Lecture 12: Naïve Bayes Classifier, Evaluation Methods

Ling 1330/2330 Intro to Computational Linguistics
Na-Rae Han, 10/5/2023

Overview

- Text classification; Naïve Bayes classifier
  - Language and Computers: Ch.5 Classifying documents
  - NLTK book: Ch.6 Learning to classify text

- Evaluating the performance of a system
  - *Language and Computers*:
    - Ch.5.4 Measuring success, 5.4.1 Base rates
  - NLTK book: Ch.6.3 Evaluation
  - Cross-validation
  - Accuracy vs. precision vs. recall
  - F-measure

Given D, chance of Spam?

$$P(SPAM \mid D) = \frac{P(SPAM, D)}{P(D)} = \frac{P(SPAM, D)}{P(SPAM, D) + P(HAM, D)}$$

P(SPAM|D)

← The probability of *a given document D* being SPAM

= 1 - P(HAM|D)

← Can calculate from P(SPAM, D) and P(HAM, D)

hi there
if you want to
make quick cash

30% chance SPAM?
90%?

A bit of background

- P(A): the probability of A occurring
  - P(SPAM): the probability of having a SPAM document.
- P(A|B): **Conditional probability**

  the probability of A occurring, given that B has occurred
  - P(f1|SPAM): given a spam document, the probability of feature1 occurring.
  - P(SPAM|D): given a specific document, the probability of it being a SPAM.
- P(A, B): **Joint probability**

  the probability of A occurring *and* B occurring
  - Same as P(B, A).
  - If A and B are independent events, same as P(A)*P(B).

    If not, same as P(A|B)*P(B) and also P(B|A)*P(A).
  - P(D, SPAM): the probability of a specific document D occurring, and it being a SPAM.

A bit of background

- P(A, B): **Joint probability**

  the probability of A occurring *and* B occurring

  - Same as P(B, A).
  - If A and B are independent events, same as P(A)*P(B).

    If not, same as P(A|B)*P(B) and also P(B|A)*P(A).
  - P(D, SPAM): the probability of a specific document D occurring, and it being a SPAM.

Throwing two dice.
A: die 1 comes up with 6.
B: die 2 comes up with an even number.
➔A and B are independent.
➔P(A,B) = P(A) * P(B)
= 1/6 * 1/2 = 1/12

Throwing one die.
A: die comes up with 6.
B: die comes up with an even number.
➔A and B are NOT independent!
➔P(A,B) = P(A|B) * P(B)
= 1/3 * 1/2 = 1/6
= P(B|A) * P(A)
= 1 * 1/6 = 1/6

Bayes' Theorem

$$\boxed{❶\; P(B \mid A) = \frac{P(B, A)}{P(A)}} = \frac{P(A \mid B) * P(B)}{P(A)}$$

- B: Pitt closing, A: snowing
- P(B|A): probability of Pitt closing, given snowy weather
- P(B, A): probability of Pitt closing and snowing

❶: the probability of Pitt closing given it's snowing is equal to the probability of Pitt closing and snowing, divided by the probability of snowing.

Snow vs. Pitt, Bayes theorem style

❶
$$\boxed{P(B|A)=\frac{P(B,A)}{P(A)}}=\frac{P(A|B)*P(B)}{P(A)}$$

- B: Pitt closing, A: snowing
  - Last year, there were 15 snowy days; Pitt closed 4 days, 3 of which were snowy days.
- P(B|A): probability of Pitt closing, given snowy weather

  = P(B,A) / P(A)

  = (3/365) / (15/365)

  = 3/15 = 0.2
- P(B, A): probability of Pitt closing and snowing

  = 3/365

❶: the probability of Pitt closing given it's snowing is equal to the probability of Pitt closing and snowing, divided by the probability of snowing.

Snow vs. Pitt, Bayes theorem style

$$P(B \mid A) = \boxed{❷ \frac{P(B, A)}{P(A)} = \frac{P(A \mid B) * P(B)}{P(A)}}$$

- B: Pitt closing, A: snowing
- P(B|A): probability of Pitt closing, given snowy weather
- P(B, A): probability of Pitt closing and snowing

❷: <u>the probability of Pitt closing AND it's snowing</u> is equal to the probability of Pitt closing (=prior) multiplied by the probability of snowing given that Pitt is closed.

