# Using Probabilities in Understanding Sentiment

Aida Nematzadeh
nematzadeh@berkeley.edu

# Language Learning and Probabilities

Learning word meanings:

*A dax!*

***dax* means dog**

*Look at the dax!*

# Language Learning and Probabilities

Anticipating the next words:

She is drinking coffee lemonade

a chair doogh

In a cold winter day, she is drinking hot cocoa

**Using knowledge given the situation: hot drink is better on a cold day.**

lemonade

# Combining Evidence & Knowledge

In a cold winter day, she is drinking hot cocoa $h_1$

evidence

lemonade $h_2$

The Bayes rule:

likelihood prior

posterior

$$P(h|e) = \frac{P(e|h)P(h)}{\sum_{h'}(e|h')P(h')}$$

P(*e*)

# Evidence & Knowledge in Action

In a cold winter day, she is drinking

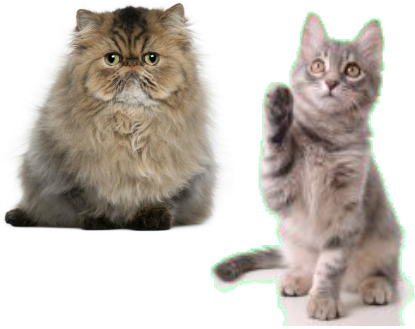
**evidence**

hot cocoa $h_1$

lemonade $h_2$

Let's calculate the probability of each hypothesis!

# When Hypotheses Are Classes



$H_1$: the cat class

classification

$H_2$: the dog class

Many language processing tasks are classification!

# Sentiment Analysis

Positive/negative orientation (sentiment) of text:

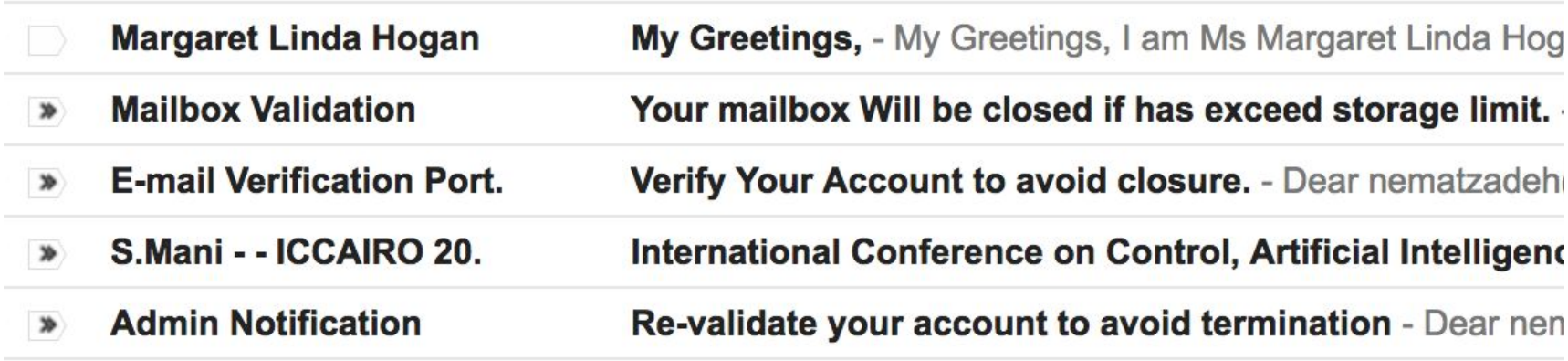
- Book, restaurant, movie and product reviews
- Political text

It's just gorgeous, like a flipbook made of dreamy vintage postcards that are somehow about contemporary life in Los Angeles.

The cinematography and special effects are fantastic, but don't actually compensate for a weak storyline, and forgettable musical numbers.

