CSE6390E Introduction to Computational Linguistics

Fully Independent Model and Naïve Bayes Model

Presented by Nikolay Yakovets

Agenda

1 Preliminaries/Last Day

2 Fully Independent Model

3 Naïve Bayes Model

Slides adapted from lectures by
Vlado Keselj
Dalhousie University

Probabilistic Model

- **Set of n random variables that capture the outcome in a model:**

$$V = (V_1, V_2, \ldots, V_n)$$

- **Each variable can be assigned a value from a finite set of different values:**

$$\{x_1, x_2, \ldots, x_m\}$$

- **Random configuration**
  - A tuple of values where each value is assigned to a variable

$$\vec{x} = (x_1, x_2, \ldots, x_n)$$

$$V_1 = x_1, V_2 = x_2, \ldots, V_n = x_n$$

Probabilistic Model

- **In modeling our problem we assume that a sequence of configurations is drawn from some random source:**

$$
\begin{aligned}
x^{(1)} &= (x_{11}, x_{12}, \ldots, x_{1n}) \\
x^{(2)} &= (x_{21}, x_{22}, \ldots, x_{2n}) \\
&\vdots \\
x^{(t)} &= (x_{t1}, x_{t2}, \ldots, x_{tn})
\end{aligned}
$$

- **Probabilistic Modeling in NLP is a general framework for modeling NLP problems using random variables, random configurations, and finding effective ways of reasoning about probabilities of these configurations.**

Computational Tasks

- **Evaluation**
  - Compute probability of a complete configuration
- **Simulation (aka Generation or Sampling)**
  - Generate random configurations
- **Inference**
  - Marginalization
    - Computing probability of a partial configuration
  - Conditioning
    - Computing conditional probability of a completion given an observation
  - Completion
    - Finding the most probable completion, given an observation
- **Learning**
  - Learning parameters of a model from data

Example: Spam Detection

- **GOAL: Automatically detect whether an arbitrary email message is spam or not**
- **Have 3 random variables in the model:**
  - *Caps* is '**Y**' if the message subject line does not contain lowercase letter, '**N**' otherwise
  - *Free* is '**Y**' if the word 'free' appears in the message subject line, '**N**' otherwise
  - *Spam* is '**Y**' if message is spam, and '**N**' otherwise
- **Mailbox is a random source**
  - Randomly select 100 messages
  - Count how many times each configuration (each email) appears

Free	*Caps*	*Spam*	Number of messages
Y	Y	Y	20
Y	Y	N	1
Y	N	Y	5

- **Last day**
  - Joint Distribution Model
    - Specify complete joint probability distribution, i.e. the probability of each complete configuration
  - Drawbacks
    - Memory cost to store table
    - Running-time to do summations
    - The sparse data problem in learning
- **Today**
  - Fully Independent Model

Fully Independent Model

- **Assume all variables are independent:**

$$P(V_1 = x_1, \ldots, V_n = x_n) = P(V_1 = x_1) \cdots P(V_n = x_n)$$

- **This is an efficient model**
  - Small number of parameters: $O(nm)$
  - Represent each component of distribution separately
    - Fetch $P(V_j = x)$ from lookup table with m parameters

Evaluation

- **BUT usually a too strong assumption!**
  - Very restricted form of joint distribution
  - Silly model as far as real applications go, not very useful
- **Translated into SPAM example:**

$$P(Free, Caps, Spam) = P(Free) \cdot P(Caps) \cdot P(Spam)$$

Example: Spam Detection

- **Assume**
  - ***Caps***, ***Free*** and ***Spam*** are independent
- **Say have the following data:**

Free	*Caps*	*Spam*	Number of messages	P
Y	Y	Y	20	0.20
Y	Y	N	1	0.01
Y	N	Y	5	0.05
Y	N	N	0	0.00
N	Y	Y	20	0.20
N	Y	N	3	0.03
N	N	Y	2	0.02
N	N	N	49	0.49
		Total:	100	1.00

Any message is **Spam** with **P=0.47** no matter what **Free** or **Caps** is!

