Intro to Text Processing
Lecture 3

Behrang Mohit

Admin

• Homework 1: will posted this afternoon
  – Due next Wednesday, Nov 6, 10 AM
  – Start Early

• Quiz-1: Next Monday
  – Closed note: All materials through Wednesday class
  – Short: 15:20 mins
Review

• Overview of natural language processing (NLP)

• The Hierarchy of text collections
  – Corpus: Large collection of text documents
    • Plain-text, Annotated corpus
  – Documents
  – Sentences
  ...

Review

• Documents
  – Plain text, structured (HTML, XML, ...)

• Sentences, words and characters
  – Finding sentence boundaries
  – Words, types and lemma
  – ASCII vs. Unicode
Today: Probabilities in Text Processing

• Questions:
  – Is there any mathematical pattern behind human languages?
  – If so, how can we benefit from it to build text processing systems.
  • Text categorization

Language as a probabilistic system

• Statistical population: individuals that follow a distribution (e.g. CMUQ staff)
• Attributes (e.g. age, gender, etc.)
• Events: individual with certain attributes
Language as a probabilistic system

- Statistical population: individuals that follow a distribution (e.g. CMUQ staff)
- Attributes (e.g. age, gender, etc.)
- Events: individual with certain attributes

- Language as a probabilistic system
  - Population: words
  - Attributes: token, types, length
  - Event: words with length 3, with certain lemma

Word frequency

- Absolute frequency: number of occurrence of units with a certain value given an attribute
  - Depends on the length of document/corpus
  - e.g.: Number of words whose length is 4
  - Hapax: terms which occur only once in a text
Distributions

• A variable can belong to different distributions:
  – Uniform dist.: values are uniformly distributed
  – Gaussian dist.: concentration around one point (mean) and then decrease.

• Question:
  – What is distr. of word frequency in text?
Intuition: not uniform!

Frequency spectrum

• Distribution curve of the vocabulary
  – X: Frequency classes (word occurring 1, 2, ... times)
  – Y: Number of words with frequency i
Frequency table

| word type | Freq (f) | Rank (r) | $f \times z$ | word type | Freq (f) | Rank (r) | $f \times z$
|-----------|---------|---------|-------------|-----------|---------|---------|-------------
| the       | 3332    | 1       | 3332        | turned    | 51      | 200     | 10200       
| and       | 2972    | 2       | 5944        | name      | 21      | 400     | 8400        
| a         | 1775    | 3       | 5235        | comes     | 16      | 500     | 8000        
| he        | 877     | 10      | 8770        | group     | 13      | 600     | 7800        
| but       | 410     | 20      | 8400        | lead      | 11      | 700     | 7700        
| be        | 294     | 30      | 8820        | friends   | 10      | 800     | 8000        
| there     | 222     | 40      | 8880        | begin     | 9       | 900     | 9100        
| one       | 172     | 50      | 8600        | family    | 8       | 1000    | 8000        
| never     | 124     | 80      | 9920        | could     | 2       | 4000    | 8000        
| two       | 104     | 100     | 10400       | applause   | 1       | 8000    | 8000        |
Zipf Law

\[ rank(w) \approx \frac{1}{f_{rc}(w)} \]

• An almost fixed reverse relation between the ranking of a word and its frequency

• Intuitive explanation
  – minimum effort: people tend to minimize their vocabulary and use a small set of words frequently
Generalization of Zipf law

- Frequency of access to webpages
  - Wikipedia
- Income distributions
- Size of earthquakes
- Music notes

Today: Probabilities in Text Processing

- Questions:
  - Is there any mathematical pattern behind human languages?
    - YES
  - If so, how can we benefit from it to build text processing systems.
Probabilities in text

• Neural indications
  – Use probabilities to model language knowledge

• Probability: A function that assigns a number $[0,1]$
  • Event $A$ vs. the large event space $S$.
    $$ P(A) = \frac{|A|}{S} $$
Probabilities in text

