Features and classification

CSC401/2511 – Natural Language Computing – Spring 2024
Lecture 5 Gerald Penn, Sean Robertson, Raed Saqur
University of Toronto
Lecture 5 overview

• Today:
  • Quick introduction to Text Classification
  • **Feature extraction** from text.
    • How to pick the right features?
    • Grammatical ‘parts-of-speech’.
      • (even when nobody is speaking)
  • **Classification** overview

• Some slides *may* be based on content from Bob Carpenter, Dan Klein, Roger Levy, Josh Goodman, Dan Jurafsky, and Christopher Manning.
Quick Intro to Text Classification

From Technology Upskilling Machine Learning Software Foundations by En-Shiun Annie Lee
Features

• Feature: *n.* A measurable **variable** that is (or *should be*) **distinctive** of something we want to model.

• We often choose features to **classify** something.
  • e.g., an emotional, whiny **tone** is likely to indicate that the speaker is not professional, scientific, nor political.
  • Note that in neural networks, e.g., ‘**features**’ often refer to something distinctive but not usually **nameable**.

• We often need **various, heterogeneous** features to adequately model something, e.g. tone plus aspects of their grammar.
Example: Feature vectors

- Values for several features of an observation can be put into a single vector.

<table>
<thead>
<tr>
<th></th>
<th># proper nouns</th>
<th># 1st person pronouns</th>
<th># commas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Damien Fahey</td>
<td>2</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Faux John Madden</td>
<td>5</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jim Gaffigan</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>
Feature vectors

- Features should be useful in **discriminating** between categories.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Counts</td>
<td>First person pronouns</td>
</tr>
<tr>
<td></td>
<td>Second person pronouns</td>
</tr>
<tr>
<td></td>
<td>Third person pronouns</td>
</tr>
<tr>
<td></td>
<td>Coordinating conjunctions</td>
</tr>
<tr>
<td></td>
<td>Past-tense verbs</td>
</tr>
<tr>
<td></td>
<td>Future-tense verbs</td>
</tr>
<tr>
<td></td>
<td>Commas</td>
</tr>
<tr>
<td></td>
<td>Colons and semi-colons</td>
</tr>
<tr>
<td></td>
<td>Dashes</td>
</tr>
<tr>
<td></td>
<td>Parentheses</td>
</tr>
<tr>
<td></td>
<td>Ellipses</td>
</tr>
<tr>
<td></td>
<td>Common nouns</td>
</tr>
<tr>
<td></td>
<td>Proper nouns</td>
</tr>
<tr>
<td></td>
<td>Adverbs</td>
</tr>
<tr>
<td></td>
<td>Wh-words</td>
</tr>
<tr>
<td></td>
<td>Modern slang acronyms</td>
</tr>
<tr>
<td></td>
<td>Words all in upper case (at least 2 letters long)</td>
</tr>
<tr>
<td>Counts</td>
<td>Average length of sentences (in tokens)</td>
</tr>
<tr>
<td>Counts</td>
<td>Average length of tokens, excluding punctuation tokens (in characters)</td>
</tr>
<tr>
<td>Counts</td>
<td>Number of sentences</td>
</tr>
</tbody>
</table>

**Higher** values → this person is referring to themselves (to their opinion, too?)

**Higher** values → looking forward to (or dreading) some future event?

**Lower** values → this tweet is more formal. Perhaps not overly sentimental?
Different features for different tasks

• **Alzheimer’s disease** involves atrophy in the brain.
  • Excessive **pauses** (acoustic disfluencies),
  • Excessive **word type repetition**, and
  • Simplistic or **short** sentences.
    • ‘**function words**’ like *the* and *an* are often **dropped**.
• To **diagnose** Alzheimer’s disease, one might measure:
  • **Proportion** of utterance spent in **silence**.
  • **Entropy** of **word type** usage.
  • **Number** of **word tokens** in a sentence.
    • **Number of prepositions** and **determiners** (explained shortly).
Features in Sentiment Analysis

- **Sentiment analysis** can involve detecting:
  - Stress or frustration in a conversation.
  - Interest, confusion, or preferences. Useful to marketers.
    - e.g., ‘got socks for xmas wanted #ps5 fml’
  - Deceit. e.g., ‘Let’s watch Netflix and chill.’
- Complicating factors include *sarcasm*, *implicitness*, and a subtle spectrum from negative to positive opinions.
- Useful features for sentiment analyzers include:
  - Trigrams.
  - First-person **pronouns**.

What does this mean?

Pronouns? Prepositions? Determiners?
Pre-processing

• **Pre-processing** involves preparing your data to make feature extraction easier or more valid.
  • E.g., *punctuation* likes to press up against words. The sequence “example,” should be counted as **two** tokens – not one.
  • We separate the punctuation, as in “example , ”.

• **There is no perfect pre-processor.**
  Mutually exclusive approaches can often **both** be justified.
  • E.g., Is *Newfoundland-Labrador* **one** word type or **two**?
    Each answer has a unique implication for splitting the dash.
  • Often, *noise-reduction* removes some information.
  • Being **consistent** is important.
Parts-of-speech (PoS)

• Linguists like to group words according to their distribution in building grammatical sentences.
  • This is similar to grouping Lego by their shapes.

