NLP Demystified

Noah Smith
Language Technologies Institute
Machine Learning Department
School of Computer Science
Carnegie Mellon University
nasmith@cs.cmu.edu
NLP?
Outline

1. Automatically categorizing documents
2. Decoding sequences of words
3. Clustering documents and/or words
Categorizing Documents: Examples

- Mosteller and Wallace (1964): authorship of the *Federalist* papers
- News categories: U.S., world, sports, religion, business, technology, entertainment, ...
- How positive or negative is a review of a film or restaurant?
- Is a given email message spam?
- What is the reading level of a piece of text?
- How influential will a research paper be?
- Will a congressional bill pass committee?
The Vision

- Human experts label some data
- Feed the data to a learning algorithm that constructs an automatic labeling function
- Apply that function to as much data as you want!
Basic Recipe for Document Categorization

1. Obtain a pool of correctly categorized documents $D$.
2. Define a function $f$ from documents to feature vectors.
3. Define a parameterized function $h_w$ from feature vectors to categories.
4. Select $h$’s parameters $w$ using a training sample from $D$.
5. Estimate performance on a held-out sample from $D$. 
1. Obtain Categorized Documents

Spinoza, 17th century rationalist
2. Define the Feature Vector Function

• Simplest choice: one dimension per word, and let \([f(d)]_j\) be the count of \(w_j\) in \(d\).

• Twists:
  – Monotonic transforms, like dividing by the length of \(d\) or taking a log.
  – Increase the weights of words that occur in fewer documents ("inverse document frequency")
  – \(n\)-grams
  – Count specially defined groupings of words
  – Statistical tests to select words likely to be informative
Basic Recipe for Document Categorization

1. Obtain a pool of correctly categorized documents $D$.
2. Define a function $f$ from documents to feature vectors.
3. Define a parameterized function $h$, from feature vectors to categories.
4. Select $h$’s parameters using a training sample from $D$.
5. Estimate performance on a held-out sample from $D$. 
3. Define a Function from Feature Vectors to Categories

- Simplest choice: linear model

$$h_w(d) = \arg \max_c w_c \mathbf{f}(d) + w_c^{bias}$$

$w_c$ is the vector of coefficients associating each feature with class $c$ (can be positive or negative).

- Advantage: interpretability
- Advantage: computational efficiency

- Some alternatives: k-nearest neighbors, decision trees, neural networks, ...
4. Select Parameters using Data

- Also known as “machine learning.”
- Many learning options for linear classifiers!

- **NB** (probabilistic interpretation)
- **LR**
- **SVM** (discriminative)
- **perceptron**
4. Select Parameters using Data

Optimization view of learning:

\[ \hat{w} = \arg \min_w R(w) + \frac{1}{|D_{\text{train}}|} \sum_{d \in D_{\text{train}}} L(d; w) \]

“regularization” to avoid overfitting

“empirical risk” = average loss over training data

Typical loss functions for linear models are convex and can be efficiently optimized using online or batch iterative algorithms with convergence guarantees.
4. Select Parameters using Data

Considerations:
• Do you want posterior probabilities, or just labels?
• What methods do you understand well enough to explain in your paper?
• What methods will your readers understand?
• What implementations are available?
  – Cost, scalability, programming language, compatibility with your workflow, ...
• How well does it work (on held-out data)?
5. Estimate Performance

- Always, always, always use **held-out** data.
  - Multiple rounds of tests? Fresh testing data!
- Consider the “most frequent class” baseline.
- Consider inter-annotator agreement.
- What to measure?
  - Accuracy
  - When one class is special: precision/recall
5. Estimate Performance

\[ h_w(d) = \arg \max_c w_c^\top f(d) + w_c^{bias} \]
Basic Recipe for Document Categorization

1. Obtain a pool of correctly categorized documents $D$.
2. Define a function $f$ from documents to feature vectors.
3. Define a parameterized function $h_w$ from feature vectors to categories.
4. Select $h$’s parameters $w$ using a training sample from $D$.
5. Estimate performance on a held-out sample from $D$. 
Outline

