Comparing a Linguistic and a Stochastic Tagger

Christer Samuelsson

Atro Voutilainen

Comparing a Linguistic and a Stochastic Tagger

- Compares HMM to EngCG-2
  - HMM is statistical, EngCG is based on hand-coded linguistic rules
- An attempt to allay fears of bias in previous EngCG results
  - Original EngCG reported 99.7% correct analysis (with some small ambiguity remaining)
  - The validity of these results was questioned
- Skeptics say:
  - Even human linguists can only agree 97% of the time – how can a machine get 99% accuracy?
  - Test corpus may be biased towards high performance for EngCG
  - EngCG tag set may be very basic, making POS tagging easy
  - Low error rate may be due to high remaining ambiguity

Background: How Does EngCG Work?

- Sequentially applied modules
  - Morphological Analyser
    - Assigns all possible POS tags to words, e.g.

```
"<free>"
        "free" A ABS
        "free" <SVO> V SUBJUNCTIVE VFIN
        "free" <SVO> V IMP VFIN
        "free" <SVO> V INF
        "free" <SVO> V PRES -SG3 VFIN
```

  - Heuristics determine possible POS tags of unseen words
  - Disambiguator
    - Remove illegitimate analyses. Can leave ambiguity!
  - Optionally, application-specific heuristics / statistical disambiguators for still ambiguous words

Background: How Does EngCG Work?

- The Disambiguator
  - Multiple passes (5 subgrammars)
    - Starts with very reliable rules, e.g.

```
REMOVE (V)
        (-1C DET) ;
```

    - Proceeds to rough heuristics; error rates increase to 10% - 30% in final two subgrammars
  -

Subgramar	# Rules	% Extra Senses Removed	% Correct Remain
1	2967	91.70	99.88
2	158	92.87	99.86
3	374	94.42	99.85
4	71	95.74	99.71
5	44	96.54	99.55

Issue One: Maximum Accuracy

- Samuelsson and Voutilainen believe inter-linguist agreement can approach 100%
- In creating benchmark corpus, agreement between two experts was measured at 99.3% before corrections
  - After correction of simple errors, agreement reached 99.96%
- Two special approaches in their case
  - EngCG tag set avoids semantically-motivated tags
  - Linguists have “Grammarian’s Manual” of most common ambiguous cases and their correct resolution
- Using statistical tests, can determine that there is a 95% chance that human evaluators agree more than 99.2% of the time on average (in these conditions)

Issue Two: Bias in Corpora

- Because the paper’s focus is on unbiased comparison, the methods used to create corpora are especially important
- Two corpora were used:
  - Training corpus: 357,000-word sample from Brown corpus
    - Used to train HMM
  - Test corpus:
    - 55,000-word sample of journalistic, scientific, and manual texts
      - No subject overlap with training corpus
        - Helps EngCG?

The Training Corpus

- Training corpus annotated with EngCG tags
  - First pass was original EngCG algorithm
  - Ambiguities resolved by expert
  - Used in testing EngCG-2; continually improved as new rules were tested and deployed
- Does this lead to a bias favoring EngCG?

The Training Corpus

- Training corpus annotated with EngCG tags
  - First pass was original EngCG algorithm
  - Ambiguities resolved by expert
  - Used in testing EngCG-2; continually improved as new rules were tested and deployed
- Does this lead to a bias favoring EngCG?
  - If tagged by EngCG, sets an upper bound on how well HMM can perform
    - Imagine if EngCG were only 50% accurate – HMM could never do better than 50%
  - However, a standard practice in NLP
    - Given many iterations of testing and correction, most incorrect classifications were most likely weeded out

The Test Corpus

- First analyzed using only morphological analyzer
- Independently disambiguated by two linguists
  - Agreement reached 99.96%
  - After correcting for clerical errors, only disagreement was on 21 words (out of 55,000) genuinely ambiguous at the meaning level
- Final "consensus corpus" made from one of two disambiguated versions

Issue Three: Simple Tagset

- Idea is that EngCG performs so well only because the tagset it uses is so simple as to make annotating copora trivial
- While one can't compare tagsets directly, their relative "difficulty" can be compared by training the same algorithm with two different tagsets and comparing error rates
  - In this case, the HMM model's performance with the EngCG tagset was compared to its performance with more common tagsets, and was found to be similar

Issue Four: Ambiguity / Accuracy Tradeoff

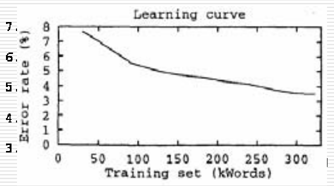
- Could be that EngCG performs so well only because of ambiguity that remains in POS assignments
  - Can't be disproven without forcing EngCG to fully disambiguate
- Rather than removing ambiguity from EngCG, authors decided to allow it in HMM
  - When annotating with HMM, allow tags with probabilities over a certain threshold to be assigned to the word in addition to the most probable tag
  - By varying threshold, vary allowable ambiguity
    - So, can set HMM ambiguity equivalent to EngCG ambiguity
- Issues?
  - HMM was not designed to work this way
  - May not take advantage of allowed ambiguity as much as EngCG

