Natural Language Processing

Info 159/259

David Bamman, UC Berkeley
Quizzes

• Take place in the first 10 minutes of class:
  • start at 3:40, end at 3:50

• We drop 3 lowest quizzes and homeworks total. For Q quizzes and H homeworks, we keep \((H+Q)-3\) highest scores.
Classification

A mapping $h$ from input data $x$ (drawn from instance space $\mathcal{X}$) to a label (or labels) $y$ from some enumerable output space $\mathcal{Y}$

$\mathcal{X} = \text{set of all documents}$
$\mathcal{Y} = \{\text{english, mandarin, greek, ...}\}$

$x = \text{a single document}$
$y = \text{ancient greek}$
Classification

\[ h(x) = y \]
\[ h(\text{μήνιν ἄειδε θεὰ}) = \text{ancient grc} \]
Classification

Let \( h(x) \) be the “true” mapping. We never know it. How do we find the best \( \hat{h}(x) \) to approximate it?

One option: rule based

If \( x \) has characters in unicode point range 0370-03FF:

\( \hat{h}(x) = \text{greek} \)
Classification

Supervised learning

Given training data in the form of \((x, y)\) pairs, learn \(\hat{h}(x)\)
# Text categorization problems

<table>
<thead>
<tr>
<th>task</th>
<th>$x$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>language ID</td>
<td>text</td>
<td>{english, mandarin, greek, \ldots}</td>
</tr>
<tr>
<td>spam classification</td>
<td>email</td>
<td>{spam, not spam}</td>
</tr>
<tr>
<td>authorship attribution</td>
<td>text</td>
<td>{jk rowling, james joyce, \ldots}</td>
</tr>
<tr>
<td>genre classification</td>
<td>novel</td>
<td>{detective, romance, gothic, \ldots}</td>
</tr>
<tr>
<td>sentiment analysis</td>
<td>text</td>
<td>{positive, negative, neutral, mixed}</td>
</tr>
</tbody>
</table>
Sentiment analysis

• Document-level SA: is the entire text positive or negative (or both/neither) with respect to an implicit target?

• Movie reviews [Pang et al. 2002, Turney 2002]
Training data

positive

“... is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius”

Roger Ebert, Apocalypse Now

negative

• “I hated this movie. Hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it.”

Roger Ebert, North
• Implicit signal: star ratings

• Either treat as ordinal regression problem (\{1, 2, 3, 4, 5\}) or binarize the labels into \{pos, neg\}
Sentiment analysis

• Is the text positive or negative (or both/neither) with respect to an explicit target within the text?

Sentiment analysis

• Political/product opinion mining
Twitter sentiment →

Job approval polls →


Figure 9: The sentiment ratio for obama (15-day window), and fraction of all Twitter messages containing obama (day-by-day, no smoothing), compared to election polls (2008) and job approval polls (2009).
Sentiment as tone

• No longer the speaker’s attitude with respect to some particular target, but rather the positive/negative tone that is evinced.
Sentiment as tone

“Once upon a time and a very good time it was there was a moocow coming down along the road and this moocow that was coming down along the road met a nicens little boy named baby tuckoo…”

http://www.matthewjockers.net/2014/06/05/a-novel-method-for-detecting-plot/
Sentiment Dictionaries

- LIWC (Linguistic Inquiry and Word Count, Pennebaker 2015)

<table>
<thead>
<tr>
<th>pos</th>
<th>neg</th>
</tr>
</thead>
<tbody>
<tr>
<td>unlimited</td>
<td>lag</td>
</tr>
<tr>
<td>prudent</td>
<td>contortions</td>
</tr>
<tr>
<td>superb</td>
<td>fright</td>
</tr>
<tr>
<td>closeness</td>
<td>lonely</td>
</tr>
<tr>
<td>impeccably</td>
<td>tenuously</td>
</tr>
<tr>
<td>fast-paced</td>
<td>plebeian</td>
</tr>
<tr>
<td>treat</td>
<td>mortification</td>
</tr>
<tr>
<td>destined</td>
<td>outrage</td>
</tr>
<tr>
<td>blessing</td>
<td>allegations</td>
</tr>
<tr>
<td>steadfastly</td>
<td>disoriented</td>
</tr>
</tbody>
</table>
Why is SA hard?

- Sentiment is a measure of a speaker’s private state, which is unobservable.

