Faculty of Computer Science, Dalhousie University *17-Oct-2023*

CSCI 4152/6509 — Natural Language Processing

Lecture 13: Naïve Bayes Model

Location: Rowe 1011 Instructor: Vlado Keselj
Time: 16:05 – 17:25

Previous Lecture

- P0 discussion: P-02
- Probabilistic modeling:
  - random variables, random models
  - full and partial model configurations
  - computational tasks in probabilistic modeling
- Joint distribution model
  - Spam example
- Fully independent model
- Naïve Bayes classification model
  - Assumption, definition
  - Graphical representation

Naïve Bayes Classification

- The classification formula becomes

$$\arg\max_{x_1} \frac{\mathrm{P}(V_2|V_1)\cdot\mathrm{P}(V_3|V_1)\cdot\ldots\cdot\mathrm{P}(V_n|V_1)\cdot\mathrm{P}(V_1)}{\mathrm{P}(V_2,V_3,\ldots,V_n)} =$$

$$\arg\max_{x_1} \mathrm{P}(V_2|V_1)\cdot\mathrm{P}(V_3|V_1)\cdot\ldots\cdot\mathrm{P}(V_n|V_1)\cdot\mathrm{P}(V_1)$$

- To calculate marginal probability in the denominator we use

$$\mathrm{P}(V_2,V_3,\ldots,V_n) = \sum_{V_1}\mathrm{P}(V_1,V_2,V_3,\ldots,V_n) =$$

$$\sum_{V_1}\mathrm{P}(V_2|V_1)\cdot\mathrm{P}(V_3|V_1)\cdot\ldots\cdot\mathrm{P}(V_n|V_1)\cdot\mathrm{P}(V_1)$$

Another way of deriving the Naïve Bayes assumption is the following:

$$\begin{aligned}
&\mathrm{P}(V_1=x_1,\ldots,V_n=x_n) = && (3)\\
&= \mathrm{P}(V_1=x_1)\mathrm{P}(V_2=x_2|V_1=x_1)\mathrm{P}(V_3=x_3|V_1=x_1,V_2=x_2)\ldots && (4)\\
&\quad \mathrm{P}(V_n=x_n|V_1=x_1,V_2=x_2,\ldots,V_{n-1}=x_{n-1}) && (5)\\
&\overset{\text{NB}}{\approx} \mathrm{P}(V_1=x_1)\mathrm{P}(V_2=x_2|V_1=x_1)\mathrm{P}(V_3=x_3|V_1=x_1)\ldots && (6)\\
&\quad \mathrm{P}(V_n=x_n|V_1=x_1) && (7)
\end{aligned}$$

Equality (3,4) holds always, and equality (5,6) is the Naïve Bayes assumption.

Summary of the Naïve Bayes Model

Naive Bayes assumption

P(V2,V3,...Vn|V1) = P(V2|V1) P(V3|V1) ... P(Vn|V1)

text features — class variable

Second way of expression Naive Bayes Assumption:

P(V1,V2,V3,...,Vn) = P(V1) P(V2,V3,..,Vn|V1) =
= P(V1) P(V2|V1) P(V3|V1) ... P(Vn|V1)

Naive Bayes Model is a set of tables

V1	P(V1)

V1	V2	P(V2|V1)

V1	Vn	P(Vn|V1)

(CPT −− Conditional Probability Tables)

Example: A Naïve Bayes Model for Spam Detection

In our spam detection example, the Naïve Bayes assumption is:

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

Hence, in order to create a Naïve Bayes model from our training 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

we calculate the following tables:

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

,

Caps	*Spam*	P(*Caps*|*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	$\frac{0+49}{1+0+3+49} \approx 0.9245$

, and

Free	*Spam*	P(*Free*|*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	$\frac{3+49}{1+0+3+49} \approx 0.9811$

.

The probability of a configuration in this model is calculated in the following way:

$$\begin{aligned} &\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N, \textit{Spam} = N) = \\ &\quad = \mathrm{P}(\textit{Spam} = N) \cdot \mathrm{P}(\textit{Caps} = N | \textit{Spam} = N) \cdot \mathrm{P}(\textit{Free} = Y | \textit{Spam} = N) \\ &\quad \approx 0.53 \cdot 0.9245 \cdot 0.0189 \approx 0.0093 \end{aligned} \tag{8}$$

12.1 Computational Tasks in the Naïve Bayes Model

We will cover the computational tasks in more details within the Bayesian Network in general.

