Cross-lingual articulatory feature information transfer for speech recognition

Mahir Morshed
17 October 2022

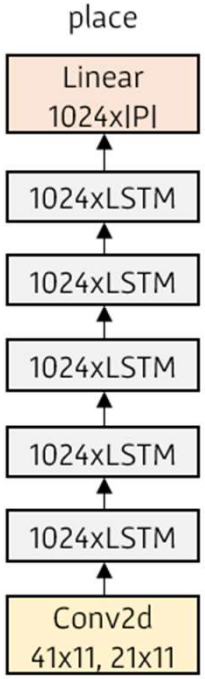
Introduction

- External linguistic domain knowledge useful!
  - Low-resource settings emphasize importance thereof
- Variation in articulatory feature detectors:
  - Binary distinctive features (doi:10.21437/Interspeech.2019-3020)
  - Non-binary articulation classes (doi:10.21437/Interspeech.2016-925)
  - Fully connected individual detectors (doi:10.1016/j.procs.2017.12.114)
  - Convolutional multi-class detectors (doi:10.21437/Interspeech.2018-2275)

Introduction

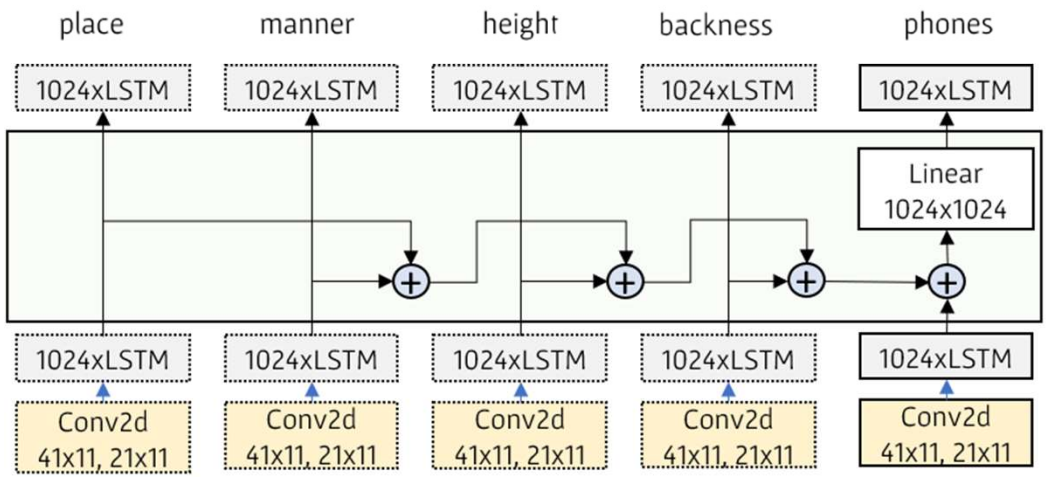
- After learning one task, other tasks easier to learn?
  - Detecting single articulatory features being prior tasks
  - Transferring learnt information to phone recognition task
  - Encouraging recognizer to learn what prior tasks did not
- Variation in end-to-end transfer learning:
  - Fully retrain (doi:10.3390/sym11020179) or partially freeze (doi:10.18653/v1/W17-2620)
  - Data-only fine-tuning (doi:10.1109/ICSDA.2016.7918980) vs. prior language model fusion (doi:10.1109/ICASSP.2019.8682918)

