CS 6501 Natural Language Processing

Text Classification and Naive Bayes

Yangfeng Ji

September 10, 2019

Department of Computer Science
University of Virginia

Overview

1. Problem Definition
2. Bag-of-Words Representation
3. Naive Bayes Classifiers
4. Classification Evaluation

Problem Definition

Case I: Sentiment Analysis

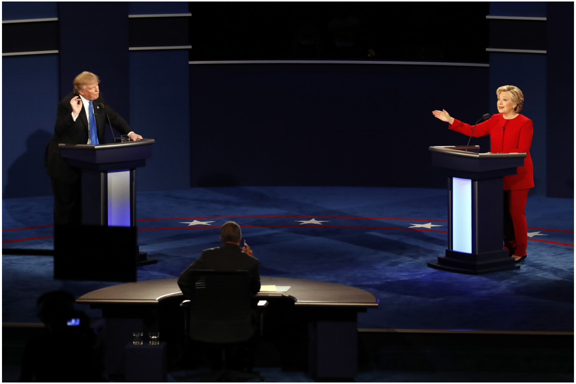
[Pang et al., 2002]

Case II: Topic Classification

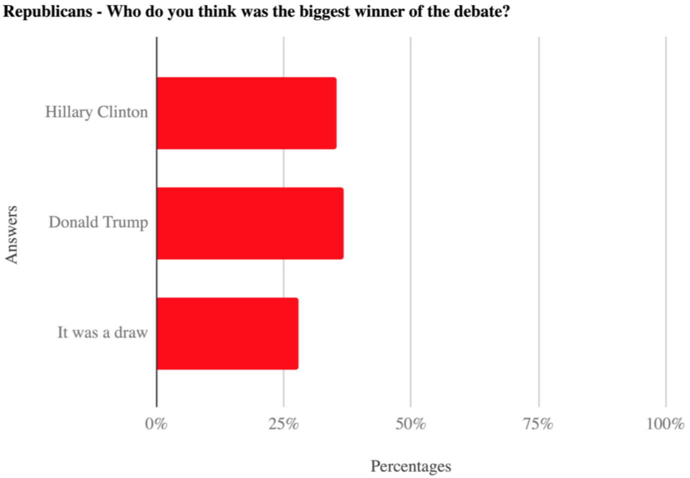
Example topics

- Business
- Arts
- Technology
- Sports
- ...

Case III: Presidential Election Debates



Case III: Presidential Election Debates (II)



Classification

- Input: a text $\boldsymbol{x}$
- Output: $y \in \mathcal{Y}$, where $\mathcal{Y}$ is the predefined category set (sample space)
  - Example: $\mathcal{Y} = \{\textsc{Positive}, \textsc{Negative}\}$

$$\text{Text } \boldsymbol{x} \longrightarrow \boxed{\text{Classifier}} \longrightarrow \text{Category } y$$

Probabilistic Formulation

With the conditional probability $P(Y \mid \boldsymbol{X})$, the prediction on Y for a given text $\boldsymbol{X} = \boldsymbol{x}$ is

$$\hat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}} P(Y = y \mid \boldsymbol{X} = \boldsymbol{x}) \tag{1}$$

Or, for simplicity

$$\hat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}} p(y \mid \boldsymbol{x}) \tag{2}$$

Key Questions

$$\hat{y} = \operatorname*{argmax}_{y \in \mathcal{Y}} P(Y = y \mid \boldsymbol{X} = \boldsymbol{x}) \tag{3}$$

1. How to represent a text as $\boldsymbol{x}$?
   - Bag-of-words representation
2. How to estimate $P(y \mid \boldsymbol{x})$?
   - Naive Bayes classifier

Bag-of-Words Representation

Bag-of-Words Representation

Tokenization: convert a text into a collection of tokens:

Super quick and really friendly staff. I'd like starting off my mornings at this store!! $\Rightarrow$ super quick and really friendly staff . I 'd $\cdots$ store ! !