- Automatically categorizing documents
2. Decoding sequences of words
3. Clustering documents and/or words
Decoding Word Sequences: Examples

- Categorizing each word by its part-of-speech or semantic class
- Recognizing mentions of named entities
- Segmenting a document into parts
- Parsing a sentence into a grammatical or semantic structure
High-Level View

classification

structured prediction
Possible Lines of Attack

1. Transform into a sequence of classification problems (see part 1).

2. Transform into a sequence labeling problem and use a variant of the Viterbi algorithm.

3. Design a representation, prediction algorithm, and learning algorithm for your particular problem.
Shameless Self-Promotion

$56.43 on amazon.com

possibly free in electronic form, through your university’s library
Lines of Attack

1. Reduce to a sequence of classification problems (see part 1).

2. Reduce to a sequence labeling problem and use a variant of the Viterbi algorithm.

3. Design a representation, prediction algorithm, and learning algorithm for your problem.
Sequence Labeling

• Input: sequence of symbols $x_1 x_2 \ldots x_L$
• Output: sequence of labels $y_1 y_2 \ldots y_L$ each $\in \Lambda$

Prediction rule:

$$h_w(x) = \arg \max_y w^\top f(x_1 \ldots x_L, y_1 \ldots y_L)$$

Problem: there are $O(|\Lambda|^L)$ choices for $y_1 y_2 \ldots y_L$!
Sequence Labeling with Local Features

A key assumption about $f$ allows us to solve the problem exactly, in $O(|\Lambda|^2 L)$ time and $O(|\Lambda|L)$ space.

$$h_w(x) = \arg \max_y w^\top f(x_1 \ldots x_L, y_1 \ldots y_L)$$

$$= \arg \max_y w^\top \left( \sum_{\ell=1}^{L-1} f_{local}(x_1 \ldots x_L, y_\ell y_{\ell+1}) \right)$$
If I knew the best label sequence for $x_1 \ldots x_{L-1}$, then $y_L$ would be easy.

That decision would depend only on state $L-1$.

$$y^*_L = \arg \max_{y_L \in \Lambda} \mathbf{w}^\top \left( \sum_{\ell=1}^{L-2} \mathbf{f}_{\text{local}}(x_1 \ldots x_L, y^*_\ell y^*_{\ell+1}) \right) + \mathbf{w}^\top \mathbf{f}_{\text{local}}(x_1 \ldots x_L, y^*_{L-1} y_L)$$

$$= \mathbf{w}^\top \left( \sum_{\ell=1}^{L-2} \mathbf{f}_{\text{local}}(x_1 \ldots x_L, y^*_\ell y^*_{\ell+1}) \right) + \arg \max_{y_L \in \Lambda} \mathbf{w}^\top \mathbf{f}_{\text{local}}(x_1 \ldots x_L, y^*_{L-1} y_L)$$

I don’t know that best sequence, but there are only $|\Lambda|$ options at $L-1$.

So I only need the score of the best sequence up to $L-1$, for each possible label at $L-1$. Call this $V[L-1, y]$ for $y \in \Lambda$. From this, I can score each label at $L$, for each hypothetical label at $L-1$.

Score of the best sequences up to $L-1$ relies similarly on score of the best sequences up to $L-2$. Ditto, at every other timestep $L-2, L-3, \ldots 1$. 

(Featurized) Viterbi Algorithm

• Precompute $V[* , * ]$ from left to right. $V[1, *] = 0$. For $\ell = 2$ to $L$, for each $y$ in $\Lambda$:

$$V[\ell, y] = \max_{y' \in \Lambda} V[\ell - 1, y'] + w^\top f_{local}(x_1 \ldots x_L, y'y)$$

$$B[\ell, y] = \arg \max_{y' \in \Lambda} V[\ell - 1, y'] + w^\top f_{local}(x_1 \ldots x_L, y'y)$$

• Backtrack and select the labels from right to left.

$$y_L^* = \arg \max_y V[L, y]$$

For $\ell = L - 1$ to $1$:

$$y_\ell^* = B[\ell + 1, y_{\ell+1}^*]$$
Part of Speech Tagging

After paying the medical bills, Frances was nearly broke.