Setup



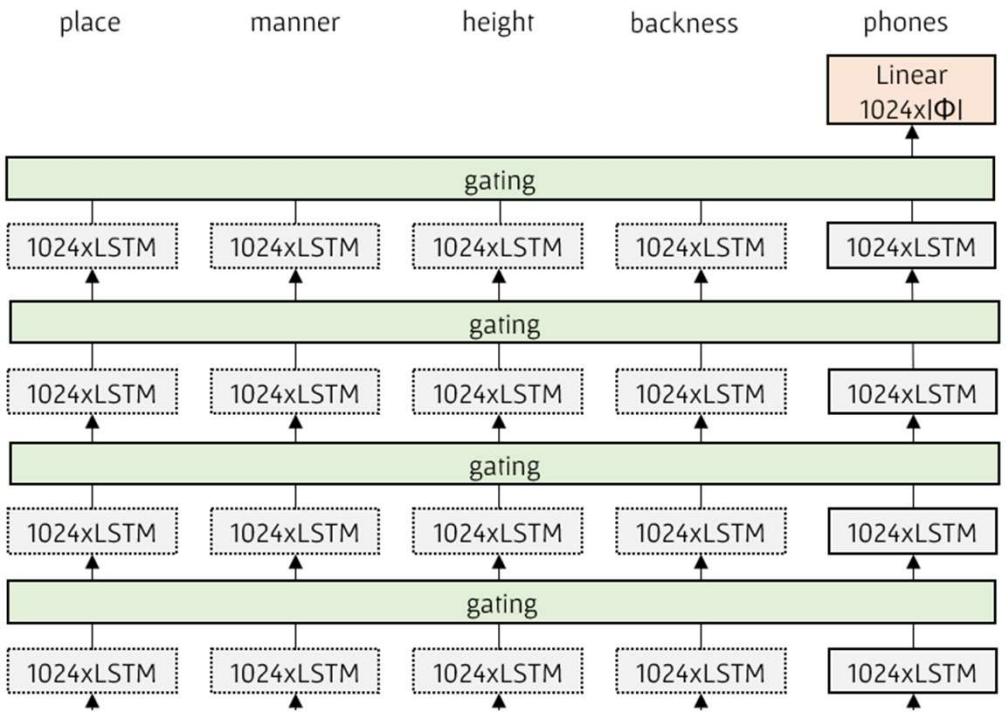
- Feature detectors: Deep Speech 2 architecture (doi:10.48550/arXiv.1512.02595)
  - Generally unchanged except for output dimension
  - In this case, P = {b, l, d, a, c, p, v, g, V, “ ”, ϵ}
    - (all places of articulation, a pseudo-character for all vowels, the space, and the CTC blank)
  - All detectors separately trained with same data (spectrogram inputs)

Setup



- Phone recognition: progressive networks
  (doi:10.48550/arXiv.1606.04671)
  - All feature detector weights otherwise frozen
  - Same spectrogram input to all networks
  - Gating mechanism used for information transfer after each recurrent layer

Setup



- Phone recognition: progressive networks
  (doi:10.48550/arXiv.1606.04671)
  - All feature detector weights otherwise frozen
  - Same spectrogram input to all networks
  - Gating mechanism used for information transfer after each recurrent layer

Dataset

- Google corpora for five South(east) Asian languages
  - Lexicon, phone set information separately available
- After filtration, some languages ‘higher-resourced’:
  - Split based on different training data sizes (10 h, 20 h, 80 h)

Bengali (`bn`, SLR53): 215.55 h → >100 h
Javanese (`jv`, SLR35): 295.88 h → >100 h
Nepali (`ne`, SLR54): 154.65 h → ~60 h
Sinhalese (`si`, SLR52): 225.6 h → >100 h
Sundanese (`su`, SLR36): 332.66 h → ~60 h

Feature sets

- Union of features across all languages, given incomplete overlap among sound annotations:
  - `bn`/`si` lack sounds annotated as mid vowels
  - `jv`/`ne`/`su` similarly lack near-open vowels
  - `ne` similarly lacks labiodentals
  - all but `ne` lack postalveolars

place: labial, labiodental, dental, alveolar, postalveolar, palatal (c), velar, glottal
manner: stop, affricate, fricative, nasal, approximant, lateral
height: close, near-close, close-mid, mid, open-mid, near-open, open (0—6)
backness: front, central, back

