Natural Language Processing

Lecture 8: Classification
Three Spelling Problems

✓ Detecting isolated non-words
✓ Fixing isolated non-words

3. Fixing errors in context
Kernighan's Model: A Noisy Channel
## across

| c         | freq(c) | \( p(t | c) \)         | %  |
|-----------|---------|------------------------|----|
| actress   | 1343    | \( p(\text{delete } t) \) | 37 |
| cress     | 0       | \( p(\text{delete } a) \) | 0  |
| caress    | 4       | \( p(\text{transpose } a \& \ c) \) | 0  |
| access    | 2280    | \( p(\text{substitue } r \text{ for } c) \) | 0  |
| across    | 8436    | \( p(\text{substitute } e \text{ for } o) \) | 18 |
| acres     | 2879    | \( p(\text{delete } s) \) | 21 |
| acres     | 2879    | \( p(\text{delete } s) \) | 23 |
Noisy Channel Model (General)
Classification
Notation

- Training examples: \( \mathbf{x} = (x_1, x_2, \ldots, x_N) \)
- Their categories: \( \mathbf{y} = (y_1, y_2, \ldots, y_N) \)
- A classifier \( C \) seeks to map \( x_i \) to \( y_i \)

\[
\mathbf{x} \rightarrow \boxed{C} \rightarrow \mathbf{y}
\]

- A learner \( L \) infers \( C \) from \( (\mathbf{x}, \mathbf{y}) \)

\[
\mathbf{x} \rightarrow \boxed{L} \rightarrow \boxed{C}
\]

\[
\mathbf{y} \rightarrow \boxed{L} \rightarrow \boxed{C}
\]
Probabilistic Classifiers

\[ X \rightarrow \text{return } \arg \max_y, \ p(y' \mid x) \rightarrow y \]
Noisy Channel Model (General)

$p(y)$

source $\rightarrow y \rightarrow X$

channel $p(x | y)$

deencode

What proportion of emails are expected to be spam vs. not spam?

What proportion of product reviews are expected to get 1,2,3,4,5 stars?
Noisy Channel Classifiers

return \[ \arg\max_y p(y) \times p(x \mid y) \]
Representing Text: Features

• Any object \( x \in \mathcal{X} \) you might be given to classify can be represented as a vector in a vector space
  – Vectors of representing text are often sparse and high-dimensional

• Designing \( \Phi \) ("Feature engineering")
  – What information do you need to solve the problem?
  – What information do you need to avoid mistakes?
  – Very common: bag-of-words
Naïve Bayes Classifier

\[
\phi_j \leftarrow [\Phi(x)]_j
\]

return

\[
\arg \max_{y'} p(y') \times \prod_j p(\phi_j \mid y')
\]
Naïve Bayes Learner

∀y, \( p(y) \leftarrow \frac{\text{count}(y)}{N} \)

∀y, \( \forall j, \forall f, p(\phi_j(x) = f \mid y) \leftarrow \frac{\text{count}(f, y)}{\text{count}(y)} \)
Linear Classifiers

C:
1. Use $\Phi(x)$ to map $x$ onto a real-valued feature space.
2. Calculate the linear score $z = w^T \Phi(x)$.
3. If $z > 0$, then return $y = \text{YES}$, else $y = \text{NO}$.

$X \rightarrow C \rightarrow Y$
Linear Classifiers
Linear Classifiers

\[ x \cdot w^T u = 0 \]
Linear Classifiers

\[ x \cdot w = 0 \]
Linear Classifiers

\[ \mathbf{x}^\top \mathbf{w} = 0 \]
Linear Classifiers (> 2 Classes)

\[
\text{return } \arg \max_y \mathbf{w}^T \Phi(x, y)
\]
Perceptron Learner

\[ w \leftarrow 0 \]

for \( t = 1 \ldots T \):

select \((x_t, y_t)\)

\# run current classifier

\( y \leftarrow \arg \max_y \ w^\top \Phi(x, y') \)

if \( y \neq y_t \) then \# mistake

\[ w \leftarrow w + \alpha [\Phi(x_t, y_t) - \Phi(x_t, y)] \]

return \( w \)
Classifiers

• Noisy Channel: Two components:
  - Most probable times prior probability
• Linear Classifiers: feature times weight
• Naive Bayes: Independent feature/weights
• Perceptron: back-propagate loss to input weights
• Choose your input features well
• Use the one that works best