f}(\mathbf{x}, y = \text{pos}) =& [\mathbf{x}^\top, \mathbf{0}^\top, \mathbf{0}^\top, 1]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neg}) =& [\mathbf{0}^\top, \mathbf{x}^\top, \mathbf{0}^\top, 1]^\top \\
\mathbf{f}(\mathbf{x}, y = \text{neut}) =& [\mathbf{0}^\top, \mathbf{0}^\top, \mathbf{x}^\top, 1]^\top
\end{aligned}$$

The feature vector is composed of individual feature functions, e.g.:

$$\begin{aligned}
f_{176}(\mathbf{x}, y) :=& x_{176} \times \delta(y = \text{pos}) \\
=& \delta(\text{great} \in \mathbf{w} \wedge y = \text{pos}) \\
f_{177}(\mathbf{x}, y) :=& x_{177} \times \delta(y = \text{pos}) \\
f_{10176}(\mathbf{x}, y) :=& x_{176} \times \delta(y = \text{neg}) \ldots
\end{aligned}$$

We usually add an "offset" feature at the end of each vector.

Prediction by addition

- We can then define **weights** for each feature:

$$\{\langle \text{great}, \mathsf{pos}\rangle = 1, \langle \text{great}, \mathsf{neg}\rangle = -1, \langle \text{great}, \mathsf{neut}\rangle = 0,$$
$$\langle \text{ugly}, \mathsf{pos}\rangle = -1, \langle \text{ugly}, \mathsf{neg}\rangle = 1, \langle \text{ugly}, \mathsf{neut}\rangle = 0,$$
$$\langle \text{buford}, \mathsf{pos}\rangle = 0, \langle \text{buford}, \mathsf{neg}\rangle = 0, \langle \text{buford}, \mathsf{neut}\rangle = 0,$$
$$\ldots\}$$

Prediction by addition

- We can then define **weights** for each feature:

$$\begin{aligned}\boldsymbol{\theta} = \{&\langle \text{great}, \mathsf{pos} \rangle = 1, \langle \text{great}, \mathsf{neg} \rangle = -1, \langle \text{great}, \mathsf{neut} \rangle = 0,\\ &\langle \text{ugly}, \mathsf{pos} \rangle = -1, \langle \text{ugly}, \mathsf{neg} \rangle = 1, \langle \text{ugly}, \mathsf{neut} \rangle = 0,\\ &\langle \text{buford}, \mathsf{pos} \rangle = 0, \langle \text{buford}, \mathsf{neg} \rangle = 0, \langle \text{buford}, \mathsf{neut} \rangle = 0,\\ &\ldots\}\end{aligned}$$

- We can arrange these weights into a vector.

Prediction by addition

- We can then define **weights** for each feature:

$$\begin{aligned}\boldsymbol{\theta} = \{&\langle \text{great}, \mathsf{pos} \rangle = 1, \langle \text{great}, \mathsf{neg} \rangle = -1, \langle \text{great}, \mathsf{neut} \rangle = 0,\\ &\langle \text{ugly}, \mathsf{pos} \rangle = -1, \langle \text{ugly}, \mathsf{neg} \rangle = 1, \langle \text{ugly}, \mathsf{neut} \rangle = 0,\\ &\langle \text{buford}, \mathsf{pos} \rangle = 0, \langle \text{buford}, \mathsf{neg} \rangle = 0, \langle \text{buford}, \mathsf{neut} \rangle = 0,\\ &\ldots\}\end{aligned}$$

- We can arrange these weights into a vector.
- The **score** for any instance and label is equal to the sum of the weights for all features in the instance:

$$\begin{aligned}\psi_{y,\mathbf{x}} =& \sum_n \theta_n f_n(\mathbf{x}, y)\\ =& \boldsymbol{\theta}^\top \mathbf{f}(\mathbf{x}, y)\\ \hat{y} =& \arg\max_y \boldsymbol{\theta}^\top \mathbf{f}(\mathbf{x}, y)\end{aligned}$$

Where do we get the weights?

- Set them by hand

Where do we get the weights?

- Set them by hand
- Probability

Where do we get the weights?

- Set them by hand
- Probability
- Discriminative learning

Where do we get the weights?

- Set them by hand
- Probability
- Discriminative learning
- ...