Feature sets

	place	manner	voicing	aspiration	prenasality	height	backness	rounding	length	schwa
bn	✓	✓	✓	✓		✓	✓	✓		
jv	✓	✓	✓	✓		✓	✓	✓	✓	✓
ne	✓	✓	✓	✓		✓	✓	✓	✓	✓
si	✓	✓	✓	✓	✓	✓	✓	✓	✓	
su	✓	✓	✓	✓		✓	✓	✓	✓	✓

- Four detectors for four non-binary features:
  - Mainly due to noted differences in realized values across languages
  - Some binary features entirely absent in some cases

Results

/baVbVd aVVvV aVbV/ vs.
/baVbVd aVvVV aVdV/ (place)

/CC6C6C C26C6 C0C6/ vs.
/C6C6C C24C6 C0C6/ (height)

/prawan seaga lima/ vs.
/prawam saga lina/ (phones)

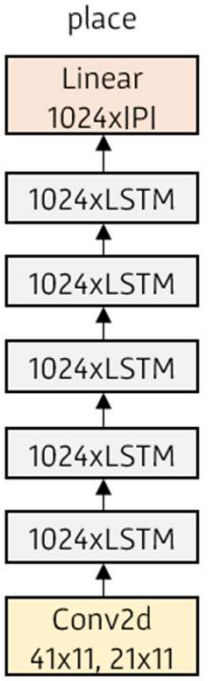
- Feature/phone stream Levenshtein distances:
  - Used for Feature Error Rates and Phone Error Rates
  - No conversion to graphemes at this time
  - (each phone mapped to exactly one feature in each class)

Results

- For comparison's sake, trained English feature detectors first:
  - Performance improved on Wall Street Journal corpus compared to prior recurrent feature detectors
  - Bengali detector results reinforced confidence

FERs	en (†)	en (ours)
place	9.4	4.24
manner	8.6	2.86
tense	8.7	3.86
back	9.2	3.82

FERs	bn (ours)
place	5.165
manner	6.165
height	5.967
position	5.052



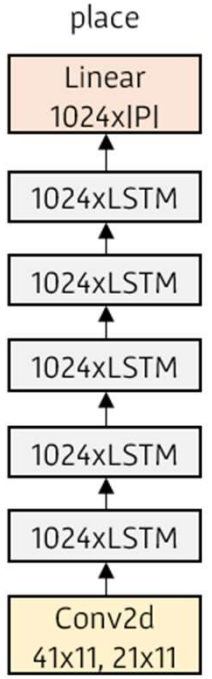
(†): Qu et al. "Combining Articulatory Features with End-to-end Learning in Speech Recognition", ICANN 2018

Results

PERs	10h	20h	80h
Javanese (jv)	17.48	12.38	11.31
Nepali (ne)	22.21	17.03	n/a
Sinhalese (si)	(*)	18.83	11.94
Sundanese (su)	3.99	2.85	n/a

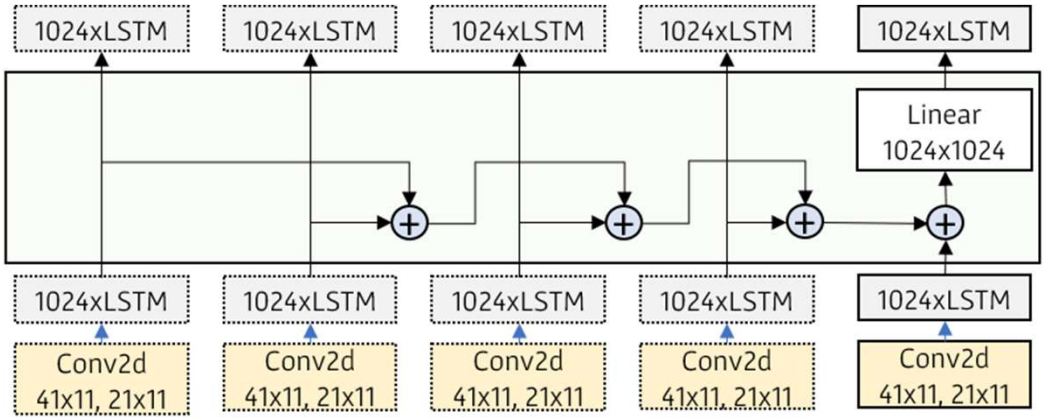
(*): did not converge

- Phone recognition baselines for defined data splits:
  - Target phone set language-specific (for jv/su) or unified (for ne/si)
  - Set choice kept for related languages