# Spam Detection

Classify email to spam and non-spam



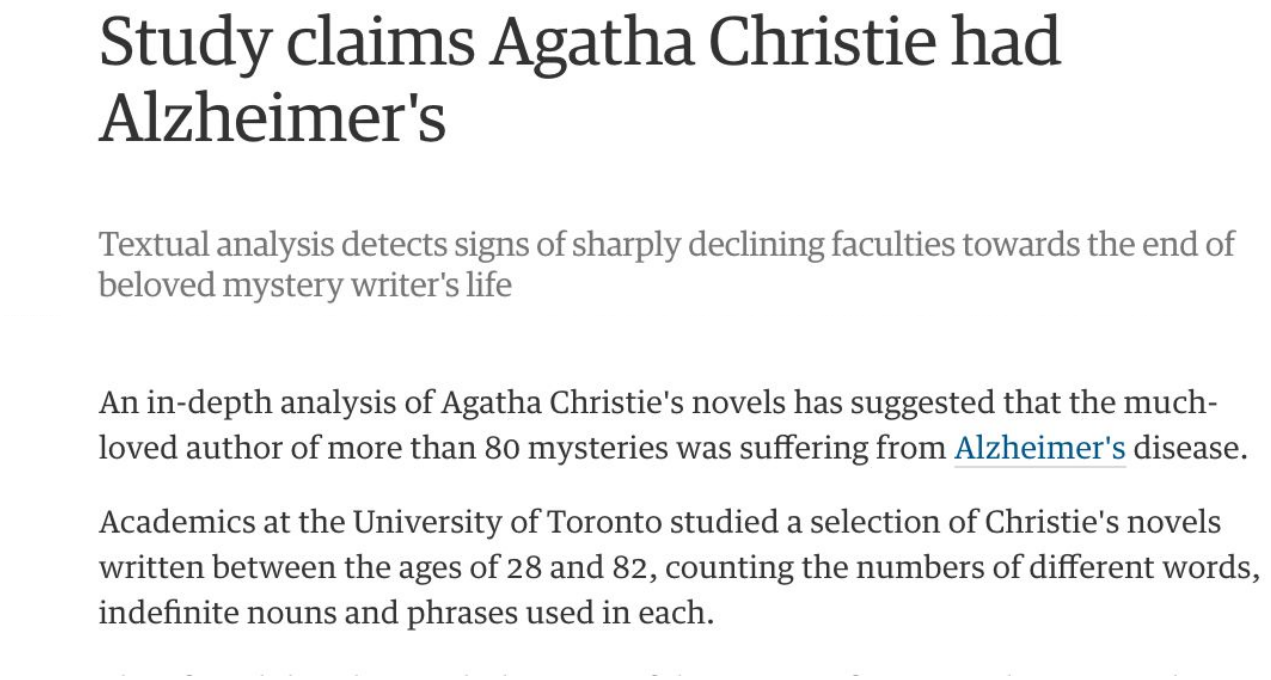
| | | |
|---|---|---|
| | **Margaret Linda Hogan** | **My Greetings,** - My Greetings, I am Ms Margaret Linda Hog |
| » | **Mailbox Validation** | **Your mailbox Will be closed if has exceed storage limit.** - |
| » | **E-mail Verification Port.** | **Verify Your Account to avoid closure.** - Dear nematzadeh |
| » | **S.Mani - - ICCAIRO 20.** | **International Conference on Control, Artificial Intelligenc** |
| » | **Admin Notification** | **Re-validate your account to avoid termination** - Dear nen |

"Dear winner"; "Click this link"; "Urgent: send me your credit card information"; ...

# Authorship Attribution

Find the text author and author's characteristics:

- Gender, age, etc



## Study claims Agatha Christie had Alzheimer's

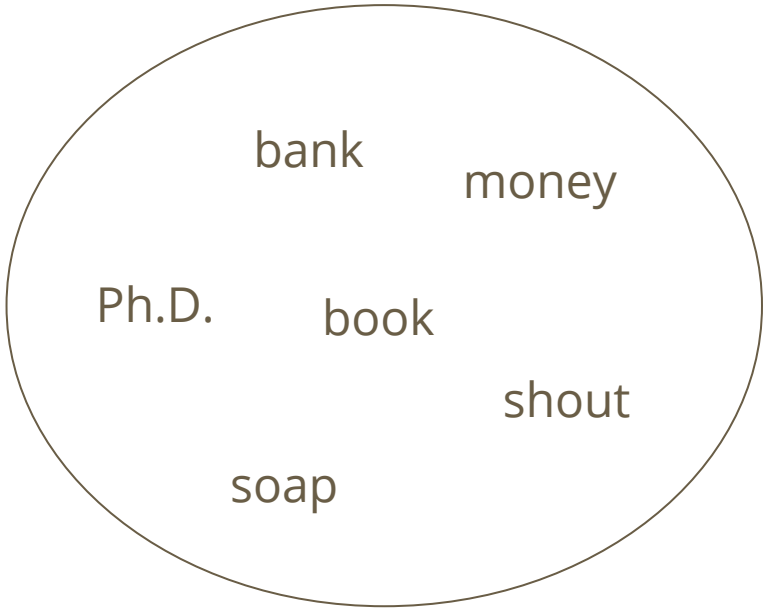
Textual analysis detects signs of sharply declining faculties towards the end of beloved mystery writer's life

