Generalizing French schwa deletion: the role of indexed constraints
Aleksei Nazarov (Utrecht University) and Brian Smith

Intro

- Two possible problems for learned models:
  - Not capturing training data (underfitting)
  - Not generalizing from training data (overfitting)
  - Tendency: tradeoff between these (Hastie et al. 2001)
- Indexed constraints designed to capture training data (Pater 2000)
  - Will this hurt generalization?
- Tested with 4 procedures for indexed constraint discovery
  - Learn from French schwa deletion corpus
  - Apply learned models to existing experimental data

Procedure

- MaxEnt (Goldwater & Johnson 2003)
  - batch-trained using hgR (Staubs 2011) from all-0 weights
- Start out without indexed constraints
  - Use word-specific gradients of constraint weights to find indexed constraints (cf. Becker 2009, Pater 2010)
- 4 increasingly involved indexation procedures
  - No indexation
  - Pre-training (use weight gradients before training)
  - Post-training (use weight gradients after training once)
  - Iterative (add one indexed constraint at a time while training in between, until convergence)

French schwa deletion (Dell 1985)

- Deletion of /ə/ (phonetically [œ]) optional in many contexts
- Rate of deletion depends on phonological context
- However, rate of deletion also depends on individual word

Examples from Racine (2008) corpus:

- e.g., VC_VC kas(o)sjoʊ 'pot'
- > #C_C s(o)sj 'canary'
- suber(o)sjoʊ 'jolt'

References


Training on corpus

- Corpus: 456 words × 12 France French speakers (Racine 2008)
  - Judgments on schwa-ful and schwa-less variants of same word
  - transformed into pseudo-frequencies
- Constraints:
- Evaluation: log-likelihood of training data (closer to 0 = better)
- As expected: data captured better as indexation gets more involved
- Performance not tied to number of indexed constraints:
  - Pre-training: 7
  - Post-training: 7
  - Iterative: 4
- More involved indexation = better

Conclusions

- Indexed constraints improve account of training data, but do not have to hurt generalization!
  - Similar to how adding random effects improves generalization in Mixed Effects models (e.g., Zymet 2018; Barr et al. 2013)
- Best generalisation behaviour when indexed constraints induced one by one, but not used in generalization to new items
- No-indexed constraint weights improved by precense of indexed constraints
- When evaluating on new data, method of generalization matters
- For example, pre-training procedure fares much worse with 0 method

Generalization to Smith & Pater (2020)

- Smith & Pater (2020): Participants choose between schwa or no schwa in French phrases containing -CCc and -Vc words followed by CV or CVVC
- Use trained procedures to predict proportion of schwa responses per context
- Constraints: *ə, *ə/non-penult, Max, *#CC, any indexed versions
- Generalization to experimental data:
  - 0 method: no indexed constraint violations
- Evaluation: sum squared error (SSE; smaller = better)
- More involved indexation = better
- 0 method of generalization slightly better?
- Except for pre-training indexation: definitely worse
- Best indexation procedure on par with No indexation

Conclusions

- Indexed constraints improve account of training data, but do not have to hurt generalization!
  - Similar to how adding random effects improves generalization in Mixed Effects models (e.g., Zymet 2018; Barr et al. 2013)
- Best generalisation behaviour when indexed constraints induced one by one, but not used in generalization to new items
- No-indexed constraint weights improved by precense of indexed constraints
- When evaluating on new data, method of generalization matters
- For example, pre-training procedure fares much worse with 0 method