CS674 Natural Language Processing

- Last week
  - Word sense disambiguation
- Today
  - SENSEVAL
  - Noisy channel model
    - » Pronunciation variation in speech recognition

SENSEVAL-2 2001

- Three tasks
  - Lexical sample
  - All-words
  - Translation
- 12 languages
- Lexicon
  - SENSEVAL-1: from HECTOR corpus
  - SENSEVAL-2: from WordNet 1.7
- 93 systems from 34 teams

Lexical sample task

- Select a sample of words from the lexicon
- Systems must then tag instances of the sample words in short extracts of text
- SENSEVAL-1: 35 words, 41 tasks
  - 700001 John Dos Passos wrote a poem that talked of `the <tag>bitter</> beat look, the scorn on the lip."
  - 700002 The beans almost double in size during roasting. Black beans are over roasted and will have a <tag>bitter</> flavour and insufficiently roasted beans are pale and give a colourless, tasteless drink.

Lexical sample task: SENSEVAL-1

Nouns		Verbs		Adjectives		Indeterminates	
-n	N	**-v**	N	**-a**	N	**-p**	N
accident	267	amaze	70	brilliant	229	band	302
behaviour	279	bet	177	deaf	122	bitter	373
bet	274	bother	209	floating	47	hurdle	323
disability	160	bury	201	generous	227	sanction	431
excess	186	calculate	217	giant	97	shake	356
float	75	consume	186	modest	270		
giant	118	derive	216	slight	218		
…	…	…	…	…	…		
TOTAL	2756	TOTAL	2501	TOTAL	1406	TOTAL	1785

All-words task

- Systems must tag almost all of the content words in a sample of running text
  - sense-tag all predicates, nouns that are heads of noun-phrase arguments to those predicates, and adjectives modifying those nouns
  - ~5,000 running words of text
  - ~2,000 sense-tagged words

Translation task

- SENSEVAL-2 task
- Only for Japanese
- word sense is defined according to translation distinction
  - if the head word is translated differently in the given expressional context, then it is treated as constituting a different sense
- word sense disambiguation involves selecting the appropriate English word/phrase/sentence equivalent for a Japanese word

SENSEVAL-2 results

Language	Task	No. of submissions	No. of teams	IAA	Baseline	Best system
Czech	AW	1	1	-	-	.94
Basque	LS	3	2	.75	.65	.76
Estonian	AW	2	2	.72	.85	.67
Italian	LS	2	2	-	-	.39
Korean	LS	2	2	-	.71	.74
Spanish	LS	12	5	.64	.48	.65
Swedish	LS	8	5	.95	-	.70
Japanese	LS	7	3	.86	.72	.78
Japanese	TL	9	8	.81	.37	.79
English	AW	21	12	.75	.57	.69
English	LS	26	15	.86	.51/.16	.64/.40

SENSEVAL-2 de-briefing

- Where next?
  - Supervised ML approaches worked best
    - » Looking at the role of feature selection algorithms
  - Need a well-motivated sense inventory
    - » Inter-annotator agreement went down when moving to WordNet senses
  - Need to tie WSD to real applications
    - » The translation task was a good initial attempt

SENSEVAL-3 2004

- 14 core WSD tasks including
  - All words (Eng, Italian): 5000 word sample
  - Lexical sample (7 languages)
- Tasks for identifying semantic roles, for multilingual annotations, logical form, subcategorization frame acquisition

English lexcial sample task

- **Data collected from the Web from Web users**
- Guarantee at least two word senses per word
- 60 ambiguous nouns, adjectives, and verbs
- test data
  - ½ created by lexicographers
  - ½ from the web-based corpus
- Senses from WordNet 1.7.1 and **Wordsmyth** (verbs)
- Sense maps provided for fine-to-coarse sense mapping
- **Filter out multi-word expressions from data sets**

English lexical sample task

Class	Nr of words	Avg senses (fine)	Avg senses (coarse)
Nouns	20	5.8	4.35
Verbs	32	6.31	4.59
Adjectives	5	10.2	9.8
Total	57	6.47	4.96

Table 1: Summary of the sense inventory

Results

- 27 teams, 47 systems
- Most frequent sense baseline
  - 55.2% (fine-grained)
  - 64.5% (coarse)
- Most systems significantly above baseline
  - Including some unsupervised systems
- Best system
  - 72.9% (fine-grained)
  - 79.3% (coarse)

The pronunciation subproblem

[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', and 'Neee-wom'!

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] → *neat, the, need, new, knee, to,* and *you*
  - Based on the (transcribed) Switchboard corpus
- Contextually-induced pronunciation variation

Probabilistic transduction

- surface representation → lexical representation
- string of symbols representing the pronunciation of a word in context → string of symbols representing the dictionary pronunciation
  - [er] → *her, were, are, their, your*
  - exacerbated by **pronunciation variation**
    - » *the* pronounced as THEE or THUH
    - » some aspects of this variation are systematic
- sequence of letters in a mis-spelled word → sequence of letters in the correctly spelled word
  - *acress* → *actress, cress, acres*

Noisy channel model

- Channel introduces noise which makes it hard to recognize the true word.
- **Goal:** build a model of the channel so that we can figure out how it modified the true word…so that we can recover it.

Decoding algorithm

- Special case of **Bayesian inference**
  - Bayesian classification
    - » Given observation, determine which of a set of classes it belongs to.
    - » Observation
      - string of phones
    - » Classify as a
      - word in the language

Pronunciation subproblem

- Given a string of phones, *O* (e.g. [ni]), determine which word from the lexicon corresponds to it
  - Consider all words in the vocabulary, *V*
  - Select the single word, *w,* such that *P (word w* | *observation O)* is highest

$$\hat{w} = \underset{w \in V}{\arg\max} \quad P(w \mid O)$$

Bayesian approach

- Use Bayes' rule to transform into a product of two probabilities, each of which is easier to compute than *P(w|O)*

$$P(x \mid y) = \frac{P(y \mid x) \quad P(x)}{P(y)}$$

$$\hat{w} = \underset{w \in V}{\arg\max} \quad \frac{\overbrace{P(O \mid w)}^{\text{likelihood}} \quad \overbrace{P(w)}^{\text{prior}}}{P(O)}$$

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

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	$ð \Rightarrow n$ / [$+nasal$] # ___	[.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]

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