• Neural indications
  – Use probabilities to model language knowledge
  – Probability: A function that assigns a number [0,1]
    • Event A vs. the large event space S.
    \[ P(A) = \frac{|A|}{S} \]
  – Dice examples:
    • Probability of getting 6
    • Probability of getting below 5
    • Probability of an even number

Probabilities

• Complex events:
  – Dice gets 2 or 4: \( p(2) + p(4) \)
  \[ P(A \cup B) = P(A) + P(B) \]

• Joint events:
  – Events take place concurrently
  – Dice gets 3 in the first try and 4 in the 2\(^{nd}\) try?
  \[ p(A,B) = p(A) \cdot p(B) \]
Estimating probabilities

- Approximate a probability with relative frequency
- **Prior** probabilities vs. **empirical** probabilities

- Linguistics events are not uniform:
  - Not all words occur uniformly
    - Zipf law
  - Translation probabilities
  - ...

Machine Learning

- Use computing to collect statistics from data (text), infer and learn some patterns and predict statistics for unseen data
  - E.g. Translation systems:
    - The probability of word $x$ being translated to word $y$
Components of a learning framework

• Data for training
  – Underlying task (e.g. text classification)
• Statistical model
• Data for testing

Labeled data

• Human-labeled data
  – a.k.a. human annotated or gold-standard data
    • Named entities
    • Part of speech
    • Translation
    • Regular text ➔ language model
Components of a learning framework

- Data for training
- Statistical model
- Data for testing

Modeling

- Model: an abstract representation of a phenomenon (usually complicated ones)
  - Example: a small model car represents some of the visual and physical specification of a real car.
    - Color, light,
    - Not all: Engine, trunk space
Modeling

• Model: an abstract representation of a phenomenon (usually complicated ones)
  – Example: a small model car represents some of the visual and physical specification of a real car.
    • Color, light,
    • Not all: Engine, trunk space

• Modeling in science and technology
  – Human intelligence ➔ Language

Modeling text and language concepts

• Neural perspective
• Simplification in modeling
  – Translation: one word to one word
  – Bigram model: A word depends only on the previous
  – Opinions: positive vs. negative
Learning from data

• **Training**
  – Input: training data
  – Output: A statistical model

• **Testing**
  – Input: test data (usually gold-standard)
  – Output: ML’s predictions or annotation

• **Tuning:**
  – Input: a held out set of training data
  – Output: A tuned statistical model

Features

• Learner does memorize the training data
  – e.g. Named entities
  – e.g question answering

• Rich features help the learner to generalize patterns:
  – Capitalization ➔ *Named entity*
  – Who question ➔ *Person* answer

• Feature extraction and engineering
Supervised vs. unsupervised

- Supervised learning uses labeled training data
- Unsupervised learning uses unlabeled (raw) training data
- Semi-supervised: mix

Classification Examples

- Classification: Learner predicts the class label for each point (instance)
Classification for text processing

- Gender identification:
  - Female or male name?
- Document classification:
  - Sport vs. politics vs. ...
- Author classification:
  - This is Shakespeare writing vs. Thomas Friedman

Problem: Document Classification

Also known as: Text categorization
Document Classification

A work once thought to be by Vincent van Gogh but later dismissed has now been confirmed as an authentic painting by the Dutch master.

Still Life With Meadow Flowers and Roses, originally considered a Van Gogh, has belonged to a Dutch museum since 1974.

But doubts crept in due to the painting style and the unusual canvas size and it was discredited in 2003.

f(Document) \rightarrow \text{Topic or class}

f: \text{doc} \rightarrow C

Supervised Learning

• Training a statistical learner (classifier)
  – Supervised learner
    • Labeled training data
Learning from the labeled data

• Labeled data allows me to collect probabilities:
  – What is the probability of any document being about Sport?
  – What is the probability of ...

• Learning is about collecting the relevant statistics from the data and building a model.