- Sometimes words are a good indicator of sentence (love, amazing, hate, terrible); many times it requires deep world + contextual knowledge.

“Valentine’s Day is being marketed as a Date Movie. I think it’s more of a First-Date Movie. If your date likes it, do not date that person again. And if you like it, there may not be a second date.”

Roger Ebert, Valentine’s Day
Classification

Supervised learning

Given training data in the form of $\langle x, y \rangle$ pairs, learn $\hat{h}(x)$

<table>
<thead>
<tr>
<th>$x$</th>
<th>$y$</th>
</tr>
</thead>
<tbody>
<tr>
<td>loved it!</td>
<td>positive</td>
</tr>
<tr>
<td>terrible movie</td>
<td>negative</td>
</tr>
<tr>
<td>not too shabby</td>
<td>positive</td>
</tr>
</tbody>
</table>
\( \hat{h}(x) \)

- The classification function that we want to learn has two different components:
  - the formal structure of the learning method (what’s the relationship between the input and output?) \( \rightarrow \) Naive Bayes, logistic regression, convolutional neural network, etc.
  - the representation of the data
Representation for SA

- Only positive/negative words in MPQA
- Only words in isolation (bag of words)
- Conjunctions of words (sequential, skip ngrams, other non-linear combinations)
- Higher-order linguistic structure (e.g., syntax)
“I hated this movie. Hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated hated this movie. Hated it. Hated every simpering stupid vacant audience-insulting moment of it. Hated the sensibility that thought anyone would like it.”

Roger Ebert, North

“… is a film which still causes real, not figurative, chills to run along my spine, and it is certainly the bravest and most ambitious fruit of Coppola's genius”

Roger Ebert, Apocalypse Now
Bag of words

Representation of text only as the counts of words that it contains
Naive Bayes

• Given access to \(<x,y>\) pairs in training data, we can train a model to estimate the class probabilities for a new review.

• With a bag of words representation (in which each word is independent of the other), we can use Naive Bayes

• Probabilistic model; not as accurate as other models (see next two classes) but fast to train and the foundation for many other probabilistic techniques.
Random variable

- A variable that can take values within a fixed set (discrete) or within some range (continuous).

\[ X \in \{1, 2, 3, 4, 5, 6\} \]

\[ X \in \{\text{the, a, dog, cat, runs, to, store}\} \]
\[ P(X = x) \]

Probability that the random variable \( X \) takes the value \( x \) (e.g., 1)

\[ X \in \{1, 2, 3, 4, 5, 6\} \]

Two conditions:

1. Between 0 and 1:
   \[ 0 \leq P(X = x) \leq 1 \]

2. Sum of all probabilities = 1
   \[ \sum_x P(X = x) = 1 \]
Fair dice

\[ X \in \{1, 2, 3, 4, 5, 6\} \]
Weighted dice

\[ X \in \{1, 2, 3, 4, 5, 6\} \]
Inference

\[ X \in \{1, 2, 3, 4, 5, 6\} \]

We want to infer the probability distribution that generated the data we see.
Probability

fair

not fair

2
Probability

fair

1 2 3 4 5 6

not fair

1 2 3 4 5 6
Probability

fair

not fair

1 2 3 4

0.0 0.1 0.2 0.3 0.4 0.5

1 2 3 4 5 6
Probability

The diagram shows the probability distribution for a fair and a not-fair dice. The numbers 1 to 6 are shown on the x-axis, with the bars indicating the probability of each number being rolled.

For a fair dice, the probability of rolling each number is approximately equal, with each number having a probability of around 0.167.

For a not-fair dice, the probability distribution is skewed, with a higher probability for the number 6, indicating it is more likely to be rolled.
Probability

fair

not fair

3

0.0 0.1 0.2 0.3 0.4 0.5
1 2 3 4 5 6

1 2 3 4 5 6
Probability

fair

not fair
Independence

- Two random variables are independent if:

\[ P(A, B) = P(A) \times P(B) \]

- In general:

\[ P(x_1, \ldots, x_n) = \prod_{i=1}^{N} P(x_i) \]

- Information about one random variable (B) gives no information about the value of another (A)

\[ P(A) = P(A \mid B) \quad P(B) = P(B \mid A) \]
$P(2, 6, 6 \mid \text{fair}) = .17 \times .17 \times .17 = 0.004913$

$P(2, 6, 6 \mid \text{not fair}) = .1 \times .5 \times .5 = 0.025$
Data Likelihood