1. Evaluation

The probability of a complete configuration is calculated using the Naïve Bayes assumption and table lookups. The formula (8) illustrates probability evaluation of a complete configuration: $\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N, \textit{Spam} = N)$

This example illustrates the fact that the Naïve Bayes model is less amenable to the sparse date problem than the joint distribution problem, in which the probability of this same configuration was estimated to be 0.

2. Simulation

Configurations are sampled by first sampling the output variable based on its table, and then the input variables using the corresponding conditional tables.

3. Inference

3.a) Marginalization. If the partial configuration includes the output variable, it can be shown that the marginal probability can be calculated using the following formula:

$$\begin{aligned} &\mathrm{P}(V_1 = x_1, \ldots, V_k = x_k) = \\ &\quad \mathrm{P}(V_1 = x_1)\mathrm{P}(V_2 = x_2 | V_1 = x_1)\mathrm{P}(V_3 = x_3 | V_1 = x_1) \ldots \\ &\quad \mathrm{P}(V_k = x_k | V_1 = x_1) \end{aligned}$$

3.b) Conditioning. Example:

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

Using Naïve Bayes assumption:

$$\begin{aligned} &\mathrm{P}(S = N, F = Y, C = N) = \\ &\quad = \mathrm{P}(S = N)\mathrm{P}(F = Y | S = N)\mathrm{P}(C = N | S = N) \\ &\quad = 0.53 \cdot 0.9245 \cdot 0.0189 \approx 0.0093 \end{aligned}$$

$$\begin{aligned}
&\mathrm{P}(F=Y, C=N) = \text{(by definition)} \\
&= \mathrm{P}(S=Y, F=Y, C=N) + \mathrm{P}(S=N, F=Y, C=N) \\
&\approx \mathrm{P}(S=Y)\mathrm{P}(F=Y|S=Y)\mathrm{P}(C=N|S=Y) + 0.0093 \\
&= 0.47 \cdot 0.5319 \cdot 0.1489 + 0.0093 \\
&\approx 0.0465
\end{aligned}$$

Finally,

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

3.c) Completion in the Naïve Bayes Model

Slide notes:

3.c) Completion in the NB Model

- Classification is the completion task:

$$\arg\max_{s \in \{Y,N\}} \mathrm{P}(S=s|F=Y, C=N)$$

- It works out that we calculate:

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

and

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

and choose the larger value.

Example:

$$\arg\max_{s \in \{Y,N\}} \mathrm{P}(S=s|F=Y, C=N) \overset{\textit{by definition}}{=} \arg\max_{s} \frac{\mathrm{P}(S=s, F=Y, C=N)}{\mathrm{P}(F=Y, C=N)}$$

$\mathrm{P}(F=Y, C=N)$ does not depend on s, hence

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

and by using Naïve Bayes assumption)

$$= \arg\max_{s} \underbrace{\mathrm{P}(S=s)\mathrm{P}(F=Y|S=s)\mathrm{P}(C=N|S=s)}_{A(s)}$$

For $s=Y$ $A(s=Y) \approx 0.0465$, and for $s=N$ $A(s=N) \approx 0.0093$; hence

$$\arg\max_{s} A(s) = Y$$

Learning

Maximum Likelihood Estimation: The parameters are estimated using a corpus.

12.2 Number of Parameters

A Naïve Bayes model with n variables V_1,... V_n is described with tables $\mathrm{P}(V_1)$, $\mathrm{P}(V_2|V_1)$, $\mathrm{P}(V_3|V_1)$, ..., $\mathrm{P}(V_n|V_1)$. These tables have constraints since each probability distribution must sum up to 1. If we assume that each variable can take one of m distinct values, then the number of parameters and constraints in required tables are:

	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	$m + (n-1)m^2$	$1 + (n-1)m$

Hence, the number of free parameters is $m + (n-1)m^2 - 1 - (n-1)m = O(m^2 n)$, which is not very large since the joint distribution model requires $O(m^n)$ parameters.

Pros and Cons of the Naïve Bayes Model

Some advantages (pros) of the Naïve Bayes Model are:
Efficiency: It is a relatively efficient method, with good running-time complexity for inference and small memory size.
No sparse data problem: Since the number of parameters is relatively small, there is usually sufficient data to train all parameters, and smoothing is relatively easy.
Performance: Even though it has a very strong and unrealistic independence assumption, the model frequently shows surprisingly good classification performance.