← Corollary of ❶! You get this by swapping A and B and solving for P(B,A)

Bayes' Theorem & spam likelihood

$$P(SPAM \mid D) = \frac{P(SPAM, D)}{P(D)} = \frac{P(SPAM, D)}{P(SPAM, D) + P(HAM, D)}$$

A document has to be either SPAM or HAM!

$$\begin{aligned} &P(SPAM, D) \\ &= P(D|SPAM) * P(SPAM) \\ &= P(SPAM) * P(D|SPAM) \\ &= P(SPAM) * P(f_1, f_2, \ldots, fn|SPAM) \\ &= P(SPAM) * P(f_1|SPAM) * P(f_2|SPAM) * \ldots * P(fn|SPAM) \quad ❷ \end{aligned}$$

- SPAM: document is spam, D: a specific document occurs
- P(SPAM|D): probability of document being SPAM, given a particular document
- P(SPAM, D): probability of D occurring and it being SPAM
- Which means: we can calculate P(SPAM|D) from

  P(SPAM, D) and P(HAM, D), which are calculated as ❷.

Probabilities of the entire document

H_1 "D is a SPAM" is closely related to P(D, SPAM):

The probability of document D occurring *and* it being a spam

$= P(SPAM) * P(D|SPAM)$

$= P(SPAM) * P(f_1, f_2, \ldots, f_n | SPAM)$ ❶

$= P(SPAM) * P(f_1|SPAM) * P(f_2|SPAM) * \ldots * P(f_n|SPAM)$ ❷

- We have all the pieces to compute this.
- "Bag-of-words" assumption ❶
- "Naïve" Bayes because ❷ assumes **feature independence**.

If all we're going to do is rule between SPAM and HAM, we can simply compare P(D, SPAM) and P(D, HAM) and <u>choose one with higher probability</u>.

- But we may also be interested in answering:

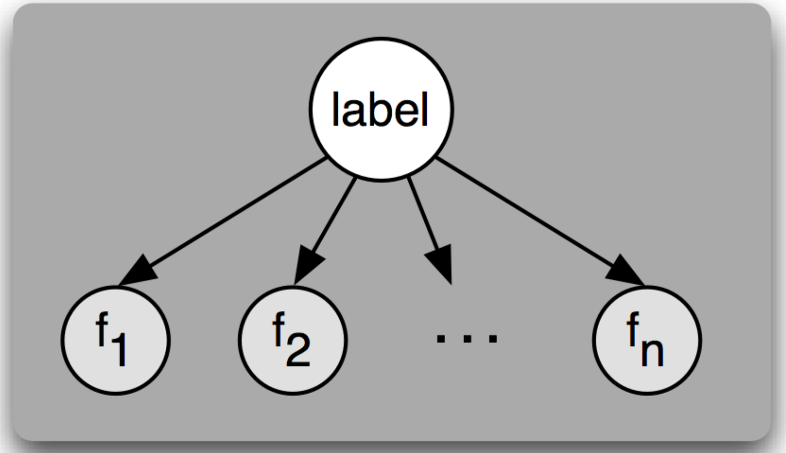
"Given D, what are the *chances* of it being a SPAM? 70%? 5%?"

← This is P(SPAM|D).

Naïve Bayes Assumption

- Given a label, a set of features f_1, f_2, … , f_n are generated with different probabilities
- The features are **independent** of each other; f_x occurring does not affect f_y occurring, etc.

  ➔ **Naïve Bayes Assumption**



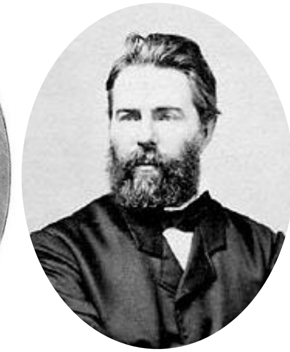
- This **feature independence assumption** simplifies combining contributions of features; you just **multiply** their probabilities:

  $P(f_1,f_2,\ldots,f_n|L) = P(f_1|L)*P(f_2|L)*\ldots*P(f_n|L)$

🡐 "Naïve" because features are often inter-dependent.