• **Part-of-speech**: *n.* lexical category or morphological class.

Nouns collectively constitute a part-of-speech (called *Noun*)
## Example parts-of-speech

<table>
<thead>
<tr>
<th>Part of Speech</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Noun</td>
<td>is usually a <strong>person,</strong> <strong>place,</strong> <strong>event,</strong> or <strong>entity</strong>.</td>
<td><em>chair, pacing, monkey, breath.</em></td>
</tr>
<tr>
<td>Verb</td>
<td>is usually an <strong>action</strong> or <strong>predicate</strong>.</td>
<td><em>run, debate, explicate.</em></td>
</tr>
<tr>
<td>Adjective</td>
<td>modifies a <strong>noun</strong> to further describe it.</td>
<td><em>orange, obscene, disgusting.</em></td>
</tr>
<tr>
<td>Adverb</td>
<td>modifies a <strong>verb</strong> to further describe it.</td>
<td><em>lovingly, horrifyingly, often</em></td>
</tr>
</tbody>
</table>
## Example parts of speech

<table>
<thead>
<tr>
<th>Part of Speech</th>
<th>Description</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preposition</td>
<td>Often specifies aspects of space, time, or means.</td>
<td>around, over, under, after, before, with</td>
</tr>
<tr>
<td>Pronoun</td>
<td>Substitutes for nouns; referent typically understood in context.</td>
<td>I, we, they</td>
</tr>
<tr>
<td>Determiner</td>
<td>Logically quantify words, usually nouns.</td>
<td>the, an, both, either</td>
</tr>
<tr>
<td>Conjunction</td>
<td>Combines words or phrases.</td>
<td>and, or, although</td>
</tr>
</tbody>
</table>
Content categories

• Some PoSs convey content labels more than function or linguistic structure.
  • Usually nouns, verbs, adjectives, adverbs.
  • **Content** categories are usually multifarious.
    • e.g., there are more **nouns** than **prepositions**.
  • **New** content words are continually **added**
    e.g., an **app**, **to google**, **to misunderstand**
  • Some **archaic** content words go **extinct**.
    e.g., **fumificate**, v., (1721-1792),
    **frenigerent**, adj., (1656-1681),
    **melanochalcographer**, n., (c. 1697).
Function categories

• Some PoSs are ‘glue’ that holds others together.
  • E.g., prepositions, determiners, conjunctions.
  • **Functional** PoS usually cover a **small** and **fixed** number of word types (i.e., a ‘**closed class**’).

• Their **semantics** depend on the contentful words with which they’re used.
  • E.g., *I’m on time* vs. *I’m on a boat*
Grammatical “features”

• There are several grammatical features that can be associated with words:
  • Case
  • Person
  • Number
  • Gender

• These features can restrict other words in a sentence.
Other features of nouns

• Proper noun: named things (e.g., “they’ve killed Bill!”)
• Common noun: unnamed things (e.g., “they’ve killed the bill!”)

• Mass noun: divisible and uncountable (e.g., “butter” split in two gives two piles of butter – not two ‘butters’)

• Count noun: indivisible and countable. (e.g., a “pig” split in two does not give two pigs)
Agreement

• Parts-of-speech *should* match (i.e., *agree*) in certain ways.

• **Articles** ‘have’ to agree with the **number** of their **noun**
  • e.g., “*these* pretzels are making me thirsty”
  • e.g., “*a* winters are coming”

• **Verbs** ‘have’ to agree (at least) with their **subject** (in English)
  • e.g., “*the dogs eats* the gravy” no number agreement
  • e.g., “Yesterday, *all my troubles seem* so far away” bad tense – should be past tense *seemed*
  • e.g., “*Can you handle me the way I are*?”
Tagging
PoS tagging

• **Tagging:** *v.g.* the process of **assigning a part-of-speech** to each word in a sequence.

• E.g., using the ‘**Penn treebank**’ tag set (see appendix):

<table>
<thead>
<tr>
<th>Word</th>
<th>The</th>
<th>nurse</th>
<th>put</th>
<th>the</th>
<th>sick</th>
<th>patient</th>
<th>to</th>
<th>sleep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tag</td>
<td>DT</td>
<td>NN</td>
<td>VBD</td>
<td>DT</td>
<td>JJ</td>
<td>NN</td>
<td>IN</td>
<td>NN</td>
</tr>
</tbody>
</table>
Ambiguities in parts-of-speech

• Word types can have many parts-of-speech.
  • E.g., back:
    • The back/JJ door (adjective)
    • On its back/NN (noun)
    • Win the voters back/RB (adverb)
    • Promise to back/VB you in a fight (verb)

• We want to determine the appropriate tag for a given token in its context.
Why is tagging useful?