NLTK function

`nltk.tokenize.wordpunct_tokenize`

Vocab

Collect tokens to build a vocab:

super quick and really friendly staff . I 'd $\cdots$ store ! !

$$\Rightarrow \left\{ \begin{array}{c} \text{SUPER} \\ \cdots \\ \text{QUICK} \\ \text{FOOD} \\ \text{FRIENDLY} \\ \text{EAT} \\ \cdots \\ \text{STAFF} \end{array} \right\}$$

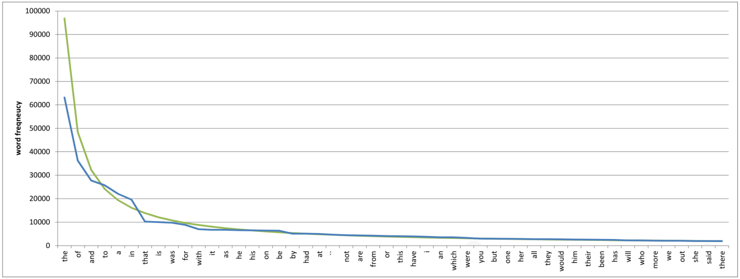
Should we collect all the tokens?

Preprocessing for Building Vocab

1. convert all characters to lowercase

$$\text{UVa}, \text{UVA} \rightarrow \textsc{uva}$$

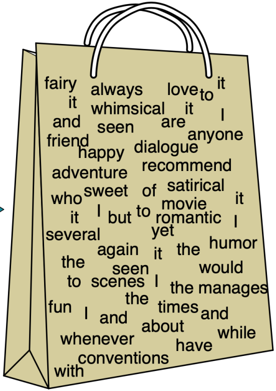
2. map low frequency words to a special token UNK



Zipf's law: $f(w_t) \propto 1/r_t$

Bag-of-Words Representation

I love this movie! It's sweet, but with satirical humor. The dialogue is great and the adventure scenes are fun... It manages to be whimsical and romantic while laughing at the conventions of the fairy tale genre. I would recommend it to just about anyone. I've seen it several times, and I'm always happy to see it again whenever I have a friend who hasn't seen it yet!



it	6
I	5
the	4
to	3
and	3
seen	2
yet	1
would	1
whimsical	1
times	1
sweet	1
satirical	1
adventure	1
genre	1
fairy	1
humor	1
have	1
great	1
…	…

Information Embedded in BoW Representation

- Lose:
  - word order
  - sentence boundary
  - paragraph boundary
  - ...
- Keep: words in texts

Naive Bayes Classifiers

Decision Rule

Given a document $\boldsymbol{x}$, the classification can be conducted as

$$\hat{y} = \operatorname{argmax}_{y \in \mathcal{Y}} p(y \mid \boldsymbol{x}; \boldsymbol{\theta}). \tag{4}$$

where $\boldsymbol{\theta}$ is the parameter of the distribution.

Bayes' Theorem

In Bayes' rule, the conditional probability $p(y \mid \boldsymbol{x})$ can be compuated via the joint probability $p(\boldsymbol{x}, y)$ as follow

$$\begin{aligned} p(y \mid \boldsymbol{x}) &= \frac{p(\boldsymbol{x}, y)}{p(\boldsymbol{x})} \\ &= \frac{p(\boldsymbol{x} \mid y)p(y)}{p(\boldsymbol{x})} \end{aligned} \tag{5}$$

- $p(y, \boldsymbol{\theta})$: prior probability of $Y = y$
- $p(\boldsymbol{x} \mid y; \boldsymbol{\theta})$:
  - conditional probability of $\boldsymbol{X} = \boldsymbol{x}$ given y
  - likelihood function of $Y = y$ given $\boldsymbol{x}$

Simplification

$$
\begin{aligned}
\hat{y} &= \operatorname*{argmax}_{y \in \mathcal{Y}} p(y \mid \boldsymbol{x}; \boldsymbol{\theta}) \\
&= \operatorname*{argmax}_{y \in \mathcal{Y}} \underbrace{p(\boldsymbol{x} \mid y)}_{\text{likelihood}} \underbrace{p(y)}_{\text{prior}} \qquad (6)
\end{aligned}
$$

Benefit of equation 6: no need to compute $p(\boldsymbol{x})$ explicitly.