Naïve Bayes classification

• Simple “naïve” classification based on the Bayes rule
• The **bag of words** representation of a document
A handful of Saudi women have taken to the streets in their cars on a day of collective protest against the ban on female drivers. Several videos of women driving have been posted online despite official warnings that women who took part risked sanctions. Some women received warning phone calls from men purporting to be from the interior ministry.
Formalizing the framework

• **Input:**
  - a document \(d\)
  - a fixed set of classes \(C = \{c_1, c_2, \ldots, c_k\}\)
  - A training set of \(m\) hand-labeled documents
    - \((d_j, c_j), \ldots, (d_m, c_m)\)

• **Output:**
  - a learned classifier \(f: d \rightarrow c\)
  - Needs to be tested on unseen set of labeled documents

Maximum A Priori (MAP)

• The class that has the highest probability
  - Probability (doc being Business): 25%
  - **Probability (doc being Art): 65%**
  - Probability (doc being Politics): 10%

\[
\hat{c} = \arg\max_{c \in C} Pr(c)
\]
Conditional probability

- Given a condition, what is the probability of an event? $P(\text{event} | \text{condition})$
  - $P(\text{rain} | \text{cloudy})$
  - $P(\text{word}=\text{President} | \text{topic}=\text{politics})$

$$P(E|C) = \frac{P(E,C)}{P(C)}$$

Bays Rule

- Reversing the order of event and condition
  - Allows us to compute the probability easier

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$
Applying the Bayes’ Rule

• For a document $d$ and a class $c$

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

Naïve Bayes Classifier

$$c_{MAP} = \arg \max_{c \in C} P(c \mid d)$$

$$= \arg \max_{c \in C} \frac{P(d \mid c)P(c)}{P(d)}$$

$$= \arg \max_{c \in C} P(d \mid c)P(c)$$
Naïve Bayes Classifier (2)

$$c_{MAP} = \arg \max_{c \in C} P(d | c) P(c)$$

What does $p(d | c)$ mean?

Features

• Document: not sufficient information
  – $Pr(\text{document } \ldots)$

• Need to represent the document to the learner:
  – Features
    • Length, author, date, links
    • Words
Naïve Bayes Classifier (3)

\[ c_{MAP} = \arg\max_{c \in C} P(d \mid c)P(c) \]

\[ = \arg\max_{c \in C} P(x_1, x_2, \ldots, x_n \mid c)P(c) \]

Representing a document with a set of features

Word features in classifier
**Naïve Bayes Independence Assumptions**

\[ P(x_1, x_2, \ldots, x_n \mid c) \]

- **The “naïve” assumptions**: Words’ positions don’t matter
  - Bag of words
- **Conditional Independence**: Feature probabilities \( P(x_i \mid c_j) \) are independent given the class \( c \).

\[ P(x_1, \ldots, x_n \mid c) = P(x_1 \mid c) \cdot P(x_2 \mid c) \cdot P(x_3 \mid c) \cdot \ldots \cdot P(x_n \mid c) \]

---

**Multinomial Naïve Bayes Classifier**

\[ c_{\text{MAP}} = \arg\max_{c \in C} P(x_1, x_2, \ldots, x_n \mid c) P(c) \]

\[ c_{\text{NB}} = \arg\max_{c \in C} P(c_j) \prod_{x \in X} P(x \mid c) \]
Final Naïve Bayes formulation

positions ← all word positions in test document

\[ c_{NB} = \arg\max_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j) \]

Two Naïve Bayes parameters

\[ c_{NB} = \arg\max_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j) \]

- Parameters: important learning statistics
- \( Pr(c) \): Prior probability of classes
  - \( Pr(C= \text{sport}) \)?
- \( Pr(x_i | c) \)?
Important parameters

• $\Pr(x_i | c)$
  – $P(x_i=\text{election} | \text{class} = \text{politics})$
  – $P(x_i=\text{soccer} | \text{class} = \text{sport})$