• First step towards many practical purposes.
  • **Speech synthesis**: how to pronounce text
    • I’m *content/JJ* vs. the *Content/NN*
    • I *object/VBP* vs. the *Object/NN*
    • I *lead/VBP* (“l iy d”) vs. it’s *lead/NN* (“l eh d”)
  • **Information extraction**:  
    • Help to find names and relations.
  • **Machine translation**:  
    • Help to identify phrase boundaries
  • **Explainability?**
Tagging as classification

- We have access to a **sequence of observations** and are expected to decide on the best assignment of a **hidden variable**, i.e., the PoS

<table>
<thead>
<tr>
<th>Hidden variable</th>
<th>Observation</th>
</tr>
</thead>
<tbody>
<tr>
<td>NN</td>
<td>she</td>
</tr>
<tr>
<td>VB</td>
<td>promised</td>
</tr>
<tr>
<td>VBN</td>
<td>to</td>
</tr>
<tr>
<td>JJ</td>
<td>back</td>
</tr>
<tr>
<td>NN</td>
<td>the</td>
</tr>
<tr>
<td>VB</td>
<td>bill</td>
</tr>
</tbody>
</table>

Observation sequence: she promised to back the bill
Reminder: Bayes’ Rule

\[ P(X|Y) = \frac{P(X)}{P(Y)} P(Y|X) \]
Statistical PoS tagging

- Determine the **most likely** tag sequence $t_{1:n}$ by:

$$\arg\max_{t_{1:n}} P(t_{1:n} | w_{1:n}) = \arg\max_{t_{1:n}} \frac{P(w_{1:n} | t_{1:n})P(t_{1:n})}{P(w_{1:n})}$$

$$= \arg\max_{t_{1:n}} \frac{P(w_{1:n} | t_{1:n})P(t_{1:n})}{P(w_{1:n})}$$

$$\approx \arg\max_{t_{1:n}} \prod_{i}^{n} P(w_i | t_i)P(t_i | t_{i-1})$$

By Bayes’ Rule

Only maximize numerator

Assuming independence

Assuming Markov
Those are hidden Markov models!

• We’ll see these soon...

Image sort of from *2001: A Space Odyssey* by MGM pictures
Word likelihood probability $P(w_i | t_i)$

- **VBZ** (verb, 3rd person singular present) is likely *is*.
- Compute $P(is | VBZ)$ by **counting** in a corpus that has **already** been **tagged**:

$$P(w_i | t_i) = \frac{\text{Count}(w_i \text{ tagged as } t_i)}{\text{Count}(t_i)}$$

  e.g.,
  $$P(is | VBZ) = \frac{\text{Count}(is \text{ tagged as } VBZ)}{\text{Count}(VBZ)} = \frac{10,073}{21,627} = 0.47$$
Tag-transition probability $P(t_i | t_{i-1})$

- Will/MD the/DT chair/NN chair/?? the/DT meeting/NN from/IN that/DT chair/NN?

a) MD → DT → NN → VB → ...
   - Will → the → chair → chair

b) MD → DT → NN → NN → ...
   - Will → the → chair → chair
Lecture Review Slide

• What are some examples of Text Classification
• **What are features?**
  • What are unique features for the specific tasks of sentiment analysis versus spam detection?
  • What are some words with multiple POS tags?
• Compute Baye’s rule for the POS tagging for an example.
Classification
General process

1. We gather a big and relevant **training** corpus.
2. We learn our **parameters** (e.g., probabilities) from that corpus to build our **model**.
3. Once that model is fixed, we use those probabilities to evaluate **testing** data.
General process

• Often, training data consist of 80% to 90% of the available data.
  • Often, some subset of this is used as a validation/development set.

• Testing data are not used for training but often come from the same corpus.
  • It often consists of the remaining available data.
  • Sometimes, it’s important to partition speakers/writers so they don’t appear in both training and testing.
  • But what if we just partitioned (un)luckily??
Better process: *K*-fold cross-validation

- **K-fold cross validation**: *n.* splitting all data into *K* partitions and iteratively testing on each after training on the rest (report means and variances).

<table>
<thead>
<tr>
<th>Part 1</th>
<th>Part 2</th>
<th>Part 3</th>
<th>Part 4</th>
<th>Part 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iteration 1</td>
<td></td>
<td></td>
<td></td>
<td>: Err1 %</td>
</tr>
<tr>
<td>Iteration 2</td>
<td></td>
<td></td>
<td>: Err2 %</td>
<td></td>
</tr>
<tr>
<td>Iteration 3</td>
<td></td>
<td>: Err3 %</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iteration 4</td>
<td>: Err4 %</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Iteration 5</td>
<td>: Err5 %</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*Testing Set* - *Training Set*
(Some) Types of classifiers

• **Generative** classifiers model the data.
  • Parameters set to maximize likelihood of training data.
  • We can *generate* new observations from these.
    • e.g., hidden Markov models

Vs.

• **Discriminative** classifiers emphasize **class boundaries**.
  • Parameters set to minimize error on training data.
    • e.g., support vector machines, decision trees.

• *...What do class boundaries look like in the data?*
Binary and linearly separable

- Perhaps the easiest case.
- Extends to dimensions $d \geq 3$, line becomes (hyper-)plane.
$N$-ary and linearly separable

- A bit harder – random guessing gives $\frac{1}{N}$ accuracy (given equally likely classes).
- We can **logically combine** $N - 1$ binary classifiers.
Class holes

• Sometimes it can be impossible to draw any lines through the data to separate the classes.

• *Are those troublesome points noise or real phenomena?*
The kernel trick

- We can sometimes linearize a non-linear case by moving the data into a higher dimension with a **kernel function**.