• The likelihood gives us a way of discriminating between possible alternative parameters, but also a strategy for picking a single best* parameter among all possibilities
Word choice as weighted dice
Unigram probability

positive reviews

negative reviews
\[ P(X = \text{the}) = \frac{\#\text{the}}{\#\text{total words}} \]
Maximum Likelihood Estimate

• This is a maximum likelihood estimate for $P(X)$; the parameter values for which the data we observe ($X$) is most likely.
Maximum Likelihood Estimate
\[
P(X | \theta_1) = 0.0000311040
\]

\[
P(X | \theta_2) = 0.0000000992 \quad (313x \text{ less likely})
\]

\[
P(X | \theta_3) = 0.0000031250 \quad (10x \text{ less likely})
\]
Conditional Probability

\[ P(X = x \mid Y = y) \]

- Probability that one random variable takes a particular value given the fact that a different variable takes another

\[ P(X_i = hate \mid Y = \oplus) \]
Sentiment analysis

“really really the worst movie ever”
Independence Assumption

really really the worst movie ever

\[ P(\text{really}, \text{really}, \text{the}, \text{worst}, \text{movie}, \text{ever}) = P(\text{really}) \times P(\text{really}) \times P(\text{the}) \times \ldots \times P(\text{ever}) \]
Independence Assumption

really really the worst movie ever

We will assume the features are independent:

\[ P(x_1, x_2, x_3, x_4, x_6, x_7 \mid c) = P(x_1 \mid c)P(x_2 \mid c) \ldots P(x_7 \mid c) \]

\[ P(x_i \ldots x_n \mid c) = \prod_{i=1}^{N} P(x_i \mid c) \]
A simple classifier

really really the worst movie ever

<table>
<thead>
<tr>
<th></th>
<th>Y=Positive</th>
<th></th>
<th>Y=Negative</th>
</tr>
</thead>
<tbody>
<tr>
<td>P(X=really</td>
<td>Y=⊕)</td>
<td>0.0010</td>
<td>P(X=really</td>
</tr>
<tr>
<td>P(X=really</td>
<td>Y=⊕)</td>
<td>0.0010</td>
<td>P(X=really</td>
</tr>
<tr>
<td>P(X=the</td>
<td>Y=⊕)</td>
<td>0.0551</td>
<td>P(X=the</td>
</tr>
<tr>
<td>P(X=worst</td>
<td>Y=⊕)</td>
<td>0.0001</td>
<td>P(X=worst</td>
</tr>
<tr>
<td>P(X=movie</td>
<td>Y=⊕)</td>
<td>0.0032</td>
<td>P(X=movie</td>
</tr>
<tr>
<td>P(X=ever</td>
<td>Y=⊕)</td>
<td>0.0005</td>
<td>P(X=ever</td>
</tr>
</tbody>
</table>
A simple classifier

really really the worst movie ever

\[
P(X = \text{“really really the worst movie ever”} \mid Y = \oplus) = 6.00e^{-18}
\]

\[
P(X = \text{“really really the worst movie ever”} \mid Y = \ominus) = 6.20e^{-17}
\]
Aside: use logs

- Multiplying lots of small probabilities (all are under 1) can lead to numerical underflow (converging to 0)

\[ \log \prod_{i} x_i = \sum_{i} \log x_i \]
A simple classifier

- The classifier we just specified is a maximum likelihood classifier, where we compare the likelihood of the data under each class and choose the class with the highest likelihood.

Likelihood: probability of data (here, under class y) \( P(X = x_i \ldots x_n \mid Y = y) \)

Prior probability of class y \( P(Y = y) \)
Bayes’ Rule

\[ P(Y = y | X = x) = \frac{P(Y = y) P(X = x | Y = y)}{\sum_y P(Y = y) P(X = x | Y = y)} \]

Prior belief that \( Y = y \) (before you see any data)

Posterior belief that \( Y = y \) given that \( X = x \)

Likelihood of the data given that \( Y = y \)
Bayes’ Rule

\[ P(Y = y|X = x) = \frac{P(Y = y)P(X = x|Y = y)}{\sum_y P(Y = y)P(X = x|Y = y)} \]

Prior belief that Y = positive (before you see any data)
Likelihood of “really really the worst movie ever” given that Y= positive
Posterior belief that Y=positive given that X=“really really the worst movie ever”
This sum ranges over y=positive + y=negative (so that it sums to 1)
**Likelihood**: probability of data (here, under class y)

\[ P(X = x_i \ldots x_n \mid Y = y) \]