Naive Bayes Assumption

To model $p(\boldsymbol{x} \mid y)$: we assume words within a text are independent with each other

- The distribution of $p(\boldsymbol{x} \mid y)$ is similar to the one of modeling tossing a dice with V faces for n times
- This is a naive assumption

Conditional probability $p(\boldsymbol{x} \mid y; \boldsymbol{\theta})$ can be written as a multinomial distribution

$$p(\boldsymbol{x} \mid y; \boldsymbol{\theta}) \propto \prod_{i=1}^{V} \theta_{i,y}^{x_i} \tag{7}$$

Parameters

- Proir probability (categorical distribution):

$$p(y) = \theta_y \tag{8}$$

- Likelihood (multinominal distribution):

$$p(\boldsymbol{x} \mid y; \boldsymbol{\theta}) \propto \prod_{i=1}^{V} {\theta_{i,y}}^{x_i} \tag{9}$$

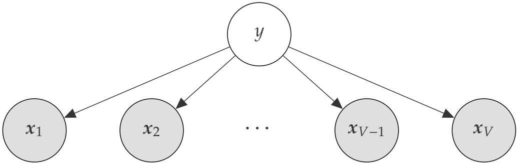
- Number of the parameters $K + KV = (K+1)V$

Probabilistic Graphical Model

Joint probability:

$$p(\boldsymbol{x}, y; \boldsymbol{\theta}) \propto \theta_y \cdot \prod_{i=1}^{V} \theta_{i,y}{}^{x_i} \tag{10}$$

The conditional independence can be represented in the following graphical model:



How to estimate $\boldsymbol{\theta} = \{\theta_y, \theta_{i,y}\}, \forall i, y$

Example

Training examples

Label	Text	Words
NEG	`just plain boring`	3
NEG	`entirely predictable and lacks energy`	5
NEG	`no surprises and very few langhs`	6
POS	`very powerful`	2
POS	`the most fun film of the summer`	7

[Jurafsky and Martin, 2019]

Parameter Estimation (I)

For prior probability

$$\theta_y = \frac{\text{Number of texts with label } y}{\text{Total number of texts}} = \frac{N_y}{N_{\text{total}}} \tag{11}$$

Example

There are three documents: two positive and three negative

$$p(\textsc{Positive}) = \frac{2}{5} \quad p(\textsc{Negative}) = \frac{3}{5} \tag{12}$$

Parameter Estimation (II)

For $\theta_{i,y} = P(x_i \mid y)$

$$\theta_{i,y} = \frac{\text{count}(x_i, y)}{\sum_{i=1}^{V} \text{count}(x_i, y)} \tag{13}$$

$$= \frac{\text{count}(x_i, y)}{\text{count}(y)} \tag{14}$$

Therefore

$$\sum_{i=1}^{V} \theta_{i,y} = 1 \tag{15}$$

Example: Build a vocabulary

Label	Text	Words
NEG	`just plain boring`	3
NEG	`entirely predictable and lacks energy`	5
NEG	`no surprises and very few langhs`	6
POS	`very powerful`	2
POS	`the most fun film of the summer`	7

Vocab:

- $\mathcal{V}$ = {just, plain, boring, entirely, . . . , summer}
- $V = 20$

Example: Estimate $p(x_i \mid y)$

Label	Text	Words
Neg	`just plain boring`	3
Neg	`entirely predictable and lacks energy`	5
Neg	`no surprises and very few langhs`	6
Pos	`very powerful`	2
Pos	`the most fun film of the summer`	7

For example

$$p(\text{just} \mid \text{NEG}) = \frac{1}{3+5+6} = \frac{1}{14} \quad p(\text{just} \mid \text{POS}) = \frac{0}{2+7} = \frac{0}{9}$$