• Adverb (RB)
• Verb (VBG, VBZ, and others)
• Determiner (DT)
• Adjective (JJ)
• Noun (NN, NNS, NNP, and others)
• Punctuation (., ,, and others)
Named Entity Recognition

With Commander Chris Ferguson at the helm,

Atlantis touched down at Kennedy Space Center.
Named Entity Recognition

With Commander Chris Ferguson at the helm, Atlantis touched down at Kennedy Space Center.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.

Noahs Arche war nicht voller Productionsfactoren, sondern Geschöpfe.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.

Noahs Arche war nicht voller Productionsfactoren, sondern Geschöpfe.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Mr. President, Noah’s ark was filled not with production factors, but with living creatures.
Basic Recipe for Sequence Labeling

1. Obtain a pool of correctly labeled sequences $D$.
2. Define a locally factored function $f$ from sequences and labelings to feature vectors.
3. Define a parameterized function $h_w$ from feature vectors to labelings.
4. Select $h$’s parameters $w$ using a training sample from $D$.
5. Estimate performance on a held-out sample from $D$. 
Structured Learners Generalize Linear Classification Learners!

- hidden Markov models ← naïve Bayes
- conditional random fields ← logistic regression
- structured perceptron ← perceptron
- structured SVM ← support vector machine
Additional Notes

• Outputs that are trees, graphs, logical forms, other strings ...
  
  parse trees (phrase structure, dependencies)
  coreference relationships among entity mentions (and pronouns)
  a huge range of semantic analyses

• Evaluation?
Dependency Parse

Robots in popular culture are there to remind us of the awesomeness of unbounded human agency.
### Frame-Semantic Parse

| Robots in popular culture are there to remind us of the awesomeness of unbounded human agency. |

<table>
<thead>
<tr>
<th>Entity</th>
<th>Stimulus</th>
<th>Evaluate</th>
<th>Cognizer</th>
<th>Descriptor</th>
<th>Organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Existence</td>
<td>Desirability</td>
<td>Evoking</td>
<td>People</td>
<td>People</td>
<td></td>
</tr>
</tbody>
</table>
Run our Parsers!

http://demo.ark.cs.cmu.edu/parse
Outline

✓ Automatically categorizing documents
✓ Decoding sequences of words
3. Clustering documents and/or words
Clustering Real Data
K-Means

Given: points \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}, K (number of clusters)

1. Arbitrarily select \(\mathbf{\mu}_1, \ldots, \mathbf{\mu}_K\).
2. Assign each \(\mathbf{x}_i\) to the nearest \(\mathbf{\mu}_j\).
3. Select each \(\mathbf{\mu}_j\) to be the mean of all \(\mathbf{x}_i\) assigned to it.
4. If all \(\mathbf{\mu}_j\) have converged stop; else go to 2.
K-Means, Visualized
K-Means, Visualized
K-Means, Visualized
K-Means, Visualized
K-Means, Visualized
K-Means, Visualized
K-Means for Text?

• Documents
  – Use the same $f$ we might use for classification.

• Words
  – Use “context” vectors ...
Where's the beef?