• How to collect:
  – For each class (e.g. sport), collect all the documents
    • $N =$ total number of words in sport document
    • $N_{\text{soccer}}$: frequency of word="soccer")
    • $\Pr(x_i=\text{soccer} | \text{sport}) = \frac{N_{\text{soccer}}}{N}$

Big picture

• Machine Learning model
  – A model for document classification
  – Formula is used to predict the class for a document.
    • Training: collect the statistics from the data
    • Testing: Use the statistics to predict

• Modeling assumptions:
  • No overlap between topics
  • Words are independent of each other for a given topic
Problem: unseen word

• What if a word in the test document is not seen in the training?
  
  \[
  - P(x_i = "Instagram" | c = "politics")
  \]

\[
c_{NB} = \arg\max_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j)
\]

Smoothing

• Reserve a tiny fraction for unseen events
Smoothing the Naïve Bayes

- Laplace (add-1):
  \[ \hat{P}(x_i | c_j) = \frac{N(X_i = x_i, C = c_j) + 1}{N(C = c_j) + k} \]
  - Applied to text:
  \[ \hat{P}(w | c) = \frac{\text{count}(w, c) + 1}{\text{count}(c) + |V|} \]
  - Total count of \(X_i\)

Problem: Small probabilities

\[ c_{NB} = \arg\max_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j) \]

- Probabilities: \([0, 1]\)
  - Problem = ?
Problem: Small probabilities

\[ c_{NB} = \arg \max_{c_j \in C} P(c_j) \prod_{i \in \text{positions}} P(x_i | c_j) \]

- Probabilities: [0, 1]
  - Problem: underflow
- Solution: Work in the log space

\[ c_{NB} = \arg \max_{c_j \in C} \log P(c_j) + \sum_{i \in \text{positions}} \log P(x_i | c_j) \]

Training a Naïve Bayes classifier

- From training corpus, extract Vocabulary
- Calculate \( P(c_j) \) terms
  - For each \( c_j \) in \( C \) do
    \[ \text{docs}_j \leftarrow \text{all docs with class } c_j \]
  \[ P(c_j) \leftarrow \frac{|\text{docs}_j|}{|\text{total # documents}|} \]
- Calculate \( P(w_k | c_j) \) terms
  - \( \text{Text}_j \leftarrow \text{single doc containing all } \text{docs}_j \)
  - For each word \( w_k \) in Vocabulary
    \[ n_k \leftarrow \text{# of occurrences of } w_k \text{ in } \text{Text}_j \]
  \[ P(w_k | c_j) \leftarrow \frac{n_k + \alpha}{|\text{Vocabulary}|} \]
Testing Naïve Bayes

- For a given document, perform the following computation for every topic and chose the one with the highest value

\[
c_{NB} = \arg\max_{c_j \in C} \log P(c_j) + \sum_{i \in \text{positions}} \log P(x_i | c_j)
\]

Other usages of Naïve Bays

- Gender and Author prediction
- Spam filtering
  - Given a document, which class (Spam vs. ok)
- Named entity recognition
  - Given a word: Is this part of a name or not
**Evaluation**

- Usually we split the labeled (gold-standard) data into:
  - 90% for training
  - 10% for testing

- **Accuracy:** What portion of documents got the right class

---

**Cross Validation**

- The training and testing split can be biased
  - Maybe lucky and easier instances go to the test
- Split the data into pools

![Diagram](train train train train test)
Cross Validation

- The training and testing split can be biased
  – Maybe lucky and easier instances go to the test
- Split the data into pools
  train  train  train  test  test
- Rotate: train, test/evaluate
Cross Validation

- The training and testing split can be biased
  - Maybe lucky and easier instances go to the test
- Split the data into pools
- Rotate: train, test/evaluate
- Average the evaluations

Review

- Document classification
  - Assigning class label to a document
  - Supervised learning
- What happens during the learning?
  - Collect the statistics from the labeled data
- What is in the model?
  - Prior and conditional probabilities