  E.g.,

  \[
  S = \frac{\sin(\sqrt{x^2 + y^2})}{\sqrt{x^2 + y^2}}
  \]

Now we have a linear decision boundary, \( S = 0! \)
Capacity and over/under-fitting

- A central challenge in machine learning is that our models should **generalize** to unseen data, so we need to set our (hyper-)parameters appropriately.

From Goodfellow
Capacity and over/under-fitting

• A central challenge in machine learning is that our models should **generalize** to unseen data, so we need to set our (hyper-)parameters appropriately.
Bias and Variance

High bias

High variance

Total Error

Error

Variance

Bias

Model Complexity
Let’s summarize a few of the classifiers from Assignment 1
Decision Tree

From Technology Upskilling Machine Learning Software Foundations by En-Shiun Annie Lee
Trees!

(The ... larch.)
Aspects of ID3

• **ID3** (iterative dichotomiser 3) is an algorithm invented by Ross Quinlan to produce decision trees from data.
• ID3 tends to build **short trees** since at each step we are removing the maximum amount of entropy possible.
• ID3 trains on the **whole training set** and does not succumb to issues related to **random initialization**.

• ID3 can **over-fit** to training data.
• Only **one attribute is used at a time** to make decisions
• It can be difficult to use **continuous** data, since many trees need to be generated to see where to break the continuum.
Random Forests

- **Random forests** *n.pl.* are *ensemble* classifiers that produce $K$ decision trees, and output the *mode* class of those trees.
  - Can support continuous features.
  - Can support non-binary decisions.
  - Support cross-validation.

- The component trees in a random forest must differ.
  - Sometimes, decision trees are *pruned* randomly.
  - Usually, different trees accept different *subsets of features*.

*That’s a good idea – can we choose the best features in a reasonable way?*
Support vector machines (SVMs)

- In binary linear classification, two classes are assumed to be separable by a line (or plane). However, many possible separating planes might exist.

- Each of these blue lines separates the training data.
  - *Which line is the best?*
Support vector machines (SVMs)

• The **margin** is the width by which the boundary could be **increased** before it hits a training datum.

• The **maximum margin linear classifier** is the linear classifier with the maximum margin.

• The **support vectors** (indicated) are those data points against which the margin is pressed.

• The bigger the margin – the less sensitive the boundary is to error.
Support vector machines (SVMs)

• The width of the margin, $M$, can be computed by the angle and displacement of the planar boundary, $x$, as well as the planes that touch data points.

• Given an initial guess of the angle and displacement of $x$ we can compute:
  • whether all data is correctly classified,
  • The width of the margin, $M$.
• We update our guess by **quadratic programming**, which is semi-analytic.
Support vector machines (SVMs)

• The maximum margin helps SVMs generalize to situations when it’s impossible to linearly separate the data.
  • We introduce a parameter that allows us to measure the distance of all data not in their correct ‘zones’.
  • We simultaneously maximize the margin while minimizing the misclassification error.
  • There is a straightforward approach to solving this system based on quadratic programming.
Support vector machines (SVMs)

• SVMs generalize to higher-dimensional data and to systems in which the data is non-linearly separable (e.g., by a circular decision boundary).
  • Using the kernel trick (from before) is common.

• Many binary SVM classifiers can also be combined to simulate a multi-category classifier.

• (Still) one of the most popular off-the-shelf classifiers.
Naïve Bayes and SoftMax

- Broadly, Bayesian probability conceives of probability *not* as frequency of some phenomenon occurring, but rather as an expectation related to our own certainty.
- Given an observation $x$, **Naïve Bayes** simply chooses the class $c \in C$ that maximizes $P(c \mid x)$.
  - This can be done in many ways.

$$\arg\max_{c} P(c \mid x) = \frac{P(c)}{P(x)} P(x \mid c)$$

Estimate the $P(\cdot)$ using Gaussians, or...
Bayesian Classifier

Given features $\mathbf{x} = [x_1, x_2, \cdots, x_D]^T$
want to compute class probabilities using Bayes Rule:

$$p(c|\mathbf{x}) = \frac{p(\mathbf{x}|c) \cdot p(c)}{p(\mathbf{x})}$$

Pr. feature given class

Pr. class given feature

In words,

Posterior for class $= \frac{\text{Pr. of feature given class } \times \text{Prior for class}}{\text{Pr. of feature}}$

To compute $p(c|\mathbf{x})$ we need: $p(\mathbf{x}|c)$ and $p(c)$. 
Independence Assumption

- Naive assumption: The features $x_i$ are conditionally independent given the class $c$.
- Allows us to decompose the joint distribution:

$$p(c, x_1, \ldots, x_D) = p(c) p(x_1|c) \cdots p(x_D|c).$$

- Compact representation of the joint distribution.
  - Prior probability of class:
    $$p(c = 1) = \pi$$
  - Conditional probability of feature given class:
    $$p(x_j = 1|c) = \theta_{jc}$$
Naïve Bayes and SoftMax

• Assume $x \in \mathbb{R}^d$, learning a linear decision boundary is tantamount to learning $W \in \mathbb{R}^{C \times d}$.