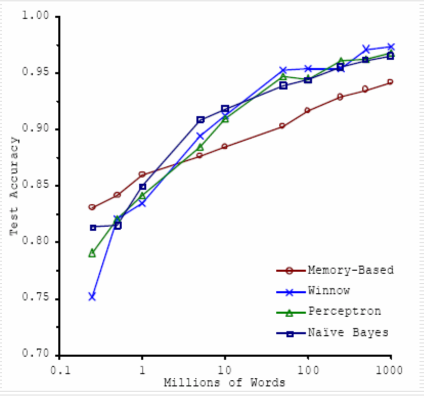
Experiment

- First, test HMM on Brown corpus at various training set sizes
  - Hold back 35,000 words from training corpus
  - Train HMM on successively larger chunks of remaining words, evaluating on held back subset
- Main experiment
  - HMM:
    - Train HMM on full 357,000 words
    - Test on 55,000 word test corpus at varying levels of allowable ambiguity
  - EngCG:
    - Run on entire training corpus at varying levels of ambiguity (number of subgrammars used)
  - Compare HMM and EngCG at same ambiguity levels

Results: HMM Testing

- Learning curve of HMM with respect to training set size

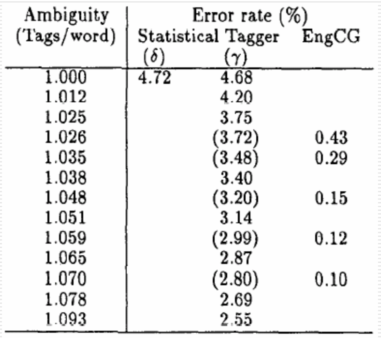




- Paper states “has leveled off at 322,000 words, indicating that little is to be gained from further training”
  - Has it?
  - Remember Scaling to Very Very Large Corpora for Natural Language Disambiguation

Results: Algorithm Comparison

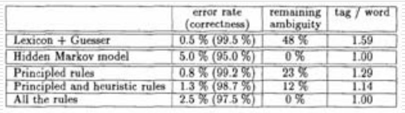
- EngCG dominates at comparable ambiguity levels
  - EngCG’s error rate ranges from 8.6 to 28 times smaller than HMM’s
- However, HMM’s performance also 1% lower than when training / testing on subset of Brown corpus
  - Indicates that training HMM on a larger corpus – and/or one that included documents similar to the benchmark corpus - could improve performance



Ambiguity (Tags/word)	Error rate (%) Statistical Tagger (δ)	(γ)	EngCG
1.000	4.72	4.68	
1.012		4.20	
1.025		3.75	
1.026		(3.72)	0.43
1.035		(3.48)	0.29
1.038		3.40	
1.048		(3.20)	0.15
1.051		3.14	
1.059		(2.99)	0.12
1.065		2.87	
1.070		(2.80)	0.10
1.078		2.69	
1.093		2.55	

Discussion

- Caveats of EngCG
  - Vastly more work to create
    - However, (Chanod and Tapanainen 1995) suggest that, given a limited amount of time to create both an HMM and a constraint-based system, the constraint-based system still outperforms the HMM



	error rate (correctness)	remaining ambiguity	tag / word
Lexicon + Guesser	0.5 % (99.5 %)	48 %	1.59
Hidden Markov model	5.0 % (95.0 %)	0 %	1.00
Principled rules	0.8 % (99.2 %)	23 %	1.29
Principled and heuristic rules	1.3 % (98.7 %)	12 %	1.14
All the rules	2.5 % (97.5 %)	0 %	1.00

Figure 2: The result in a difficult test sample with many lexicon mismatches

  - Does not disambiguate fully, and therefore unsuitable for some tasks
    - Could be corrected for by using statistical tagger on remaining ambiguities
- Was the analysis valid?
  - Even if the methods weren’t perfect, was the case that EngCG outperforms HMMs convincingly made?

Discussion

- Why *does* EngCG perform so well?
  - Simple answer:
    - If you really care enough to implement 3,600 constraint rules, you get your money’s worth
  - Slightly longer answer:
    - Errors in HMMs are tough to fix
      - Tuning parameters has an unpredictable effect on performance
    - Markov assumption limits HMM’s performance
      - E.g., “**Two** of the fastest **fish**”: **Fish** could be a singular or plural noun – no way to know with bigram, trigram, or even quadrigram methods
      - Also treating words as ambiguity classes
    - Errors in rule-based methods, however, are always fixable: just add another rule