An in-depth analysis of Agatha Christie's novels has suggested that the much-loved author of more than 80 mysteries was suffering from Alzheimer's disease.

Academics at the University of Toronto studied a selection of Christie's novels written between the ages of 28 and 82, counting the numbers of different words, indefinite nouns and phrases used in each.

# Learning Word Representations

Word2Vec models:


Class 1: positive examples

Class 2: negative examples

# Classification: Formalism

Given an input & fixed classes $C=c_1, c_2, \ldots, c_M$, find:

- the predicted class $c_i$
- The probability of each class

**documents**

Supervised training: uses data points & their gold-standard labels, $(d_1, c_1), (d_2, c_2), \ldots, (d_N, c_N)$

Goal: Find the correct class for the new data point

# Classification Algorithms

Generative models

prior

$$P(c|d) = \frac{P(d|c)P(c)}{P(d)}$$

posterior

Full distribution

Can use prior info

Naive Bayes

Discriminative models

$$P(d|c)$$

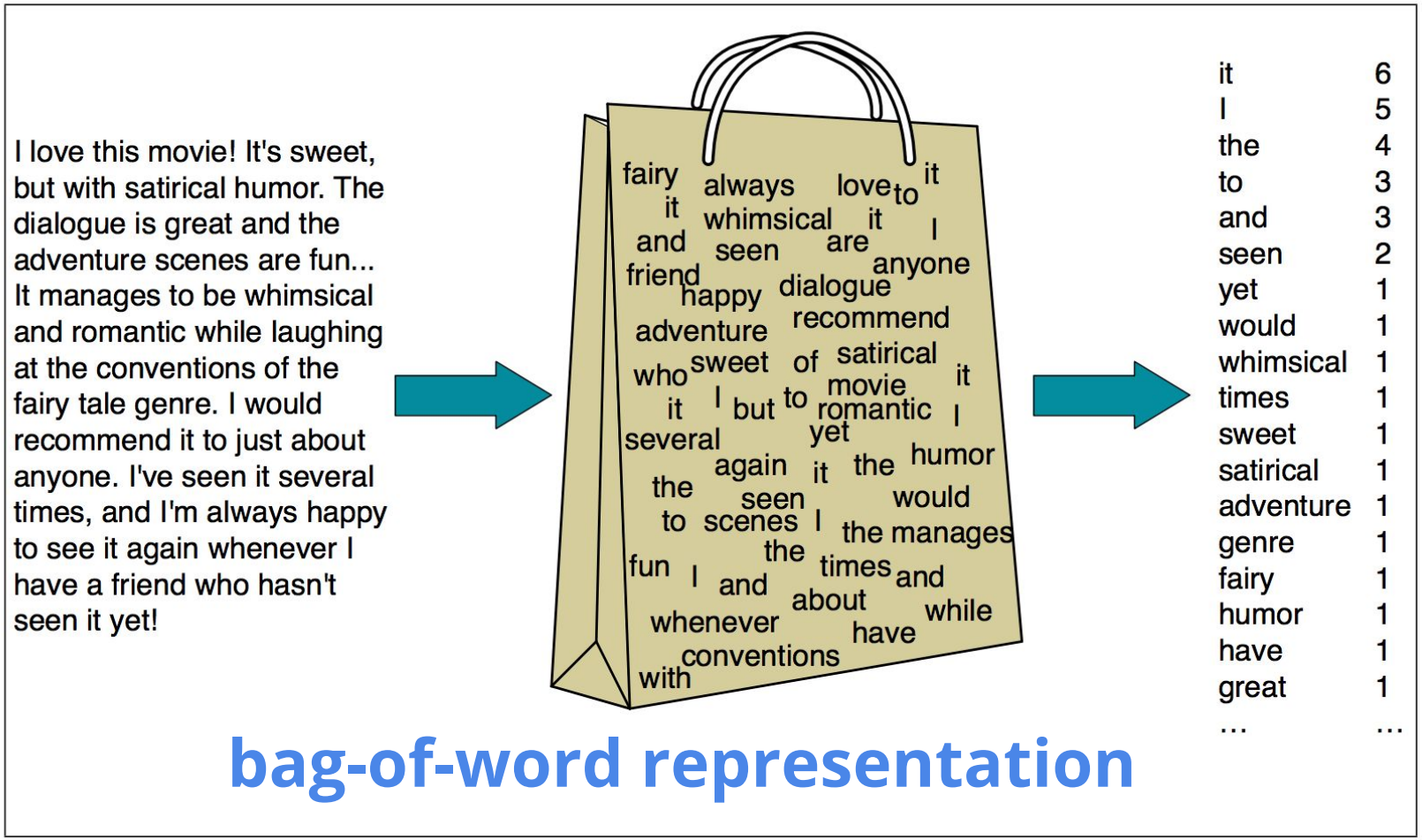
likelihood

Easier to train

Used more in practice

Support Vector Machine & Word2Vec

# Naive Bayes Classifier: Input



# Naive Bayes Classifier: Formulation

Find the best/correct class for the document:

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}}\ P(c|d)$$ posterior

*i.e.*, the class with the maximum posterior prob.

**M.A.P estimation**

*What are the disadvantages of this?*

How do we calculate P(*c*|*d*)? Let's expand it.

# Naive Bayes Classifier: Formulation

Find the best/correct class for the document:

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}}\ P(c|d)$$ posterior

likelihood prior