$P(Class|features) = P(features|Class) \times P(Class)$

$\forall c \in C: f_c = W[c, \cdots ] \cdot x = \sum_{i=1}^{d} W[c, i] \cdot x[i]$ 

Uh oh – $f_c$ can be negative and we want something on $[0,1]$, to be a probability. Solution: Just raise it with an exponent

Softmax:

$$P(y|x) = \frac{\exp(f_y)}{\sum_{c=1}^{C} \exp(f_c)}$$
Naive Bayes Properties

- An amazingly cheap learning algorithm!
- **Training time**: Estimate parameters using maximum likelihood.
  - Compute co-occurrence counts of each feature with the labels. Requires only one pass through the data!
- **Test time**: Apply Bayes’ Rule.
  - Cheap because of the model structure. For more general models, Bayesian inference can be very expensive and/or complicated.
- Analysis easily extends to prob. distributions other than Bernoulli.
- Less accurate in practice compared to discriminative models due to its “naive” independence assumption.
Features and classification

- We talked about:
  - How preprocessing can affect feature extraction.
  - What parts-of-speech are, and how to identify them.
  - How to prepare data for classification
  - SVMs
  - Decision trees (which are parts of random forests)
Why do we use K-fold cross-validation and what should we watch out for?

Explain Underfitting and Overfitting with respect to the Bias-Variance Tradeoff.

What does SVM maximize and minimize simultaneously?

What are some tricks that SVM uses?
Readings

• J&M: 5.1-5.5 (2nd edition)
• M&S: 16.1, 16.4
# Appendix – prepositions from CELEX

<table>
<thead>
<tr>
<th>of</th>
<th>540,085</th>
<th>through</th>
<th>14,964</th>
<th>worth</th>
<th>1,563</th>
<th>pace</th>
<th>12</th>
</tr>
</thead>
<tbody>
<tr>
<td>in</td>
<td>331,235</td>
<td>after</td>
<td>13,670</td>
<td>toward</td>
<td>1,390</td>
<td>nigh</td>
<td>9</td>
</tr>
<tr>
<td>for</td>
<td>142,421</td>
<td>between</td>
<td>13,275</td>
<td>plus</td>
<td>750</td>
<td>re</td>
<td>4</td>
</tr>
<tr>
<td>to</td>
<td>125,691</td>
<td>under</td>
<td>9,525</td>
<td>till</td>
<td>686</td>
<td>mid</td>
<td>3</td>
</tr>
<tr>
<td>with</td>
<td>124,965</td>
<td>per</td>
<td>6,515</td>
<td>amongst</td>
<td>525</td>
<td>o’er</td>
<td>2</td>
</tr>
<tr>
<td>on</td>
<td>109,129</td>
<td>among</td>
<td>5,090</td>
<td>via</td>
<td>351</td>
<td>but</td>
<td>0</td>
</tr>
<tr>
<td>at</td>
<td>100,169</td>
<td>within</td>
<td>5,030</td>
<td>amid</td>
<td>222</td>
<td>ere</td>
<td>0</td>
</tr>
<tr>
<td>by</td>
<td>77,794</td>
<td>towards</td>
<td>4,700</td>
<td>underneath</td>
<td>164</td>
<td>less</td>
<td>0</td>
</tr>
<tr>
<td>from</td>
<td>74,843</td>
<td>above</td>
<td>3,056</td>
<td>versus</td>
<td>113</td>
<td>midst</td>
<td>0</td>
</tr>
<tr>
<td>about</td>
<td>38,428</td>
<td>near</td>
<td>2,026</td>
<td>amidst</td>
<td>67</td>
<td>o’</td>
<td>0</td>
</tr>
<tr>
<td>than</td>
<td>20,210</td>
<td>off</td>
<td>1,695</td>
<td>sans</td>
<td>20</td>
<td>thru</td>
<td>0</td>
</tr>
<tr>
<td>over</td>
<td>18,071</td>
<td>past</td>
<td>1,575</td>
<td>circa</td>
<td>14</td>
<td>vice</td>
<td>0</td>
</tr>
</tbody>
</table>
## Appendix – particles

<table>
<thead>
<tr>
<th>aboard</th>
<th>aside</th>
<th>besides</th>
<th>forward(s)</th>
<th>opposite</th>
<th>through</th>
</tr>
</thead>
<tbody>
<tr>
<td>about</td>
<td>astray</td>
<td>between</td>
<td>home</td>
<td>out</td>
<td>throughout</td>
</tr>
<tr>
<td>above</td>
<td>away</td>
<td>beyond</td>
<td>in</td>
<td>outside</td>
<td>together</td>
</tr>
<tr>
<td>across</td>
<td>back</td>
<td>by</td>
<td>inside</td>
<td>over</td>
<td>under</td>
</tr>
<tr>
<td>ahead</td>
<td>before</td>
<td>close</td>
<td>instead</td>
<td>overhead</td>
<td>underneath</td>
</tr>
<tr>
<td>alongside</td>
<td>behind</td>
<td>down</td>
<td>near</td>
<td>past</td>
<td>up</td>
</tr>
<tr>
<td>apart</td>
<td>below</td>
<td>east, etc.</td>
<td>off</td>
<td>round</td>
<td>within</td>
</tr>
<tr>
<td>around</td>
<td>beneath</td>
<td>eastward(s), etc.</td>
<td>on</td>
<td>since</td>
<td>without</td>
</tr>
</tbody>
</table>
## Appendix – conjunctions