1. fertility. Organ meats such as beef and chicken liver, tongue and head controlling scours. _HOW TO FEED: BEEF AND DAIRY CALVES_ — 0.2 gram Dy ing process discolors the treated beef and liquid accumulates in prepackag say. He did say she could get her beef and vegetables in cans this summer and feed efficiency of fattening beef animals. _HOW TO FEED:_ At the steaks, chops, chicken and prime beef as well as Tom's favorite dish, stub ross from him was surmounted by a beef barrel with ends knocked out. In t counter of boards laid across two beef barrels. There was, of course, no Because Holstein cattle weren't a beef breed, they were rarely seen on a 2-5 grams of phenothiazine daily; beef calves—.5 to 1.5 grams daily de lities of this drug. _HOW TO FEED: BEEF CATTLE (FINISHING RATION)_ — To dairy cows and lesser amounts to beef cattle and poultry. About 90 percen raises enough poultry, pigs, and beef cattle for most of their needs. Lo on of liver abscesses in feed-lot beef cattle. Prevention of bacterial pne pal feed bunk types for dairy and beef cattle: (1) Fence-line bunks—catt es feed efficiency. _HOW TO FEED: BEEF CATTLE_ — 10 milligrams of diet the rations you are feeding your beef, dairy cattle, and sheep are adequa tive business more profitable for beef, dairy, and sheep men. The tar o bear. She was ready to kill the beef, dress it out, and with vegetables . She had raised a calf, grown it beef-fat. She had, with her own work-wea with feeding low-moisture corn in beef-feeding programs. Several firms ar he shelf life (at 35 F) of fresh beef from 5 days to 5 or 6 weeks. Howeve canned pork products. Tests with beef have been largely unsuccessful beca for eggs, pigs to eat garbage, a beef herd and wastes of all kinds. Separ their money's worth. A good many beef-hungry settlers were accepting the
chicken

y the irradiated and refrigerated chicken. Acceptance of radiopasteurization
torehouse"

chicken. Glendora dropped a chicken and a flurry of feathers, and went
will specialize in steaks, chops, chicken and prime beef as well as Tom's fa
ard as the one concerned with the chicken and the egg. Which came first? Is
he millions of buffalo and prairie chicken "$!
"Come on, there's some cold chicken and we'll see what else". They wen
ves to extend the storage life of chicken

CHICKEN CADILLAC# Use one 6-ounce chicken
ion juice, to about half cover the chicken breasts. Bake slowly at least one-
d, in butter. Sprinkle over top of chicken around, they had a hard time". #CHICKEN
successful, and the shelf life of chicken can be extended to a month or more
ay from making a cake, building a chicken coop, or producing a book, to found
, they decided, but a deck full of chicken
ty Johnson reaching around the wire chicken eye glittering behind dull silver chicken
wine in the pot roast or that the chicken had been marinated in brandy, and
ved this same game and called it "Chicken". He could not go through the f
f the Mexicans hiding in a little chicken house had passed through his head,
I'll never forget him cleaning the chicken in the tub". A story, no doubt
. Organ meats such as beef and chicken liver, tongue and heart are planne
p. "Miss Sarah, I can't cut up no chicken. Miss Maude say she won't". Aga
pot. "What is it"? he asked. "Chicken", Mose said, and theatrically licke
im"? Adam shook his head. "Chicken", Mose said. She was a child too m
Hypothetical Counts based on Syntactic Dependencies

<table>
<thead>
<tr>
<th></th>
<th>Modified-by-ferocious(adj)</th>
<th>Subject-of-devour(v)</th>
<th>Object-of-pet(v)</th>
<th>Modified-by-African(adj)</th>
<th>Modified-by-big(adj)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lion</td>
<td>15</td>
<td>5</td>
<td>0</td>
<td>6</td>
<td>15</td>
</tr>
<tr>
<td>Dog</td>
<td>7</td>
<td>3</td>
<td>8</td>
<td>0</td>
<td>12</td>
</tr>
<tr>
<td>Cat</td>
<td>1</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>Elephant</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>...</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Brown Clustering

Given: corpus of length N, K

1. Assign each word to its cluster (V clusters)
2. Repeat V – K times:
   • Find the single merge \((c_j, c_k)\) that results in a new clustering with the highest Quality score
   • Prepend \(c_j\)'s bitstring with 0 and \(c_k\)'s with 1 (and the same for all their descendents)
Mini-Example