Results

- Progressive networks using all four Bengali feature detectors:
  - Increased degradation with smaller training size
  - Only one setting yielding ‘improvement’ compared to baseline



PERs	10h	20h
Javanese (jv)	22.25	13.61
Nepali (ne)	(*)	18.14
Sinhalese (si)	28.25	20.11
Sundanese (su)	43.93(†)	3.17

(*): did not converge
(†): see Discussion

Intermediate discussion

- Some greater issues:
  - Using recognizer network outputs as part of gating
  - Less overlap between `bn`'s and others' phone sets (more on this later)
- Some lesser issues:
  - Distribution of `su` data (most sentences one of two types)
  - Shorter lengths of `ne` sentences
  - Lexicon inconsistencies with produced outputs

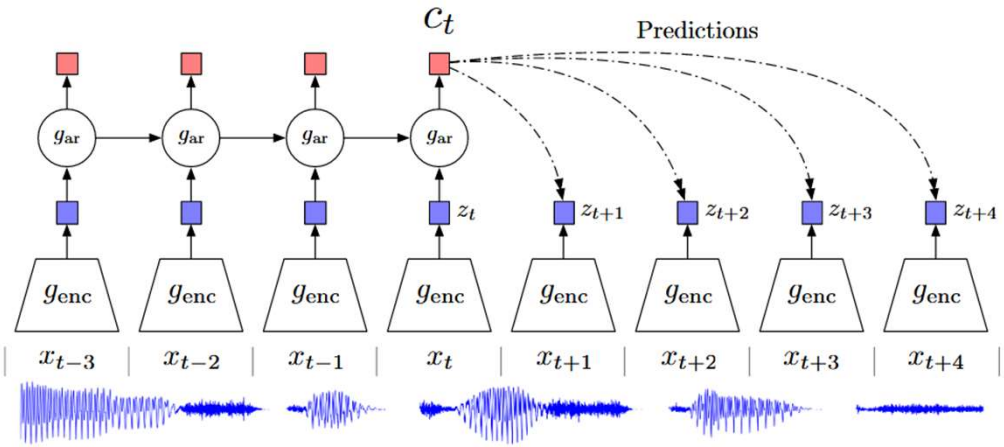
Onward…

- To handle overlap, mix training languages!
  - Using all but `su` in equal proportion
  - Approaches, but does not surpass, monolingual case

	CER 20h	CER 40h	WER 20h	WER 40h
place	13.80	7.110	68.40	40.20
manner	11.80	6.590	64.90	40.00
height	20.50	11.10	80.50	53.00
position	13.80	8.010	69.30	45.60