<table>
<thead>
<tr>
<th>Word</th>
<th>Frequency</th>
<th>Word</th>
<th>Frequency</th>
<th>Word</th>
<th>Frequency</th>
<th>Word</th>
<th>Frequency</th>
<th>Word</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>and</td>
<td>514,946</td>
<td>yet</td>
<td>5,040</td>
<td>considering</td>
<td>174</td>
<td>forasmuch as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>that</td>
<td>134,773</td>
<td>since</td>
<td>4,843</td>
<td>lest</td>
<td>131</td>
<td>however</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>but</td>
<td>96,889</td>
<td>where</td>
<td>3,952</td>
<td>albeit</td>
<td>104</td>
<td>immediately</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>or</td>
<td>76,563</td>
<td>nor</td>
<td>3,078</td>
<td>providing</td>
<td>96</td>
<td>in as far as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>as</td>
<td>54,608</td>
<td>once</td>
<td>2,826</td>
<td>whereupon</td>
<td>85</td>
<td>in so far as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>if</td>
<td>53,917</td>
<td>unless</td>
<td>2,205</td>
<td>seeing</td>
<td>63</td>
<td>inasmuch as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>when</td>
<td>37,975</td>
<td>why</td>
<td>1,333</td>
<td>directly</td>
<td>26</td>
<td>insomuch as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>because</td>
<td>23,626</td>
<td>now</td>
<td>1,290</td>
<td>ere</td>
<td>12</td>
<td>insomuch that</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>so</td>
<td>12,933</td>
<td>neither</td>
<td>1,120</td>
<td>notwithstanding</td>
<td>3</td>
<td>like</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>before</td>
<td>10,720</td>
<td>whenever</td>
<td>913</td>
<td>according as</td>
<td>0</td>
<td>neither nor</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>though</td>
<td>10,329</td>
<td>whereas</td>
<td>867</td>
<td>as if</td>
<td>0</td>
<td>now that</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>than</td>
<td>9,511</td>
<td>except</td>
<td>864</td>
<td>as long as</td>
<td>0</td>
<td>only</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>while</td>
<td>8,144</td>
<td>till</td>
<td>686</td>
<td>as though</td>
<td>0</td>
<td>provided that</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>after</td>
<td>7,042</td>
<td>provided</td>
<td>594</td>
<td>both and</td>
<td>0</td>
<td>providing that</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>whether</td>
<td>5,978</td>
<td>whilst</td>
<td>351</td>
<td>but that</td>
<td>0</td>
<td>seeing as</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>for</td>
<td>5,935</td>
<td>suppose</td>
<td>281</td>
<td>but then</td>
<td>0</td>
<td>seeing as how</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>although</td>
<td>5,424</td>
<td>cos</td>
<td>188</td>
<td>but then again</td>
<td>0</td>
<td>seeing that</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>until</td>
<td>5,072</td>
<td>supposing</td>
<td>185</td>
<td>either or</td>
<td>0</td>
<td>without</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Appendix – Penn TreeBank PoS tags