Bitstrings that share a prefix are in the same cluster, at some level of granularity.
Clusters from Brown et al. (1992)
Clusters from Owoputi et al. (2013)  
(56M Tweets)

<table>
<thead>
<tr>
<th>acronyms for laughter</th>
<th>Imao Imfao Imaoo Imaoooo hahahahaha lool ctfu rofl loool Imfao Imfaooo Imao000 Imbo lololol</th>
</tr>
</thead>
<tbody>
<tr>
<td>onomatopoeic laughers</td>
<td>haha hahaha hehe hahahaha hahah aha hehehe ahaha hah hahahah kk hahaa aahh</td>
</tr>
<tr>
<td>affirmative</td>
<td>yes yep yup nope yess yesss yessss ofcourse yeap likewise yepp yesh yw yuup yus</td>
</tr>
<tr>
<td>negative</td>
<td>yeah yea nah naw yeahh nooo yeh noo noooo yeea ikr nvm yeahhh nahh nooooo</td>
</tr>
<tr>
<td>metacomment</td>
<td>smh jk #fail #random #fact smfh #smh #winning #realtalk smdh #dead #justsaying</td>
</tr>
</tbody>
</table>
## Clusters from Owoputi et al. (2013) (56M Tweets)

<table>
<thead>
<tr>
<th>Second Person Pronoun</th>
<th>u yu yuh yhu uu yuu yew y0u yuhh youh yhuu iget yoy yooh yuo yue juu dya youz yyou</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prepositions</td>
<td>w fo fa fr fro ov fer fir whit abou aft serie fore fah fuh w/her w/that fron isn agains</td>
</tr>
<tr>
<td>“Contractions”</td>
<td>tryna gon finna bouta trynna boutta gne fina gonn tryina fenna qone trynaa qon</td>
</tr>
<tr>
<td>Going To</td>
<td>gonna gunna gona gna guna gnna ganna qonna gonnna gana qunna gonne goona</td>
</tr>
<tr>
<td>So+</td>
<td>sooo sooo soooo soooooo soooooooo sooooooooo sooooooooooo soooooooooooooo</td>
</tr>
</tbody>
</table>
## Clusters from Owoputí et al. (2013) (56M Tweets)

<table>
<thead>
<tr>
<th>Sentiword</th>
<th>Examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>michevious</td>
<td>;) :p :-) xd ;-) ;d (; :3 ;p =p :-p =)) ;] xdd #gno xddd &gt;:) ;-p &gt;:d 8-) ;-d</td>
</tr>
<tr>
<td>happy</td>
<td>:) (; =) ;) )] :') =] ^_^ :)) ^:^ [; ;) )((: ^__^ (= ^-^ :))))</td>
</tr>
<tr>
<td>sad</td>
<td>:( /-_- -. -:( :'( d: :</td>
</tr>
<tr>
<td>love</td>
<td>&lt;3 xoxo &lt;33 xo &lt;333 #love s2 &lt;URL-twitition.com&gt; #neversaynever &lt;3333</td>
</tr>
<tr>
<td>F-word + ing</td>
<td>fucking fuckin freaking bloody freakin friggin effin effing fuckn fucken frickin fukin f'n fckn flippin fkn motherfucking fckin f*cking fricken fuknuccin fcking fukkin</td>
</tr>
</tbody>
</table>
Browse our Twitter Clusters!

http://www.ark.cs.cmu.edu/TweetNLP/cluster_viewer.html
Additional Notes

• Soft clustering allows items to have *mixed membership* in different clusters.
  – Typically accomplished with probabilistic models
  – Latent Dirichlet allocation is a popular and Bayesian model

• Evaluation?

• One view of clusters: feature creation!
Summary

supervised classification

(5 steps: data, features, prediction function, learning, evaluation)

structured prediction

local factoring + dynamic programming

alternating or greedy optimization

unsupervised clustering