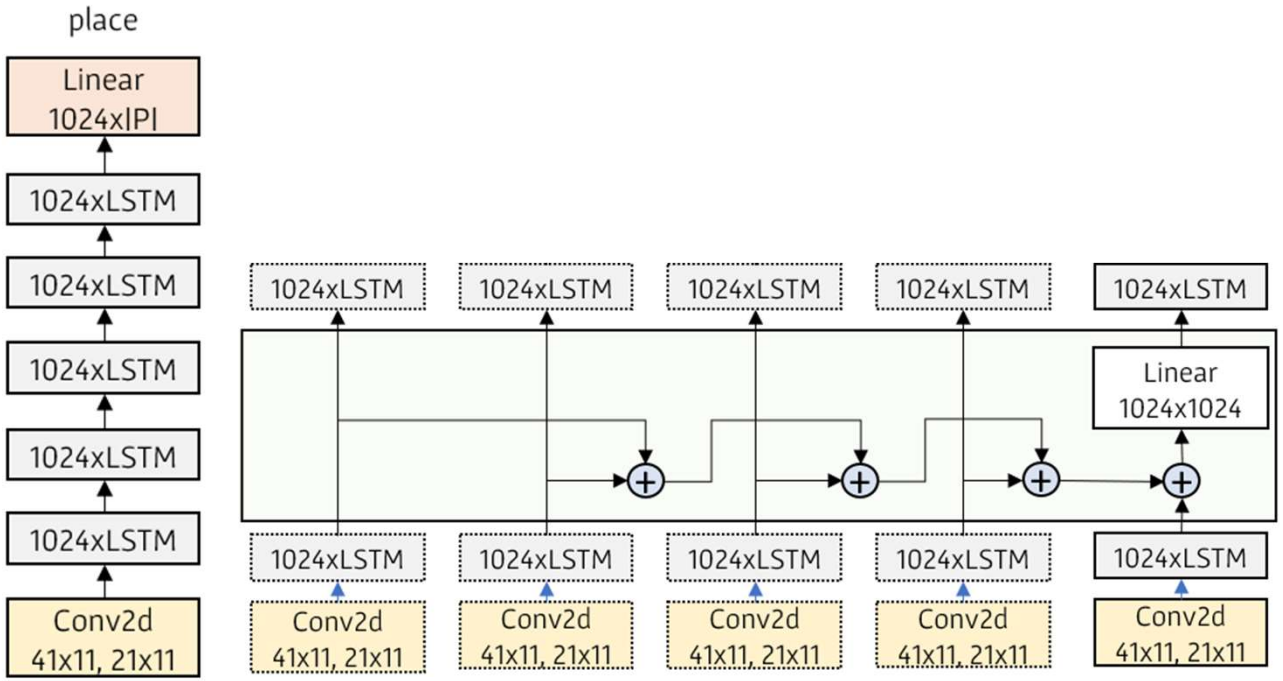
Contrastive predictive coding

(doi:10.48550/arXiv.1807.03748)



- Unsupervised representation learning:
  - Project input at timestep into compressed latent space
  - Maximize mutual information between projection and future timesteps
  - Regularizations considered—left-or-right (LorR) and self-expression (SE)

Setup



- Convolutions replaced with CPC features
  - Architecture unchanged otherwise
  - Training as before

Results

- While CPC can help, regularization can help even more
  - Development performances after 9 epochs for place detectors with Bengali data
  - Training sizes for CPC layer also significant

	FER
original	10.3
CPC 100h	9.62
CPC 360h	9.3
CPC LorR 360h	8.35
CPC SE 360h	8.19

Results

FERs	place SE	manner SE	height SE
Javanese (jv)	7.266	5.442	9.339
Nepali (ne)	7.389	6.705	8.174
Sinhalese (si)	8.921	7.217	10.896
Bengali (bn)	7.707	5.813	8.379

- Self-expression CPC doesn't particularly improve matters
  - Evaluation performance after training with 160h of four-language data
  - (PERs with convolutions averaged across languages with place 7.110, manner 6.590)

Results

- Left-or-right regularization does wonders!
  - Evaluation performances after training with 160h of four-language data

FERs	place SE	place LorR	manner SE	manner LorR
Javanese (jv)	7.266	6.741	5.442	5.073
Nepali (ne)	7.389	6.782	6.705	6.083
Sinhalese (si)	8.921	8.060	7.217	7.147
Bengali (bn)	7.707	6.947	5.813	5.750

Conclusions

- Recurrent articulatory feature detectors:
  - Better performance here than with prior recurrent detectors
- Progressive networks for information transfer:
  - Effective in one setting, but unchanging in others (new developments since then?)
- Lots of possible improvements and explorations:
  - Alternate audio input features (which you now have seen!)
  - Changes to progressive gating mechanism
  - Training feature detectors with multilingual data (which you now have seen!)

Thank you!