Optimizing non-native speech recognition for CALL applications

Joost van Doremalen, Helmer Strik, Catia Cucchiarini

1. Introduction

- **Context**

Development of ASR-based CALL application that gives feedback at *grammar* and *pronunciation* to students of Dutch as a 2nd language with different language backgrounds

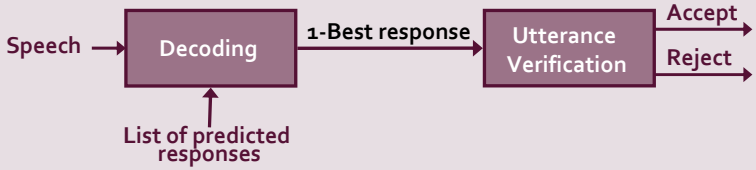
- **Problems for ASR of non-native speakers**
  - Deviations in pronunciation
  - Deviations in syntax and morphology
  - Disfluencies: *filled pauses, repetitions, restarts, repairs*

- **In our application we will use 2 strategies to deal with these problems**

(1) Constrain output of learner so that speech becomes more predictable: choose from fixed predicted list of responses

(2) Improve speech decoding

- **Application architecture**



- **Goal of this research**

Improve speech decoding by optimizing the 3 main knowledge sources: the *language model*, *acoustic models* and *pronunciation lexicon*

2. Method

- **Testing material:** non-native part of the JASMIN speech corpus containing speakers giving answers to questions about a hypothetical journey
- **45 speakers**: 60% female and 40% male, 25 different L1 backgrounds
- All speakers give answers to **39 questions**
- After deletion of crosstalk, background noise and whispering: **1325 utterances**

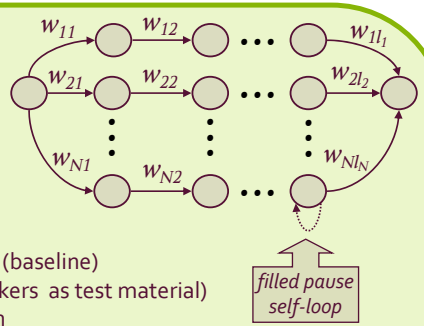
- To **simulate ASR task in application,** 39 lists of ~45 answers were extracted
- These lists **mimic lists of predicted responses** in our application because they contain:
  - Responses to relatively *closed questions*
  - Grammatically correct and incorrect responses

- **Language Model**

(1) Finite State Machine with parallel arcs for answers (baseline)

(2) Baseline + filled pause self-loops between words

(3) Baseline + filled pauses between words (taken from manual transcription) → *oracle* model



- **Acoustic Models**

(1) Triphone Gaussian Mixture Models trained on native read speech (baseline)

(2) Baseline models retrained on non-native read speech (same speakers as test material)

(3) Monophone baseline models retrained on non-native read speech

- **Pronunciation Lexicon**

(1) Canonical (baseline)

(2) Lexicon with pronunciation variants generated from rewrite rules computed from non-native read speech

(3) Same as **(2)** but with prior probabiltities calculated from rule frequencies

- **Evaluation**

Setups evaluated using Utterance Error Rate

Reference transcriptions and decoding results compared **after deletion of filled pauses**

At this stage → not necessary to discriminate between phonetically close responses: when phonetic distance to corresponding transcription is below certain threshold, result classified as correct

3. Results

		phonetic distance			
Acoustic Model	**Language Model**	**=0**	**<5**	**<10**	**<15**
native (tri)	without loops	28.9	28.4	26.1	24.6
native(tri)	with loops	14.9	14.6	12.6	11.0
native(tri)	with positions	14.7	14.4	13.1	12.0
non-native (tri)	without loops	22.4	22.4	19.9	18.4
non-native (tri)	with loops	10.0	9.7	7.9	6.9
non-native (tri)	with positions	9.4	9.1	7.8	7.1
non-native (mono)	with loops	11.9	11.5	9.3	8.1

Lexicon	**Priors**	**=0**	**<5**	**<10**	**<15**
canonical	-	10.0	9.7	7.9	6.9
2 variants	no	10.0	9.9	8.2	6.7
2 variants	yes	10.0	9.7	8.3	7.0
3 variants	no	11.2	10.9	8.5	7.1
3 variants	yes	10.6	10.1	8.7	6.2
4 variants	no	11.5	11.3	8.9	7.5
4 variants	yes	10.4	10.9	9.7	7.2
5 variants	no	11.5	11.3	8.9	7.5
5 variants	yes	10.4	10.0	8.7	7.2

4. Discussion & Conclusions

- Large and significant improvements by optimizing
  - Language model: LM with loops not significantly different from oracle LM
  - Acoustic models: Using only 8 minutes per speaker as retraining material
- Pronunciation modelling did not produce significant improvements: **difficult to capture diffuse pattern of variation** with different mother tongues