- **Estimate probability tables for independent variables:**

Free	$\mathrm{P}(\textit{Free})$
Y	$\frac{20+1+5+0}{100} = 0.26$
N	$\frac{20+3+2+49}{100} = 0.74$

Caps	$\mathrm{P}(\textit{Caps})$
Y	$\frac{20+1+20+3}{100} = 0.44$
N	$\frac{5+0+2+49}{100} = 0.56$

Spam	$\mathrm{P}(\textit{Spam})$
Y	$\frac{20+5+20+2}{100} = 0.47$
N	$\frac{1+0+3+49}{100} = 0.53$

MLE Learning

Example: Spam Detection

- Say want to know the probability of configuration:

$$(Caps = Y, Free = N, Spam = N)$$

- Then the probability in fully independent model evaluates to the following:

$$\begin{aligned} &\mathrm{P}(\mathit{Free} = Y, \mathit{Caps} = N, \mathit{Spam} = N) = \\ &= \quad \mathrm{P}(\mathit{Free} = Y) \cdot \mathrm{P}(\mathit{Caps} = N) \cdot \mathrm{P}(\mathit{Spam} = N) = 0.26 \cdot 0.56 \cdot 0.53 \\ &= \quad 0.077168 \approx 0.08 \end{aligned}$$

Fetch from lookup table

Evaluation

Computational Tasks

- **Evaluation**
- **Simulation (Sampling)**
  - For every $j = 1, \ldots, n$ independently sample x_j according to lookup table value of $P(V_j = x_j)$
  - Conjoin $(x_1, \ldots, x_n)$ to form a compete configuration
- **Inference – Marginalization**

$$
\begin{aligned}
\boxed{\mathrm{P}(V_1 = x_1, \ldots, V_k = x_k)} &= \sum_{y_{k+1}} \cdots \sum_{y_n} \mathrm{P}(V_1 = x_1, \ldots, V_k = x_k, V_{k+1} = y_{k+1}, \ldots, V_n = y_n) \\
&= \sum_{y_{k+1}} \cdots \sum_{y_n} \mathrm{P}(V_1 = x_1) \cdots \mathrm{P}(V_k = x_k) \mathrm{P}(V_{k+1} = y_{k+1}) \cdots \mathrm{P}(V_n = y_n) \\
&= \mathrm{P}(V_1 = x_1) \cdots \mathrm{P}(V_k = x_k) \left[\sum_{y_{k+1}} \mathrm{P}(V_{k+1} = y_{k+1}) \left[\sum_{y_{k+2}} \cdots \left[\sum_{y_n} \mathrm{P}(V_n = y_n) \right] \right] \right] \\
&= \mathrm{P}(V_1 = x_1) \cdots \mathrm{P}(V_k = x_k) \left[\sum_{y_{k+1}} \mathrm{P}(V_{k+1} = y_{k+1}) \right] \cdots \left[\sum_{y_n} \mathrm{P}(V_n = y_n) \right] \\
&= \boxed{\mathrm{P}(V_1 = x_1) \cdots \mathrm{P}(V_k = x_k)}
\end{aligned}
$$

Sum-product computation:
y_k is constant for summation over y_{k+1}!

- **Only have to lookup and multiply n numbers!**

Computational Tasks

❖ **Inference – Conditioning**

$$\boxed{\mathrm{P}(V_{k+1}=y_{k+1},\ldots,V_n=y_n \mid V_1=x_1,\ldots,V_k=x_k)}$$

$$= \frac{\mathrm{P}(V_1=x_1,\ldots,V_k=x_k,V_{k+1}=y_{k+1},\ldots,V_n=y_n)}{\mathrm{P}(V_1=x_1,\ldots,V_k=x_k)}$$

$$= \frac{\cancel{\mathrm{P}(V_1=x_1)\cdots\mathrm{P}(V_k=x_k)}\mathrm{P}(V_{k+1}=y_{k+1})\cdots\mathrm{P}(V_n=y_n)}{\cancel{\mathrm{P}(V_1=x_1)\cdots\mathrm{P}(V_k=x_k)}}$$

