Announcements

- Project 5: Classification up now!
  - Due date now after contest
  - Also: drop-the-lowest

- Contest: In progress!
  - New staff bot (w/ extra credit)
  - New achievements

So Far: Foundational Methods

- Information retrieval:
  - Given information needs, produce information
  - Includes, e.g. web search, question answering, and classic IR

- Web search: not exactly classification, but rather ranking

Now: Advanced Applications

Web Search / IR

- $x = \text{"Apple Computers"}$

Feature-Based Ranking

- $x = \text{"Apple Computers"}$

\[
f(x) = [0.3 \ 5 \ 0 \ 0 \ \ldots]\]

\[
f(x) = [0.8 \ 4 \ 2 \ 1 \ \ldots]\]
Perceptron for Ranking

- Inputs $x$
- Candidates $y$
- Many feature vectors: $f(x, y)$
- One weight vector: $w$
  - Prediction:
    \[ y = \text{arg max}_y \ w \cdot f(x, y) \]
  - Update (if wrong):
    \[ w = w + f(x, y^*) - f(x, y) \]

Inverse RL: Motivation

- How do we specify a task like this?

Autonomous Helicopter Setup

- State: $s = (x, y, z, \phi, \theta, \dot{x}, \dot{y}, \dot{z}, \dot{\psi}, \dot{\theta}, \ddot{\psi})$
- Actions (control inputs):
  - $a_{\text{main_rot}}$: Main rotor longitudinal cyclic pitch control (affects pitch rate)
  - $a_{\text{main_roll}}$: Main rotor latitudinal cyclic pitch control (affects roll rate)
  - $a_{\text{tail_rot}}$: Main rotor collective pitch (affects main rotor thrust)
  - $a_{\text{tail_roll}}$: Tail rotor collective pitch (affects tail rotor thrust)
- Transitions (dynamics):
  - $s_{t+1} = f(s_t, a_t) + w_i$
  - $f$ encodes helicopter dynamics
  - $w_i$ is a probabilistic noise model
- Can we solve the MDP yet?

Helicopter MDP

- Rewards for hovering:
  \[ R(s) = -\alpha_1(x - z)^2 + \alpha_2(y - y^*)^2 + \alpha_3(z - z^*)^2 + \alpha_4(\dot{x} - \dot{x}^*)^2 + \alpha_5(\dot{y} - \dot{y}^*)^2 + \alpha_6(\dot{z} - \dot{z}^*)^2 \]
- Rewards for "Tic-Toc"?
  - Problem: what’s the target trajectory?
  - Just write it down by hand?

Problem: What’s the Reward?

- Rewards for hovering:
- Rewards for "Tic-Toc"?

Apprenticeship Learning

- Goal: learn reward function from expert demonstration
- Assume $R(s) = w \cdot f(s)$
- Get expert demonstrations $s = (s_0, s_1, \ldots, s_n)$
- Guess initial policy $\pi_0$
- Repeat:
  - Find $w_i$ which make the expert better than $\{\pi_0, \pi_1, \ldots, \pi_{i-1}\}$
    \[ w_i \leftarrow \text{distinguish}(\pi^*, \{\pi_0, \pi_1, \ldots, \pi_{i-1}\}) \]
  - Solve MDP for new weights $w_i$:
    \[ \pi_i \leftarrow \text{solve}(\text{MDP}(w_i)) \]
Pacman Apprenticeship!

- Demonstrations are expert games
- Features defined over states $s$
- Score of a state given by: $w \cdot f(s)$
- Learning goal: find weights which explain expert actions

Helicopter Apprenticeship?

Probabilistic Alignment

- Intended trajectory satisfies dynamics.
- Expert trajectory is a noisy observation of one of the hidden states.
  - But we don’t know exactly which one.

Alignment of Samples

- Result: inferred sequence is much cleaner!

Final Behavior

What is NLP?