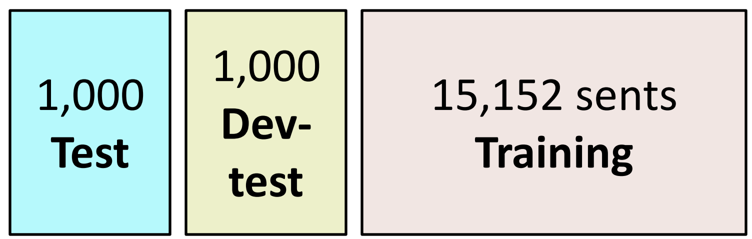
🡐 `f1:'contains-`*`linguistics`*`:YES'` and `f2:'contains-`*`syntax`*`:YES'` are not independent.

Homework 4: Who Said It?



- Jane Austen or Herman Melville?
  - *I never met with a disposition more truly amiable.*
  - *But Queequeg, do you see, was a creature in the transition stage -- neither caterpillar nor butterfly.*
  - *Oh, my sweet cardinals!*

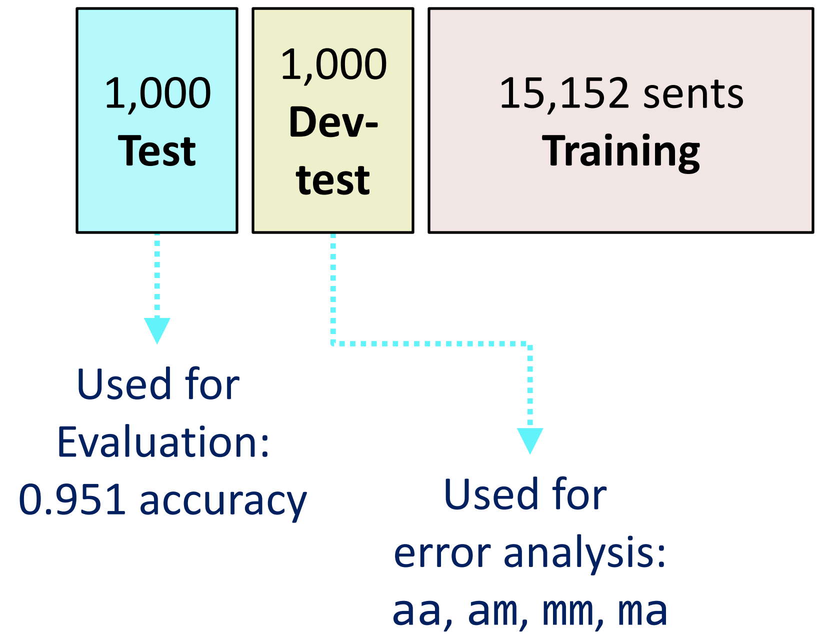
- Task: build a Naïve Bayes classifier and explore it
- Do three-way partition of data:
  - test data
  - development-test data
  - training data



`whosaid`: a Naïve Bayes classifier

- How did the classifier do?
  - 0.951 accuracy on the test data, using a fixed random data split.
- Probably outperformed your expectation.
- What's behind this high accuracy? How does the NB classifier work?

  ➔ HW4 PART [B]
- How good is 0.951?