$$= \boxed{\mathrm{P}(V_{k+1}=y_{k+1})\cdots\mathrm{P}(V_n=y_n)}$$

❖ **Only have to lookup and multiply** $n-k$ **numbers**

Example: Spam Detection

- **Inference – Completion**

$$y^*_{k+1}, \ldots, y^*_n = \underset{y_{k+1},\ldots,y_n}{\arg\max}\ \mathrm{P}(V_{k+1}=y_{k+1}, \ldots, V_n=y_n | V_1=x_1, \ldots, V_k=x_k)$$

Conditioning

$$= \underset{y_{k+1},\ldots,y_n}{\arg\max}\ \mathrm{P}(V_{k+1}=y_{k+1}) \cdots \mathrm{P}(V_n=y_n)$$

$$= \left[\underset{y_{k+1}}{\arg\max}\,\mathrm{P}(V_{k+1}=y_{k+1})\right] \cdots \left[\underset{y_n}{\arg\max}\,\mathrm{P}(V_n=y_n)\right]$$

Sum-product computation:
y_k is constant for argmax over y_{k+1}!

- **Only have to search through m possible completions for each of $n-k$ variables separately**

Pros and Cons

- **Joint Distribution Model vs. Fully Independent Model**

Advantages

- **Efficient**

  $O(nm)$ Vs. $O(m^n)$

  Lot less to compute!

- **No sparse data problem**

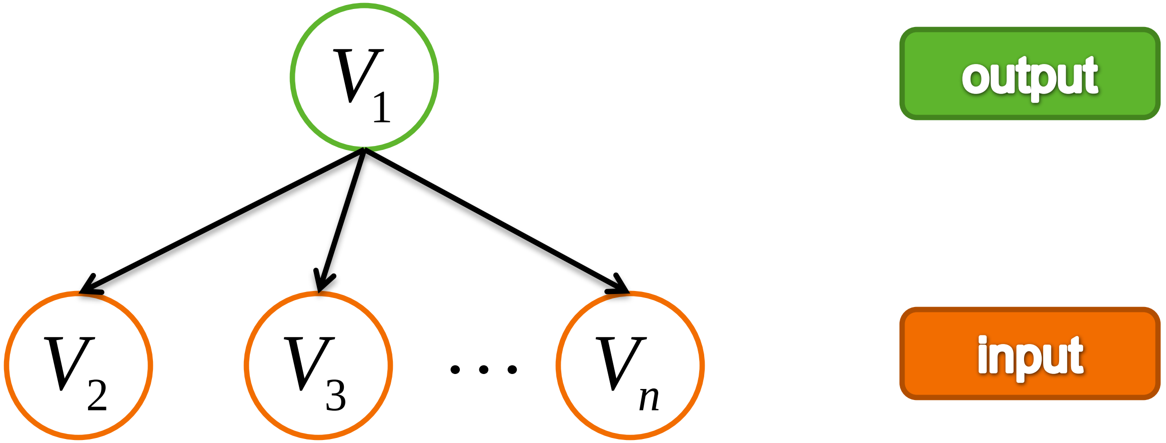
Disadvantages

- **Too strong assumption!**
- **Too little structure**
- **Usually does not model accurately**

- **Structured probability models are a compromise solution between these two models**
  - Address the issue of sparse data
  - Model important dependencies among random variables

Naïve Bayes Model

- **Structured Probability Model**
- **Assume that all variables are independent except one distinguished variable – class variable**
- **Graphical representation:**



Naïve Bayes Model

- Assume V_1 is the output variable and $V_2, \ldots, V_n$ are input variables
- Then classification problem is a conditional probability computation problem:

$$\mathrm{P}(V_1 = x_1 | V_2 = x_2, V_3 = x_3, \ldots, V_n = x_n)$$

- After applying Bayes theorem we obtain:

$$P(V_1 | V_2, V_3, \ldots, V_n) = \frac{P(V_2, V_3, \ldots, V_n | V_1)}{P(V_2, V_3, \ldots, V_n)} = \frac{\boxed{P(V_2 | V_1) \cdot P(V_3 | V_1) \cdot \ldots \cdot P(V_n | V_1) \cdot P(V_1)}}{P(V_2, V_3, \ldots, V_n)}$$