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}}\ \frac{P(d|c)P(c)}{P(d)}$$

**Is P(*d*) of important?**

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}}\ P(d|c)P(c)$$

# Naive Bayes Classifier: Formulation

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} \; \underbrace{P(d|c)}_{\text{likelihood}} \; \underbrace{P(c)}_{\text{prior}}$$

How do we calculate P(*d* | *c*)?

Represent the document as a set of features.

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} \; \overbrace{P(f_1, f_2, \ldots, f_n|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

# Naive Bayes Classifier: Formulation

$$\hat{c} = \underset{c \in C}{\operatorname{argmax}} \overbrace{P(f_1, f_2, \ldots, f_n|c)}^{\text{likelihood}} \overbrace{P(c)}^{\text{prior}}$$

How do we calculate P($f_1$,$f_2$,…,$f_n$|$c$)?

Naive Bayes assumption: Independent features.

$$P(f_1, f_2, \ldots, f_n|c) = P(f_1|c).P(f_2|c).\ldots.P(f_n|c)$$

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}}\ P(c) \prod_{f \in F} P(f|c)$$

# Naive Bayes on Documents

Given a new document:

$$\text{positions} \leftarrow \text{all word positions in test document}$$

$$c_{NB} = \underset{c \in C}{\text{argmax}}\ P(c) \prod_{i \in positions} P(w_i|c)$$

Practical trick:

$$c_{NB} = \underset{c \in C}{\text{argmax}}\ \log P(c) + \sum_{i \in positions} \log P(w_i|c)$$

# Let's Look an Example!

| | Cat | Documents |
|---|---|---|
| Training | - | just plain boring |
| | - | entirely predictable and lacks energy |
| | - | no surprises and very few laughs |
| | + | very powerful |
| | + | the most fun film of the summer |
| Test | ? | predictable with no fun |

$$c_{NB} = \underset{c \in C}{\operatorname{argmax}}\ P(c) \prod_{i \in positions} P(w_i|c)$$