**Prior** probability of class y

\[ P(Y = y) \]

**Posterior** belief in the probability of class y after seeing data

\[ P(Y = y \mid X = x_i \ldots x_n) \]
Naive Bayes Classifier

\[
P(Y = \oplus) P(X = \text{“really . . .”} \mid Y = \oplus) \\
\frac{P(Y = \oplus) P(X = \text{“really . . .”} \mid Y = \oplus) + P(Y = \ominus) P(X = \text{“really . . .”} \mid Y = \ominus)}
\]

Let’s say \( P(Y = \oplus) = P(Y = \ominus) = 0.5 \)
(i.e., both are equally likely a priori)

\[
\frac{0.5 \times (6.00 \times 10^{-18})}{0.5 \times (6.00 \times 10^{-18}) + 0.5 \times (6.2 \times 10^{-17})}
\]

\[
P(Y = \oplus \mid X = \text{“really . . .”}) = 0.088
\]

\[
P(Y = \ominus \mid X = \text{“really . . .”}) = 0.912
\]
Naive Bayes Classifier

• To turn probabilities into a classification decisions, we just select the label with the highest posterior probability

\[ \hat{y} = \arg \max_{y \in Y} \, P(Y \mid X) \]

\[
P(Y = \oplus \mid X = "really \ldots") = 0.088
\]
\[
P(Y = \ominus \mid X = "really \ldots") = 0.912
\]
Taxicab Problem

“A cab was involved in a hit and run accident at night. Two cab companies, the Green and the Blue, operate in the city. You are given the following data:

- 85% of the cabs in the city are Green and 15% are Blue.

- A witness identified the cab as Blue. The court tested the reliability of the witness under the same circumstances that existed on the night of the accident and concluded that the witness correctly identified each one of the two colors 80% of the time and failed 20% of the time.

What is the probability that the cab involved in the accident was Blue rather than Green knowing that this witness identified it as Blue?”

(Tversky & Kahneman 1981)
Prior Belief

- Now let’s assume that there are 1000 times more positive reviews than negative reviews.

- \( P(Y = \text{negative}) = 0.000999 \)
- \( P(Y = \text{positive}) = 0.999001 \)

\[
\frac{0.999001 \times (6.00 \times 10^{-18})}{0.999001 \times (6.00 \times 10^{-18}) + 0.000999 \times (6.2 \times 10^P(-17))}
\]

\[
P(Y = \oplus | X = \text{“really ...”}) = 0.990
\]
\[
P(Y = \ominus | X = \text{“really ...”}) = 0.010
\]
Priors

- Priors can be informed (reflecting expert knowledge) but in practice, but priors in Naive Bayes are often simply estimated from training data

\[ P(Y = \oplus) = \frac{\#\oplus}{\#\text{total texts}} \]
Smoothing

• Maximum likelihood estimates can fail miserably when features are never observed with a particular class.

What's the probability of:

- 2
- 4
- 6
Smoothing

- One solution: add a little probability mass to every element.

maximum likelihood estimate

\[ P(x_i \mid y) = \frac{n_{i,y}}{n_y} \]

smoothed estimates

\[ P(x_i \mid y) = \frac{n_{i,y} + \alpha}{n_y + V\alpha} \]

- same \( \alpha \) for all \( x_i \)

\[ P(x_i \mid y) = \frac{n_{i,y} + \alpha_i}{n_y + \sum_{j=1}^{V} \alpha_j} \]

- possibly different \( \alpha \) for each \( x_i \)

- \( n_{i,y} \) = count of word \( i \) in class \( y \)
- \( n_y \) = number of words in \( y \)
- \( V \) = size of vocabulary
Smoothing

MLE

smoothing with $\alpha = 1$
Naive Bayes training

Training a Naive Bayes classifier consists of estimating these two quantities from training data for all classes $y$:

$$P(Y = y | X = x) = \frac{P(Y = y) P(X = x | Y = y)}{\sum_y P(Y = y) P(X = x | Y = y)}$$

At test time, use those estimated probabilities to calculate the posterior probability of each class $y$ and select the class with the highest probability.
• Naive Bayes’ independence assumption can be killer

• One instance of *hate* makes seeing others much more likely (each mention does contribute the same amount of information)