Assume $V_2, V_3, \ldots, V_n$ are conditionally independent given V_1

The conditional probabilities can be efficiently computed and stored, and they eliminate sparse data problem

Computational Tasks

- Evaluation

Holds always

$$P(V_1 = x_1, \ldots, V_n = x_n) =$$

$$= P(V_1 = x_1)P(V_2 = x_2 | V_1 = x_1)P(V_3 = x_3 | V_1 = x_1, V_2 = x_2) \ldots$$

$$P(V_n = x_n | V_1 = x_1, V_2 = x_2, \ldots, V_{n-1} = x_{n-1})$$

$$\overset{NB}{\approx} \boxed{P(V_1 = x_1)P(V_2 = x_2 | V_1 = x_1)P(V_3 = x_3 | V_1 = x_1) \ldots}$$

$$\boxed{P(V_n = x_n | V_1 = x_1)}$$

Fetch from lookup table

By Naïve Bayes assumption!

Computational Tasks

- **Assume**
  - ***Free*** and ***Caps*** are **input** variables
  - ***Spam*** is **output** variable
- **Naïve Bayes Assumption:**

$$\mathrm{P}(\mathit{Free}, \mathit{Caps}, \mathit{Spam}) = \mathrm{P}(\mathit{Spam}) \cdot \mathrm{P}(\mathit{Free}|\mathit{Spam}) \cdot \mathrm{P}(\mathit{Caps}|\mathit{Spam})$$

- **Say we have the following data:**

Free	*Caps*	*Spam*	Number of messages
Y	Y	Y	20
Y	Y	N	1
Y	N	Y	5
Y	N	N	0
N	Y	Y	20
N	Y	N	3
N	N	Y	2
N	N	N	49
		Total:	100

Computational Tasks

❖ **Compute the following tables:**

$Spam$	$\mathrm{P}(Spam)$
Y	$\frac{20+5+20+2}{100} = 0.47$
N	$\frac{1+0+3+49}{100} = 0.53$

Maximum Likelihood Estimation (MLE)

The parameters of the model are estimated using a corpus

$Free$	$Spam$	$\mathrm{P}(Free \mid Spam)$
Y	Y	$\frac{20+5}{20+5+20+2} \approx 0.5319$
Y	N	$\frac{1+0}{1+0+3+49} \approx 0.0189$
N	Y	$\frac{20+2}{20+5+20+2} \approx 0.4681$
N	N	$\frac{3+49}{1+0+3+49} \approx 0.9811$

$Caps$	$Spam$	$\mathrm{P}(Caps \mid Spam)$
Y	Y	$\frac{20+20}{20+5+20+2} \approx 0.8511$
Y	N	$\frac{1+3}{1+0+3+49} \approx 0.0755$
N	Y	$\frac{5+2}{20+5+20+2} \approx 0.1489$
N	N	$\frac{0+49}{1+0+3+49} \approx 0.9245$

Computational Tasks

- Say want to evaluate the following configuration:

$$
\begin{aligned}
&\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N, \textit{Spam} = N) = \\
&= \boxed{\mathrm{P}(\textit{Spam} = N) \cdot \mathrm{P}(\textit{Caps} = N | \textit{Spam} = N) \cdot \mathrm{P}(\textit{Free} = Y | \textit{Spam} = N)} \\
&\approx 0.53 \cdot 0.9245 \cdot 0.0189 \approx \boxed{0.0093}
\end{aligned}
$$

Fetch from lookup table

- Observe that in the Joint Distribution Model we had:

$$\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N, \textit{Spam} = N) = \boxed{0.00}$$

- This illustrates the fact that that the Naïve Bayes model is less amenable to the sparse data problem!