Example: Prediction

For a given sentence "`predictable and boring`"

$$p(\text{predictable} \mid \textsc{neg}) = \frac{1}{14} \qquad p(\text{predictable} \mid \textsc{pos}) = \frac{0}{9}$$

$$p(\text{and} \mid \textsc{neg}) = \frac{2}{14} \qquad p(\text{and} \mid \textsc{pos}) = \frac{0}{9}$$

$$p(\text{boring} \mid \textsc{neg}) = \frac{1}{14} \qquad p(\text{boring} \mid \textsc{pos}) = \frac{0}{9}$$

Therefore,

$$\begin{aligned} p(S, \text{neg}) &= \frac{3}{5} \cdot \frac{1}{14} \cdot \frac{2}{14} \cdot \frac{1}{14} \\ p(S, \text{pos}) &= \frac{2}{5} \cdot \frac{0}{9} \cdot \frac{0}{9} \cdot \frac{0}{9} \end{aligned}$$

Example: Prediction (II)

For a given sentence "`predictable with no fun`"

$$p(\text{predictable} \mid \textsc{neg}) = \frac{1}{14} \qquad p(\text{predictable} \mid \textsc{pos}) = \frac{0}{9}$$

$$p(\text{no} \mid \textsc{neg}) = \frac{1}{14} \qquad p(\text{and} \mid \textsc{pos}) = \frac{0}{9}$$

$$p(\text{fun} \mid \textsc{neg}) = \frac{0}{14} \qquad p(\text{fun} \mid \textsc{pos}) = \frac{1}{9}$$

Therefore,

$$\begin{aligned} p(S, \text{neg}) &= \frac{3}{5} \cdot \frac{1}{14} \cdot \frac{1}{14} \cdot \frac{0}{14} \\ P(S, \text{pos}) &= \frac{2}{5} \cdot \frac{0}{9} \cdot \frac{0}{9} \cdot \frac{1}{9} \end{aligned}$$

Smoothing

To eliminate zero probability, adding a small number α to the counts:

$$p(x_i \mid y) = \frac{\text{count}(x_i, y) + \alpha}{\sum_{i=1}^{V}\{\text{count}(x_i, y) + \alpha\}} \tag{16}$$

$$= \frac{\text{count}(x_i, y) + \alpha}{\sum_{i=1}^{V} \text{count}(x_i, y) + \alpha V} \tag{17}$$

with $\alpha > 0$ as a **hyper-parameter**.

For example, with $\alpha = 1$ and $V = 20$

$$p(\text{fun} \mid \textsc{neg}) = \frac{0+1}{14+20} \quad p(\text{fun} \mid \textsc{pos}) = \frac{1+1}{9+20}$$

An Alternative View

- Likelihood: $p(\boldsymbol{x} \mid y; \boldsymbol{\theta}) \propto \prod_{i=1}^{V} \theta_{i,y}{}^{x_i}$
- Prior: $p(y; \boldsymbol{\theta}) = \theta_y$

$$p(\boldsymbol{x}, y; \boldsymbol{\theta}) \propto \theta_y \cdot \prod_{i=1}^{V} \theta_{i,y}{}^{x_i} \tag{18}$$

Or

$$\log p(\boldsymbol{x}, y; \boldsymbol{\theta}) \propto \log \theta_y + \sum_{i=1}^{V} (x_i \cdot \log \theta_{i,y}) \tag{19}$$

An Alternative View (II)

$$\log p(\boldsymbol{x}, y) \propto \log \theta_y + \sum_{i=1}^{V} (x_i \cdot \log \theta_{i,y}) \tag{20}$$

$$= b_y + \boldsymbol{w}_y^\top \boldsymbol{x} \tag{21}$$

where

$$\begin{aligned} b_y &= \log \theta_y \\ \boldsymbol{w}_y^\top &= [\log \theta_{1,y}, \ldots, \log \theta_{V,y}] \\ \boldsymbol{x}^\top &= [x_1, \ldots, x_V] \end{aligned}$$

It's a linear classifier (with respect to $\boldsymbol{x}$)

Classification Evaluation

A Development Set

- Training set $\mathcal{T} = \{(\boldsymbol{x}^{(i)}, y^{(i)})\}_{i=1}^{N}$
- Development set $\mathcal{D} = \{(\boldsymbol{x}^{(j)}, y^{(j)})\}_{j=1}^{M}$
- Test set $\mathcal{U} = \{(\boldsymbol{x}^{(l)}, y^{(l)})\}_{l=1}^{L}$