• We can mitigate this by not reasoning over counts of tokens but by their presence absence

<table>
<thead>
<tr>
<th></th>
<th>Apocalypse now</th>
<th>North</th>
</tr>
</thead>
<tbody>
<tr>
<td>the</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>of</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>hate</td>
<td>0</td>
<td>9</td>
</tr>
<tr>
<td>genius</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>bravest</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>stupid</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>like</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Multinomial Naive Bayes

Discrete distribution for modeling count data (e.g., word counts; single parameter $\theta$

$$\theta = \begin{bmatrix}
\text{the} & \text{a} & \text{dog} & \text{cat} & \text{runs} & \text{to} & \text{store} \\
3 & 1 & 0 & 1 & 0 & 2 & 0 \\
531 & 209 & 13 & 8 & 2 & 331 & 1
\end{bmatrix}$$
### Multinomial Naive Bayes

Maximum likelihood parameter estimate

\[ \hat{\theta}_i = \frac{n_i}{N} \]

<table>
<thead>
<tr>
<th>count n</th>
<th>the</th>
<th>a</th>
<th>dog</th>
<th>cat</th>
<th>runs</th>
<th>to</th>
<th>store</th>
</tr>
</thead>
<tbody>
<tr>
<td>531</td>
<td>209</td>
<td>13</td>
<td>8</td>
<td>2</td>
<td>331</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

\[ \theta = 0.48 \quad 0.19 \quad 0.01 \quad 0.01 \quad 0.00 \quad 0.30 \quad 0.00 \]
Bernoulli Naive Bayes

- Binary event (true or false; \{0, 1\})
- One parameter: \( p \) (probability of an event occurring)

Examples:
- Probability of a particular feature being true (e.g., review contains “hate”)

\[ \hat{p}_{mle} = \frac{1}{N} \sum_{i=1}^{N} x_i \]
## Bernoulli Naive Bayes

<table>
<thead>
<tr>
<th></th>
<th>X1</th>
<th>X2</th>
<th>X3</th>
<th>X4</th>
<th>X5</th>
<th>X6</th>
<th>X7</th>
<th>X8</th>
<th>pMLE</th>
</tr>
</thead>
<tbody>
<tr>
<td>f1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0.375</td>
</tr>
<tr>
<td>f2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0.125</td>
</tr>
<tr>
<td>f3</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0.750</td>
</tr>
<tr>
<td>f4</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0.500</td>
</tr>
<tr>
<td>f5</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.000</td>
</tr>
</tbody>
</table>
## Bernoulli Naive Bayes

<table>
<thead>
<tr>
<th></th>
<th>Positive</th>
<th></th>
<th>Negative</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>X1</td>
<td>X2</td>
<td>X3</td>
<td>X4</td>
<td>X5</td>
</tr>
<tr>
<td><strong>f_1</strong></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td><strong>f_2</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td><strong>f_3</strong></td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>f_4</strong></td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td><strong>f_5</strong></td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>
Tricks for SA

• Negation in bag of words: add negation marker to all words between negation and end of clause (e.g., comma, period) to create new vocab term [Das and Chen 2001]

• I do not [like this movie]

• I do not like_NEG this_NEG movie_NEG
Sentiment Dictionaries

- MPQA subjectivity lexicon (Wilson et al. 2005)

- LIWC (Linguistic Inquiry and Word Count, Pennebaker 2015)

<table>
<thead>
<tr>
<th>pos</th>
<th>neg</th>
</tr>
</thead>
<tbody>
<tr>
<td>unlimited</td>
<td>lag</td>
</tr>
<tr>
<td>prudent</td>
<td>contortions</td>
</tr>
<tr>
<td>superb</td>
<td>fright</td>
</tr>
<tr>
<td>closeness</td>
<td>lonely</td>
</tr>
<tr>
<td>impeccably</td>
<td>tenuously</td>
</tr>
<tr>
<td>fast-paced</td>
<td>plebeian</td>
</tr>
<tr>
<td>treat</td>
<td>mortification</td>
</tr>
<tr>
<td>destined</td>
<td>outrage</td>
</tr>
<tr>
<td>blessing</td>
<td>allegations</td>
</tr>
<tr>
<td>steadfastly</td>
<td>disoriented</td>
</tr>
</tbody>
</table>
Homework 1: due 9/4

Annotate the sentiment by the writer toward the people and organizations mentioned.