<table>
<thead>
<tr>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
<th>Tag</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>CC</td>
<td>coordin. conjunction</td>
<td><em>and, but, or</em></td>
<td>SYM</td>
<td>symbol</td>
<td>+, %, &amp;</td>
</tr>
<tr>
<td>CD</td>
<td>cardinal number</td>
<td><em>one, two, three</em></td>
<td>TO</td>
<td>“to”</td>
<td>to</td>
</tr>
<tr>
<td>DT</td>
<td>determiner</td>
<td><em>a, the</em></td>
<td>UH</td>
<td>interjection</td>
<td><em>ah, oops</em></td>
</tr>
<tr>
<td>EX</td>
<td>existential ‘there’</td>
<td><em>there</em></td>
<td>VB</td>
<td>verb, base form</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>FW</td>
<td>foreign word</td>
<td><em>mea culpa</em></td>
<td>VBD</td>
<td>verb, past tense</td>
<td><em>ate</em></td>
</tr>
<tr>
<td>IN</td>
<td>preposition/sub-conj</td>
<td><em>of, in, by</em></td>
<td>VBG</td>
<td>verb, gerund</td>
<td><em>eating</em></td>
</tr>
<tr>
<td>JJ</td>
<td>adjective</td>
<td><em>yellow</em></td>
<td>VBN</td>
<td>verb, past participle</td>
<td><em>eaten</em></td>
</tr>
<tr>
<td>JJR</td>
<td>adj., comparative</td>
<td><em>bigger</em></td>
<td>VBP</td>
<td>verb, non-3sg pres</td>
<td><em>eat</em></td>
</tr>
<tr>
<td>JJS</td>
<td>adj., superlative</td>
<td><em>wildest</em></td>
<td>VBZ</td>
<td>verb, 3sg pres</td>
<td><em>eats</em></td>
</tr>
<tr>
<td>LS</td>
<td>list item marker</td>
<td><em>1, 2, One</em></td>
<td>WDT</td>
<td>wh-determiner</td>
<td><em>which, that</em></td>
</tr>
<tr>
<td>MD</td>
<td>modal</td>
<td><em>can, should</em></td>
<td>WP</td>
<td>wh-pronoun</td>
<td><em>what, who</em></td>
</tr>
<tr>
<td>NN</td>
<td>noun, sing. or mass</td>
<td><em>llama</em></td>
<td>WP$</td>
<td>possessive wh-</td>
<td><em>whose</em></td>
</tr>
<tr>
<td>NNS</td>
<td>noun, plural</td>
<td><em>llamas</em></td>
<td>WRB</td>
<td>wh-adverb</td>
<td><em>how, where</em></td>
</tr>
<tr>
<td>NNP</td>
<td>proper noun, singular</td>
<td><em>IBM</em></td>
<td>$</td>
<td>dollar sign</td>
<td>$</td>
</tr>
<tr>
<td>NNPS</td>
<td>proper noun, plural</td>
<td><em>Carolinas</em></td>
<td>#</td>
<td>pound sign</td>
<td>#</td>
</tr>
<tr>
<td>PDT</td>
<td>predeterminer</td>
<td><em>all, both</em></td>
<td>“</td>
<td>left quote</td>
<td>‘ or “</td>
</tr>
<tr>
<td>POS</td>
<td>possessive ending</td>
<td>’s</td>
<td>”</td>
<td>right quote</td>
<td>‘ or ”</td>
</tr>
<tr>
<td>PRP</td>
<td>personal pronoun</td>
<td><em>I, you, he</em></td>
<td>(</td>
<td>left parenthesis</td>
<td>[, (, {, &lt;</td>
</tr>
<tr>
<td>PRPS</td>
<td>possessive pronoun</td>
<td><em>your, one’s</em></td>
<td>)</td>
<td>right parenthesis</td>
<td>], ), }, &gt;</td>
</tr>
<tr>
<td>RB</td>
<td>adverb</td>
<td><em>quickly, never</em></td>
<td>,</td>
<td>comma</td>
<td>,</td>
</tr>
<tr>
<td>RBR</td>
<td>adverb, comparative</td>
<td><em>faster</em></td>
<td>.</td>
<td>sentence-final punc</td>
<td>. ! ?</td>
</tr>
<tr>
<td>RBS</td>
<td>adverb, superlative</td>
<td><em>fastest</em></td>
<td>:</td>
<td>mid-sentence punc</td>
<td>: ; … − − − −</td>
</tr>
<tr>
<td>RP</td>
<td>particle</td>
<td><em>up, off</em></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example – Hero classification

<table>
<thead>
<tr>
<th>Hero</th>
<th>Hair length</th>
<th>Height</th>
<th>Age</th>
<th>Hero Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aquaman</td>
<td>2”</td>
<td>6’2”</td>
<td>35</td>
<td>Hero</td>
</tr>
<tr>
<td>Batman</td>
<td>1”</td>
<td>5’11”</td>
<td>32</td>
<td>Hero</td>
</tr>
<tr>
<td>Catwoman</td>
<td>7”</td>
<td>5’9”</td>
<td>29</td>
<td>Villain</td>
</tr>
<tr>
<td>Deathstroke</td>
<td>0”</td>
<td>6’4”</td>
<td>28</td>
<td>Villain</td>
</tr>
<tr>
<td>Harley Quinn</td>
<td>5”</td>
<td>5’0”</td>
<td>27</td>
<td>Villain</td>
</tr>
<tr>
<td>Martian Manhunter</td>
<td>0”</td>
<td>8’2”</td>
<td>128</td>
<td>Hero</td>
</tr>
<tr>
<td>Poison Ivy</td>
<td>6”</td>
<td>5’2”</td>
<td>24</td>
<td>Villain</td>
</tr>
<tr>
<td>Wonder Woman</td>
<td>6”</td>
<td>6’1”</td>
<td>108</td>
<td>Hero</td>
</tr>
<tr>
<td>Zatanna</td>
<td>10”</td>
<td>5’8”</td>
<td>26</td>
<td>Hero</td>
</tr>
</tbody>
</table>

Test data

| Red Hood | 2” | 6’0” | 22 | ? |

Characters © DC
Example – Hero classification

• How do we split?
  • Split on *hair length*?
  • Split on *height*?
  • Split on *age*?

• Let’s compute the **information gain** for each:

\[
Gain(Q) = H(S) - \sum_{child \text{ set}} p(\text{child set})H(\text{child set})
\]
Split on hair length?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]
Split on hair length?

Gain(Question) = \( H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \)

\[
H(S) = \frac{h}{h+v} \log_2 \left( \frac{h+v}{h} \right) + \frac{v}{h+v} \log_2 \left( \frac{h+v}{v} \right)
\]

\[
H(5h, 4v) = \frac{5}{9} \log_2 \left( \frac{9}{5} \right) + \frac{4}{9} \log_2 \left( \frac{9}{4} \right) = 0.9911 \text{ bits}
\]
Split on hair length?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set}) H(\text{child set}) \]
Split on hair length?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]

**Hair Length ≤ 5”?**

- **YES**
  
  \[ H(4h,1v) = \frac{4}{5} \log_2 \left( \frac{5}{4} \right) + \frac{1}{5} \log_2 \left( \frac{5}{1} \right) = 0.7219 \]

- **NO**
Split on hair length?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set}) H(\text{child set}) \]

**Hair Length ≤ 5”?**

**YES**

\[ H(4h, 1v) = \frac{4}{5} \log_2 \left( \frac{5}{4} \right) + \frac{1}{5} \log_2 \left( \frac{5}{1} \right) = 0.7219 \]

**NO**

\[ H(2h, 2v) = \frac{2}{4} \log_2 \left( \frac{4}{2} \right) + \frac{2}{4} \log_2 \left( \frac{4}{2} \right) = 1 \]
Split on hair length?