Computational Tasks

- **Simulation:**
  - Configurations are sampled by:
    - Sample the **output** variable based on its table
    - Sample the **input** variables using corresponding conditional tables
- **Inference – Marginalization**

$$\mathrm{P}(V_1 = x_1, \ldots, V_k = x_k) =$$

$$\boxed{\mathrm{P}(V_1 = x_1)\mathrm{P}(V_2 = x_2 | V_1 = x_1)\mathrm{P}(V_3 = x_3 | V_1 = x_1)\ldots}$$

$$\boxed{\mathrm{P}(V_k = x_k | V_1 = x_1)}$$

Fetch from lookup table

If the partial configuration includes the output variable, then compute marginal configuration as shown

Computational Tasks

❖ **Inference – Conditioning – Example**

Want to find the probability of message being spam given that header contains word "Free" and not all letters are uppercase

$$\boxed{\mathrm{P}(S=N|F=Y,C=N)} = \frac{\mathrm{P}(S=N,F=Y,C=N)}{\mathrm{P}(F=Y,C=N)}$$

Fetch from lookup table

$$\mathrm{P}(S=N,F=Y,C=N) =$$

By Naïve Bayes assumption!

$$= \boxed{\mathrm{P}(S=N)\mathrm{P}(F=Y|S=N)\mathrm{P}(C=N|S=N)}$$

$$= 0.53 \cdot 0.9245 \cdot 0.0189 \approx 0.093$$

Computational Tasks

❖ Inference – Conditioning – Example Continued

$$\begin{aligned}
\mathrm{P}(F=Y, C=N) &= \\
&= \mathrm{P}(S=Y, F=Y, C=N) + \mathrm{P}(S=N, F=Y, C=N) \\
&\approx \boxed{\mathrm{P}(S=Y)\mathrm{P}(F=Y|S=Y)\mathrm{P}(C=N|S=Y)} + 0.093 \\
&= 0.47 \cdot 0.5319 \cdot 0.1489 + 0.093 \\
&\approx 0.0465
\end{aligned}$$

By definition

Fetch from lookup table

- Finally,

$$\mathrm{P}(S=N|F=Y, C=N) = \frac{0.0093}{0.0465} \approx 0.2$$

Computational Tasks

- **Inference – Completion - Example**

By definition

$$\arg\max_{s\in\{Y,N\}} P(S=s \mid F=Y, C=N) = \arg\max_s \frac{P(S=s, F=Y, C=N)}{\boxed{P(F=Y, C=N)}} = ..$$

Does not depend on s

By Naïve Bayes Assumption

$$.. = \arg\max_s \boxed{P(S=s)P(F=Y \mid S=s)P(C=N \mid S=s)}$$

$s=Y$	$s=N$
$A(S)=0.0465$	$A(S)=0.0093$

$$\arg\max_s A(s) = Y$$

Computational Tasks

- **Learning**
  - Maximum Likelihood Estimation: The parameters are estimated using a corpus
- **Number of Parameters**
  - A Naïve Bayes model with n variables $V_1, \ldots, V_n$ is described with tables:

	parameters	constraints
table $\mathrm{P}(V_1)$	m	1
table $\mathrm{P}(V_2 \| V_1)$	m^2	m
table $\mathrm{P}(V_3 \| V_1)$	m^2	m
$\vdots$	$\vdots$	$\vdots$
table $\mathrm{P}(V_n \| V_1)$	m^2	m
sum	$m + (n-1)m^2$	$1 + (n-1)m$

Total: $O(m^2 n)$

Pros and Cons

❖ **Joint Distribution Model vs. Fully Independent Model**

Advantages

- **Efficient**

  $O(m^2 n)$ vs. $O(nm)$ vs. $O(m^n)$

  Lot less to compute than Joint Distribution Model!

- **No sparse data problem**

- **Surprisingly good performance (accuracy), e.g. in text classification**

Disadvantages

- **Can be over-simplifying!**
- **Practically, dependencies exist among attributes**
- **Cannot model more than one “output” variable**

Thank You!