Cross-validation

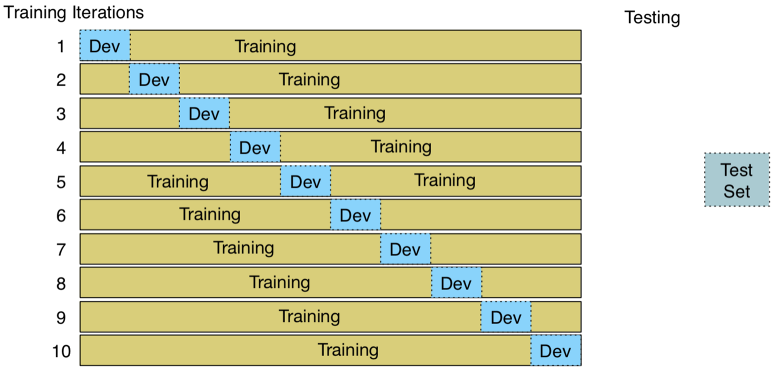


Figure: 10-fold cross validation.

Evaluation Measurements

- Accuracy
- Precision, recall, and F-measure

[Eisenstein, 2018, Sec 4.4]

Accuracy

Given N examples, with $y^{(i)}$ is the ground-truth label of the i-th example, and $\hat{y}^{(i)}$ is the predicted label

$$\textsc{acc}(\boldsymbol{y}, \hat{\boldsymbol{y}}) = \frac{1}{N}\delta(y^{(i)}, \hat{y}^{(i)}) \tag{22}$$

δ function is defined as

$$\delta(y, \hat{y}) = \begin{cases} 1 & y = \hat{y} \\ 0 & y \neq \hat{y} \end{cases} \tag{23}$$

Confusion Matrix

		Ground truth	
		POSITIVE	NEGATIVE
Prediction	POSITIVE	True Positive (TP)	False Positive (FP)
	NEGATIVE	False Negative (FN)	True Negative (TN)

Accuracy:

$$\mathrm{acc}(y, \hat{y}) = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \tag{24}$$

Recall, Precision and F Measure

		Ground truth	
		POSITIVE	NEGATIVE
Prediction	POSITIVE	True Positive (TP)	False Positive (FP)
	NEGATIVE	False Negative (FN)	True Negative (TN)

Recall:

$$r(y, \hat{y}) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{25}$$

Precision:

$$p(y, \hat{y}) = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{26}$$

F measure:

$$F(y, \hat{y}) = \frac{2 \cdot p \cdot r}{p + r} \tag{27}$$

Example: Balanced case

		Ground truth	
		POSITIVE	NEGATIVE
Prediction	POSITIVE	480	30
	NEGATIVE	20	470

- Accuracy: $\text{acc} = \frac{480+470}{480+20+30+470} = 0.95$
- Precision: $p = \frac{480}{480+30} \approx 0.94$
- Recall: $r = \frac{480}{480+20} \approx 0.96$
- F-measure: $F = \frac{2\times0.94\times0.96}{0.94+0.96} \approx 0.95$

Example: Unbalanced case

		Ground truth	
		POSITIVE	NEGATIVE
Prediction	POSITIVE	80	30
	NEGATIVE	20	870

- Accuracy: $\text{acc} = \frac{80+870}{80+20+30+870} = 0.95$
- Precision: $p = \frac{80}{80+30} \approx 0.73$
- Recall: $r = \frac{80}{80+20} = 0.80$
- F-measure: $F = \frac{2\times0.94\times0.96}{0.94+0.96} \approx 0.76$

Reference

Eisenstein, J. (2018).
Natural Language Processing.
MIT Press.

Jurafsky, D. and Martin, J. (2019).
Speech and language processing.

Pang, B., Lee, L., and Vaithyanathan, S. (2002).
Thumbs up?: sentiment classification using machine learning techniques.
In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10*, pages 79–86. Association for Computational Linguistics.