Some disadvantages (cons) of the Naïve Bayes Model are:
Too strong independence assumption: The strong independence assumption often affects performance for many domains. In other words, the model is too simplistic.
Only one "output" variable: The model is designed as a classification problem; i.e., it contains only one hidden, or output, variable; which value can be inferred. Many problems require that we infer the value of multiple variables, and the only way to apply Naïve Bayes model to those problems is to build separate models for all hidden variables. In that case we would not capture any inter-dependencies among those variables.

Additional Notes on Naïve Bayes Model

- Text classification: how do we choose features?
- Two options:
  - Bernoulli Naïve Bayes — binary variables for each word
  - Multinomial Naïve Bayes — variable for each word position
- Zero-probability problem
  - Smoothing using $+1$ or similar addition (Laplace smoothing)

The *Bernoulli Naïve Bayes* model uses a variable for each distinct word in the vocabulary, with values 1 if the word is present, or 0 if not. Training is done on per-document basis. The name comes from the Bernoulli distribution as defined in the probability theory, which is distribution of a random variable having value 1 with a probability p and 0 with the probability $q = 1 - p$. This is the distribution we use to model probability that a word is in a document of a given class.

The *Multinomial Naïve Bayes* model uses a variable for each word position, and the value of the variable is the actual word. All conditional probabilities for these variables are the same, but they are collected in one large table. The model is trained on one 'mega-document'; i.e., a document with concatenated all documents of a class. The model is named after the Multinomial distribution in the probability theory, which models the outcome of n repeated trials, where each trial can have one of k different results, with probabilities p_1, p_2, ..., p_k. In the Multinomial Naïve Bayes model, n is the length of a text, and individual trials are word positions, where words are taken from a vocabulary of size k.

12.3 Spam Example Summary

Let us take a look at a summary of the Spam Example for the three discussed models: Joint Distribution, Fully Independent, and Naïve Bayes model. In all three models, the initial training data was the same, represented in the following table:

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

The **Joint Distribution Model** is represented using a joint probability distribution table, learned from the training data as:

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

As an example, the conditional probability $\mathrm{P}(\textit{Spam} = Y | \textit{Free} = Y, \textit{Caps} = N)$ would be evaluated as:

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = Y | \textit{Free} = Y, \textit{Caps} = N) = \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N)} \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N) + \mathrm{P}(\textit{Spam} = N, \textit{Free} = Y, \textit{Caps} = N)} \\
&= \frac{0.05}{0.05 + 0.00} = 1.00
\end{aligned}$$

The **Fully Independent Model** is represented using a set of independent probability tables for all variables, learned from the training data as:

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

,

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

and

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

Using the same example, the conditional probability $\mathrm{P}(\textit{Spam} = Y|\textit{Free} = Y, \textit{Caps} = N)$ would be evaluated as:

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = Y|\textit{Free} = Y, \textit{Caps} = N) = \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N)} \\
&= \frac{\mathrm{P}(\textit{Spam} = Y) \cdot \mathrm{P}(\textit{Free} = Y) \cdot \mathrm{P}(\textit{Caps} = N)}{\mathrm{P}(\textit{Free} = Y) \cdot \mathrm{P}(\textit{Caps} = N)} \\
&= \mathrm{P}(\textit{Spam} = Y) = 0.47
\end{aligned}$$

The **Naïve Bayes Model** is represented using a set of conditional probability tables, learned from the training data as:

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

Caps	*Spam*	P(*Caps*|*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	$\frac{0+49}{1+0+3+49} \approx 0.9245$

Free	*Spam*	P(*Free*|*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	$\frac{3+49}{1+0+3+49} \approx 0.9811$

.

Using the same example, the conditional probability $\mathrm{P}(\textit{Spam} = Y|\textit{Free} = Y, \textit{Caps} = N)$ would be evaluated as:

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = Y|\textit{Free} = Y, \textit{Caps} = N) = \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Free} = Y, \textit{Caps} = N)} \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N) + \mathrm{P}(\textit{Spam} = N, \textit{Free} = Y, \textit{Caps} = N)}
\end{aligned}$$

We first calculate:

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N) = \\
&= \mathrm{P}(\textit{Spam} = Y) \cdot \mathrm{P}(\textit{Free} = Y|\textit{Spam} = Y) \cdot \mathrm{P}(\textit{Caps} = N|\textit{Spam} = Y) \\
&= 0.47 \cdot 0.5319 \cdot 0.1489 \approx 0.047248677
\end{aligned}$$

and

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = N, \textit{Free} = Y, \textit{Caps} = N) = \\
&= \mathrm{P}(\textit{Spam} = N) \cdot \mathrm{P}(\textit{Free} = Y|\textit{Spam} = N) \cdot \mathrm{P}(\textit{Caps} = N|\textit{Spam} = N) \\
&= 0.53 \cdot 0.0189 \cdot 0.9245 \approx 0.009260717
\end{aligned}$$

so finally, based on the above equation,

$$\begin{aligned}
&\mathrm{P}(\textit{Spam} = Y|\textit{Free} = Y, \textit{Caps} = N) = \\
&= \frac{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N)}{\mathrm{P}(\textit{Spam} = Y, \textit{Free} = Y, \textit{Caps} = N) + \mathrm{P}(\textit{Spam} = N, \textit{Free} = Y, \textit{Caps} = N)} \\
&= \frac{0.047248677}{0.047248677 + 0.009260717} \approx 0.836120752
\end{aligned}$$

13 N-gram Model

To understand better the significance of the N-gram Model, let us first introduce the *language modeling*—an important task in the probabilistic NLP. *Language Modeling* can be described as the task of building a probabilistic model that can estimate the probability of an arbitrary natural languate sentence; i.e., of the probabilisty P(sentence).

One application of this problem is in speech recognition. In speech recognition, we are interested in

$$\arg\max_{\text{sentence}} \mathrm{P}(\text{sentence}|\text{sound})$$

This is equal to:

$$\begin{aligned}\arg\max_{\text{sentence}} \mathrm{P}(\text{sentence}|\text{sound}) &= \arg\max_{\text{sentence}} \frac{\mathrm{P}(\text{sentence}, \text{sound})}{\mathrm{P}(\text{sound})} \\ &= \arg\max_{\text{sentence}} \mathrm{P}(\text{sentence}, \text{sound}) \\ &= \arg\max_{\text{sentence}} \mathrm{P}(\text{sound}|\text{sentence})\mathrm{P}(\text{sentence})\end{aligned}$$

It is easier to estimate $\mathrm{P}(\text{sound}|\text{sentence})$ than $\mathrm{P}(\text{sentence}, \text{sound})$, and it is done by an *acoustic model,* while $\mathrm{P}(\text{sentence})$ is estimated by a *language model.*

Slide notes:

Language Modeling
- Task of estimating probability of arbitrary utterance in a language
- Alternative task: Predicting the next token in a sequence: e.g., the next word or words, in a sentence, or next character or characters
- N-gram model: a "natural" model for this task

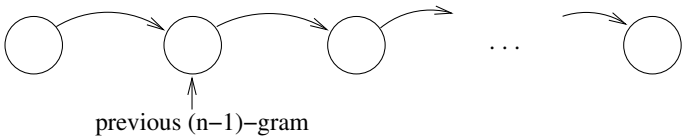
N-gram model is very simple and it is among the most successful models for language modelling; trigram ($n = 3$) word model in particular. In an n-gram model, we calculate joint distribution for all n-tuples of consecutive words (or characters). For example, in the trigram model, we count the number of occurrences of each triple of consecutive words from a corpus. Using this statistics, we can estimate the probability of arbitrary word w_3 following two given words w_1 and w_2: $\mathrm{P}(w_3|w_1w_2)$. It is useful to assign some small probability to unseen triples as well (using a technique called *smoothing*). If we use two "dummy" words '$\cdot$' at the beginning of each sentence, then the probability of arbitrary sentence can be calculated as:

$$\mathrm{P}(w_1w_2 \ldots w_n) = \mathrm{P}(w_1|\cdot\cdot)\mathrm{P}(w_2|w_1\cdot)\mathrm{P}(w_3|w_2w_1) \ldots \mathrm{P}(w_n|w_{n-1}w_{n-2})$$

N-gram Model: Notes

- Reading: Chapter 4 of [JM]
- Use of log probabilities
  - similarly as in the Naïve Bayes model for text
- Graphical representation

Graphical Representation



Use of log probabilities

Multiplying a large number of probability values, which are typically small, gives a very small result (close to zero), so in order to avoid floating-point underflow, we should use addition of logarithms of the probabilities in the model.