CS674 Natural Language Processing

- Last class
  - Likelihood computation for spelling correction
  - Minimum edit distance
- Today
  - Bayesian method for pronunciation

The knights who say 'Ni'

[spooky music][music stops]

Head Knight of Ni: Ni!

Knights of Ni: Ni! Ni! Ni! Ni! Ni!

Arthur: Who are you?

Head Knight: We are the Knights Who Say…'Ni'! …

We are the keepers of the sacred words: 'Ni', 'Peng',

The pronunciation subproblem

- Given a series of phones, compute the most probable word that generated them.
- Simplifications
  - Given the correct string of phones
    - » Speech recognizer relies on probabilistic estimators for each phone, so it's never entirely sure about the identification of any particular phone
  - Given word boundaries
- "I [ni]…"
  - [ni] → *the, neat, need, new, knee, to,* and *you*
  - Based on the (transcribed) Switchboard corpus
- Contextually-induced pronunciation variation

No candidate generation

- Use corpus to expand each pronunciation in advance with all possible variants
- [ni] is stored with the list of words that can generate it

Scoring the candidates

- Compute

$$\hat{w} = \underset{w \in W}{\arg\max} \overbrace{P(y \mid w)}^{\text{likelihood}} \overbrace{P(w)}^{\text{prior}}$$

- where *y* represents the sequence of phones (e.g. [ni])
- and *w* represents the candidate word

Probabilistic rules for generating pronunciation likelihoods

- Take the rules of pronunciation (see chapter 4 of J&M) and associate them with probabilities
  - Nasal assimilation rule:
- Compute the probabilities from a large labeled corpus (like the transcribed portion of Switchboard)
- Run the rules over the lexicon to generate different possible surface forms each with its own probability

Sample rules that account for [ni]

Word	Rule Name	Rule	P
the	nasal assimilation	$\eth \Rightarrow n$ / [+*nasal*] # ___	[.15]
neat	final t deletion	$t \Rightarrow \emptyset$ / V ___ #	[.52]
need	final d deletion	$d \Rightarrow \emptyset$ / V ___ #	[.11]
new	u fronting	$u \Rightarrow i$ / ___ # [*y*]	[.36]

Computing the prior

- Using the relative frequency of the word in a large corpus
  - Brown corpus and Switchboard Treebank

w	freq(w)	P(w)
knee	61	.000024
the	114,834	.046
neat	338	.00013
need	1417	.00056
new	2625	.001

Final results

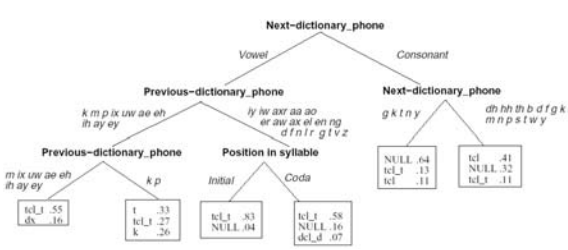
- *new* is the most likely
- Turns out to be wrong
  - *"I [ni]…"*

w	p(y|w)	p(w)	p(y|w)p(w)
new	.36	.001	.00036
neat	.52	.00013	.000068
need	.11	.00056	.000062
knee	1.00	.000024	.000024
the	0	.046	0

Decision trees for encoding lexical-to-surface pronunciation mappings

- Alternative to writing probabilistic pronunciation rules by hand is to learn the rules
- Decision tree approach
  - Riley (1991), Withgott and Chen (1993)
- Input to decision tree: a lexical phone described in terms of a set of features
- Output: classification and a probability

Example



Automatic induction of decision trees

- Riley / Withgott and Chen
  - Used CART (Breiman et al. 1984)
  - C4.5 is an alternative
- How are decision trees induced automatically?
  - Training examples
  - Top-down induction

Training data

- One tree for each lexical phone, p
  - One example for each occurrence lexical phone in corpus
  - Class value: surface realization of p
  - Features: previous-lexical-phone, next-lexical-phone, position-in-syllable