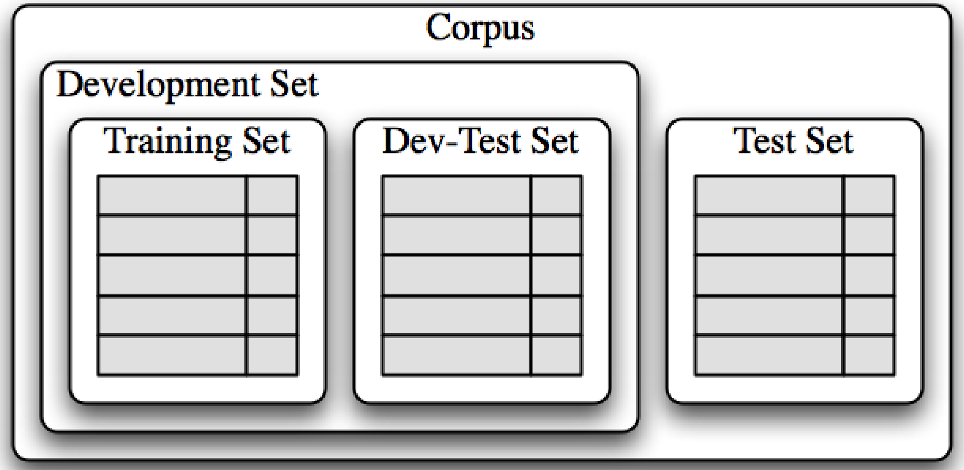
Common evaluation setups

- **Training** vs. **testing** partitions
  1. Training data ← classifier is trained on this section
  2. Testing data ← classifier's performance is measured

- Training, testing, + **development-testing**

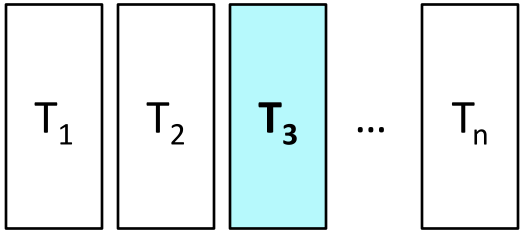
  + 3. Development testing data

  ←In feature engineering, researcher can error-analyze the data to improve performance



Cross validation

- But what if our training/testing split is somehow biased?
  - ➔ We could randomize
    - ➔ or use cross-validation.



- ***n*-fold cross validation method**
  - Partition the data set into equally sized *n* sets
  - Conduct *n* rounds of training-testing, each using 1 partition as testing and the rest *n-1* partitions for training
  - And then take an average of the *n* accuracy figures
  - ← More reliable accuracy score. Performance evaluation is less dependent on a particular training-testing split
  - ← We can see how widely performance varies across different training sets

Confusion matrices

- When classifying among 3+ labels, **confusion matrices** can be informative
- L1 classification of ESL essays:

predicted L1 \ true L1	ARA	DEU	FRA	HIN	ITA	JPN	KOR	SPA	TEL	TUR	ZHO
ARA	57	0	3	9	1	8	2	9	6	10	2
DEU	6	79	5	2	7	4	2	5	0	1	3
FRA	2	7	60	3	8	0	3	5	1	1	3
HIN	5	1	1	46	3	1	2	7	19	6	4
ITA	5	4	10	2	67	2	3	14	0	4	3
JPN	2	1	4	0	5	72	20	0	0	2	6
KOR	1	0	0	0	1	2	51	6	1	8	6
SPA	6	4	8	12	1	3	2	45	11	6	1
TEL	10	1	0	17	3	2	3	1	53	2	1
TUR	5	2	6	7	1	6	5	5	7	53	8
ZHO	1	1	3	2	3	0	7	3	2	7	63

Accuracy as a measure

- **Accuracy**: of all labeling decisions that a classifier made, how many of them are *correct*?
  - POS tagger
  - Name gender identifier
  - `whosaid`: Austen/Melville author classifier
  - Document topic identifier
  - Movie review classifier: positive/neg. ("sentiment classifier")

Accuracy as a measure

- **Accuracy**: of all labeling decisions that a classifier made, how many of them are *correct*?
- Interpreting accuracy numbers
  - A movie review sentiment classifier tests 85% accurate. Is this good or bad?
    - What if it turns out 80% movie reviews are positive?
    - How about 60%?
  - A document topic identifier tests 60% accurate. Good or bad?
    - What if 55% of documents are on "Politics"?
    - What if there are as many as 20 different topics, and the largest category only accounts for 10% of the data?

  ← These questions cannot be answered without considering **base probabilities** (**priors**).