Gain(\text{HairLength} \leq 5") = 0.9911 - \frac{5}{9} \cdot 0.7219 - \frac{4}{9} \cdot 1 = 0.00721

Gain(\text{Question}) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set})
Example – Hero classification

• How do we split?
  • Split on *hair length*?  \( \text{Gain}(\text{HairLength} \leq 5") = 0.00721 \)
  • Split on *height*?
  • Split on *age*?

• Let’s compute the **information gain** for each:

\[
\text{Gain}(Q) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set})
\]
Split on height?

\[ \text{Gain(} \text{Question}) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]
Split on height?

$\text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set})$

$H(S) = \frac{h}{h+v} \log_2 \left( \frac{h+v}{h} \right) + \frac{v}{h+v} \log_2 \left( \frac{h+v}{v} \right)$

$H(5h, 4v) = \frac{5}{9} \log_2 \left( \frac{9}{5} \right) + \frac{4}{9} \log_2 \left( \frac{9}{4} \right) = 0.9911 \text{ bits}$

Height ≤ 6’0”?

YES

NO
Split on height?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]

1. **Height \leq 6'0"**
   - \[ H(2h, 3v) = \frac{2}{5} \log_2 \left( \frac{5}{2} \right) + \frac{3}{5} \log_2 \left( \frac{5}{3} \right) = 0.971 \]
   - **YES**

2. **Height > 6'0"**
   - \[ H(3h, 1v) = \frac{3}{4} \log_2 \left( \frac{4}{3} \right) + \frac{1}{4} \log_2 \left( \frac{4}{1} \right) = 0.813 \]
   - **NO**

CSC 401/2511 – Spring 2024
Split on height?

\[ \text{Gain}(\text{Question}) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]

\[ \text{Gain}(\text{Height} \leq 6'0") = 0.9911 - \frac{5}{9}[0.971] - \frac{4}{9}[0.813] = 0.0903 \]
Example – Hero classification

• How do we split?
  • Split on hair length? \( \text{Gain}(\text{HairLength} \leq 5") = 0.00721 \)
  • Split on height?
  • Split on age?

• Let’s compute the information gain for each:

\[
\text{Gain}(Q) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set})
\]
Split on age?

\[ \text{Gain(Question)} = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \]

\[
H(S) = \frac{h}{h+v} \log_2 \left( \frac{h+v}{h} \right) + \frac{v}{h+v} \log_2 \left( \frac{h+v}{v} \right)
\]

\[
H(5h, 4v) = \frac{5}{9} \log_2 \left( \frac{9}{5} \right) + \frac{4}{9} \log_2 \left( \frac{9}{4} \right) = 0.9911 \text{ bits}
\]
Split on age?

Gain(Question) = \( H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set}) \)

\[
H(1h, 4v) = \frac{1}{5} \log_2 \left( \frac{5}{1} \right) + \frac{4}{5} \log_2 \left( \frac{5}{4} \right) = 0.722
\]

\[
H(4h, 0v) = \frac{4}{4} \log_2 \left( \frac{4}{4} \right) + 0 \log_2 (\infty) = 0
\]
Split on age?

\[ \text{Gain(Question)} = H(S) - \sum_\text{child set} p(\text{child set})H(\text{child set}) \]

\[ \text{Gain(Age} \leq 30) = 0.9911 - \frac{5}{9}[0.722] - \frac{4}{9}[0] = 0.590 \]
Example – Hero classification

• How do we split?
  • Split on *hair length*? \( \text{Gain}(\text{HairLength} \leq 5'') = 0.00721 \)
  • Split on *height*? \( \text{Gain}(\text{Height} \leq 6'0'') = 0.0903 \)
  • Split on *age*? \( \text{Gain}(\text{Age} \leq 30) = 0.590 \)

• Let’s compute the information gain for each:

\[
\text{Gain}(Q) = H(S) - \sum_{\text{child set}} p(\text{child set})H(\text{child set})
\]
The resulting tree

- Splitting on age resulted in the greatest information gain.

- We’re left with one heterogeneous set, so we recurse and find that hair length results in a complete classification of the training data.
We just need to keep track of the attribute questions – not the training data.

How are the following characters classified?

<table>
<thead>
<tr>
<th>Person</th>
<th>Hair length</th>
<th>Height</th>
<th>Age</th>
</tr>
</thead>
<tbody>
<tr>
<td>Red Hood</td>
<td>2”</td>
<td>6’0”</td>
<td>22</td>
</tr>
<tr>
<td>Green Arrow</td>
<td>1”</td>
<td>6’2”</td>
<td>38</td>
</tr>
<tr>
<td>Bane</td>
<td>0”</td>
<td>5’8”</td>
<td>29</td>
</tr>
</tbody>
</table>

- Inspired from Allan Neymark’s (San Jose State University) Simpsons example.