- Fundamental goal: analyze and process human language, broadly, robustly, accurately…
- End systems that we want to build:
  - Ambitious: speech recognition, machine translation, information extraction, dialog interfaces, question answering...
  - Modest: spelling correction, text categorization…
Problem: Ambiguities

- **Headlines:**
  - Enraged Cow Injures Farmer With Ax
  - Hospitals Are Sued by 7 Foot Doctors
  - Ban on Nude Dancing on Governor’s Desk
  - Iraqi Head Seeks Arms
  - Local HS Dropouts Cut in Half
  - Juvenile Court to Try Shooting Defendant
  - Stolen Painting Found by Tree
  - Kids Make Nutritious Snacks

- Why are these funny?

Grammar: PCFGs

- Natural language grammars are very ambiguous!
- PCFGs are a formal probabilistic model of trees
  - Each “rule” has a conditional probability (like an HMM)
  - Tree’s probability is the product of all rules used
- Parsing: Given a sentence, find the best tree – search!

```
ROOT → S 375/420
S → NP VP . 320/392
NP → PRP 127/539
VP → VBD ADJP 32/401
```

Syntactic Analysis

- Machine Translation

- Translate text from one language to another
- Recombines fragments of example translations
- Challenges:
  - What fragments? [learning to translate]
  - How to make efficient? [fast translation search]

The Problem with Dictionary Look-ups

Example from Douglas Hofstadter
A Brief and Biased History

When I look at an article in Russian, I say, "This is really written in English, but it has been coded into some strange symbols, I will now proceed to decode."

*Machine Translation* precisely means going by automated from machine; reliable source text to useful target text. In this context, there has been no machine translation...

Berkeley’s first MT grant

MT is the “first” non-numeral compute task

ALPAC report deems MT bad

Statistical MT thrives

47 58 66 90’s 00’s

Data-Driven Machine Translation

Target language corpus:

I will get it soon  See you later  He will do it

Sentence-aligned parallel corpus:

Yo lo haré mañana  I will do it tomorrow

Quisitos pronto  See you soon

Quisitos pronto  See you around

Machine translation system:

Yo lo haré pronto

Nokia: Sentiendo

Model of translation

I will do it soon

Learning to Translate

CLASSIC SOUPS

<table>
<thead>
<tr>
<th>Sm.</th>
<th>Lg.</th>
</tr>
</thead>
<tbody>
<tr>
<td>57.</td>
<td>House Chicken Soup (Chicken, Celery, Potato, Onion, Carrot)</td>
</tr>
<tr>
<td>58.</td>
<td>Chicken Rice Soup</td>
</tr>
<tr>
<td>59.</td>
<td>Chicken Noodle Soup</td>
</tr>
<tr>
<td>60.</td>
<td>Carrot Noodle soup</td>
</tr>
<tr>
<td>61.</td>
<td>Tomato Puree Drop Soup</td>
</tr>
<tr>
<td>62.</td>
<td>Macaroni Drop Soup</td>
</tr>
<tr>
<td>63.</td>
<td>Hot &amp; Sour Soup</td>
</tr>
<tr>
<td>64.</td>
<td>Egg Drop Soup</td>
</tr>
<tr>
<td>65.</td>
<td>Three-Broth Soup</td>
</tr>
<tr>
<td>66.</td>
<td>Tofu Vegetable Soup</td>
</tr>
<tr>
<td>67.</td>
<td>Chicken Corn Cream Soup</td>
</tr>
<tr>
<td>68.</td>
<td>Crab Meat Corn Cream Soup</td>
</tr>
<tr>
<td>69.</td>
<td>Seafood Soup</td>
</tr>
</tbody>
</table>

Example from Alan Lopez

The HMM Model

Model Parameters

Emissions: P(A) = Gracias (E) = Thank  P(A) = 2 (A) = 1

Levels of Transfer

Machine Translation

Yo lo hare de muy buen grado

Después lo verás

Model of translation

I will do it later
A Statistical Translation Model

Synchronous Derivation

S

Yo lo haré después

S

I will do it later

Synchronous Grammar Rules

S → (Yo lo haré ADV ; I will do it ADV )
ADV → (después ; later )

A Statistical Model

Translation model components factor over applied rules

How will the these rules supported by the data?

Language model factors over n-grams

How will the these rule be counted supported by the data?