Base probabilities

- **Base probabilities (priors)**

  The probability of a randomly drawn sample to have a label x
  - `whosaid`? POS tagger? Disease test?
  - `whosaid`: 'melville' has a higher prior than 'austen'
  - POS tagger: 'Noun' may have the highest prior than other tags
  - Disease test: 'Negative' is typically much higher than 'Positive'
- **Base rate neglect**
  - A cognitive bias humans have
  - We tend to assume that base probabilities are equal
- **Base performance**
  - The "absolute bottom line" for system performances

    = <u>the highest base probability</u>

  ex. POS tagger: if 20% of all words are 'Noun', then the worst-performing system can be constructed which blindly assigns 'Noun' to every word, whose accuracy is 20%.

When accuracy isn't a good measure

- A **medical test for a disease** is 96% accurate. Good or bad?
  - What if 95% of population is free of the disease?
- A **grammatical error detector** is 96% accurate. Good or bad?
  - Suppose 95% of all sentences are error-free.

    ← <u>Accuracy alone doesn't tell the whole story.</u>
- We are interested in:
  - Of all "ungrammatical" flags the system raises, what % is correct?

    ← This is the **precision** rate.
  - Of all actual ungrammatical sentences, what % does the system correctly capture as such?

    ← This is the **recall** rate.

Outcome of a diagnostic test

- A grammatical error detector as a diagnostic test
  - Positive: has grammatical error
  - Negative: is error-free

		Real	
		Has grammatical error	Is error-free
Test	positive	**True positives**	False positives
	negative	False negatives	**True negatives**

- **Accuracy**:

$$(Tp + Tn) / (Tp + Tn + Fp + Fn)$$

🡐 When the data is predominantly error-free (high base rate), this is not a meaningful measure of system performance.

Outcome of a diagnostic test

- A grammatical error detector as a diagnostic test
  - Positive: has grammatical error
  - Negative: is error-free

		Real	
		Has grammatical error	Is error-free
Test	positive	❶ **True positives**	False positives
	negative	False negatives	**True** negatives

  - **Precision**:

    Rate of "True positives" out of all positive rulings (❶)

    = Tp / (Tp + Fp)

Outcome of a diagnostic test

- A grammatical error detector as a diagnostic test
  - Positive: has grammatical error
  - Negative: is error-free

		Real	
		Has grammatical error	Is error-free
Test	positive	❷ **True positives**	False positives
	negative	False negatives	**True** negatives

- **Recall**:

  Rate of "True positives" out of all actual positive cases (❷)

  = Tp / (Tp + Fn)

Precision vs. recall

- **Precision** and **recall** are in a trade-off relationship.
  - Highly precise grammatical error detector:
    Ignores many lower-confidence cases → drop in recall
  - High recall (captures as many errors as possible):
    many non-errors will also be flagged → drop in precision

- In developing a real-world application, picking the right trade-off point between the two is an important usability issue.
  - A **grammar checker** for general audience (MS-Word, etc)
    - Higher precision or higher recall?
  - Same, but for English learners.
    - Higher precision or higher recall?

F-measure

- **Precision** and **recall** are in a trade-off relationship.
  ← Both measures should be taken into consideration when evaluating performance
- **F-measure**
  - Also called F-score, F_1 score
  - An overall measure of a test's accuracy:
    Combines *precision* (P) and *recall* (R) into a single measure
  - Harmonic mean →

$$F_1 = \frac{2PR}{P+R}$$

  - Best value: 1,
    worst value: 0
  - = average if P=R,
    < average if P and R different

Wrapping up

- HW 4 Part A, B due on Tue
  - **Don't procrastinate**! Part B is more complex.

- Next class (Tue)
  - HW4 review
  - Midterm review

- Midterm exam on Thursday → NEXT SLIDE

Midterm exam: what to expect

- 10/12 (Thursday)
  - 75 minutes.
  - At LMC's PC Lab (G17 CL)
- Exam format:
  - Closed book. All pencil-and-paper.
  - Topical questions: "what is/discuss/analyze/find out/calculate…"
  - Bring your calculator! →

- A letter-sized cheat sheet allowed.
  - Front and back.
  - Hand-written only.