# Let's Look an Example!

| | Cat | Documents |
|---|---|---|
| Training | - | just plain boring |
| | - | entirely predictable and lacks energy |
| | - | no surprises and very few laughs |
| | + | very powerful |
| | + | the most fun film of the summer |
| Test | ? | predictable with no fun |

$$\hat{P}(c) = \frac{N_c}{N_{doc}}$$

$$\hat{P}(w_i|c) = \frac{count(w_i, c)}{\sum_{w \in V} count(w, c)}$$

# Add-one Smoothing

What is P("predictable" | +)?

$$\hat{P}(w_i|c) = \frac{count(w_i, c) + 1}{\sum_{w \in V} (count(w, c) + 1)}$$

$$= \frac{count(w_i, c) + 1}{(\sum_{w \in V} count(w, c)) + |V|}$$

Adding prior probability to unseen words.

Unknown words (words not in V) are removed.

# Naive Bayes Demo

## Affective Text: Data Annotated for Emotions and Polarity

- Affective Text is a data set consisting of 1000 test headlines and 200 development headlines, each of them annotated with the six Eckman emotions and the polarity orientation. [download] (July 13, 2007).
  - Carlo Strapparava and Rada Mihalcea, SemEval-2007 Task 14: Affective Text, in Proceedings of the 4th International Workshop on the Semantic Evaluations (SemEval 2007), Prague, Czech Republic, June 2007. [pdf]
  - Read more about the task here.

# Improving Sentiment Analysis

Binarize the frequency of a word in a document.

**Four original documents:**

- − it was pathetic the worst part was the boxing scenes
- − no plot twists or great scenes
- + and satire and great plot twists
- + great scenes great film

**After per-document binarization:**

- − it was pathetic the worst part boxing scenes
- − no plot twists or great scenes
- + and satire great plot twists
- + great scenes film

| | NB Counts | | Binary Counts | |
|---|---|---|---|---|
| | + | − | + | − |
| and | 2 | 0 | 1 | 0 |
| boxing | 0 | 1 | 0 | 1 |
| film | 1 | 0 | 1 | 0 |
| great | 3 | 1 | 2 | 1 |
| it | 0 | 1 | 0 | 1 |
| no | 0 | 1 | 0 | 1 |
| or | 0 | 1 | 0 | 1 |
| part | 0 | 1 | 0 | 1 |
| pathetic | 0 | 1 | 0 | 1 |
| plot | 1 | 1 | 1 | 1 |
| satire | 1 | 0 | 1 | 0 |
| scenes | 1 | 2 | 1 | 2 |
| the | 0 | 2 | 0 | 1 |
| twists | 1 | 1 | 1 | 1 |
| was | 0 | 2 | 0 | 1 |
| worst | 0 | 1 | 0 | 1 |

What are other possible scores?

# Improving Sentiment Analysis

Dealing with negation:

*I didn't love the food* **vs** *I loved the food.*

Add a prefix after negation (n't, not, no, never)

*I didn't NOT-love NOT-the NOT-food:*

# Improving Sentiment Analysis

Use **sentiment lexicon**s when there is no enough labeled training data.

- **L**inguistic **I**nquiry and **W**ord **C**ount (LIWC)
- NRC Word-Emotion Association Lexicon

\+ : *admirable, beautiful, confident, dazzling, ecstatic, favor, glee, great*

− : *awful, bad, bias, catastrophe, cheat, deny, envious, foul, harsh, hate*

Use Two features -- positive vs negative words

# Naive Bayes as a Language Model

Model prob. of generating sentences in the lang.

NB: set of class-specific unigram lang. models.

Likelihood of a sentence: $$P(s|c) = \prod_{i \in positions} P(w_i|c)$$

How can we calculate P(*s*)?

# Evaluating Classifier Performance

Consider a binary detection task.

- Label text spam (*positive*) or ~spam (*negative*).

*Gold labels*: human labels used as ground truth.

- Gold label is either spam (*true*) or ~spam (*false*).

Need *metrics* to quantify classifier's performance.

# Evaluation Metrics: Precision/Recall

*Accuracy* is a natural metric, but rarely used.

- Problem with unbalanced classes. (Why?)

**Precision**: correctly labeled + / all labeled +.

**Recall**: correctly labeled + / all truly +.

**F-measure** combines both: harmonic mean.

- Weighs min of two more heavily.
- Who to invite to your valentine's day party?

# Evaluation Metrics