Anlirika: an LSTM–CNN Flow Twister for Spoken Language Identification

Andrei Shcherbakov[#], Liam Whittle[✧], Ritesh Kumar[●], Siddharth Singh[●], Matthew Coleman[✧], Ekaterina Vylomova[#]

[#] University of Melbourne [✧] Monash University
[●] Bhim Rao Ambedkar University

ultrasparc@yandex.ru

Task

"Anlirika" system was submitted to SIGTYP 2021 Shared Task on RobustSLI (Salesky et al., 2021).

The code is available at
`https://github.com/andreas-softwareengineer-pro/speech-language-classifier`

- In terms of the task, systems are trained to predict **language** id from an **audio signal**.
- Importantly, the task aims at development of robust systems that can generalize well to new domains and speakers.
  - Many languages are under-resourced and lacks speaker diversity.
  - Therefore, it is essential for a system to be speaker-invariant and robust

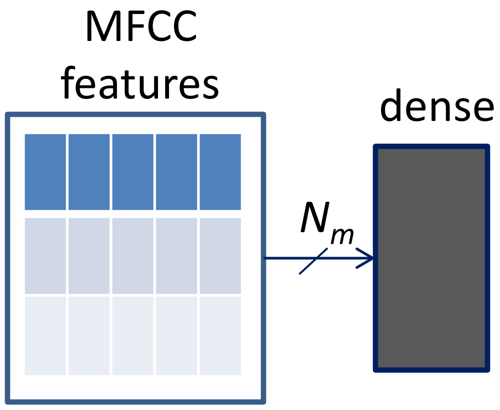
Dataset

- 16 typologically diverse languages from Afro-Asiatic, Austronesian, Basque,Dravidian, Indo-European, Niger-Congo, and Tai-Kadai families
- **Train set:** from the CMU Wilderness dataset (Black, 2019)
  - ➢ speech utterances from the Bible; predominantly a single speaker per language
  - ➢ 4,000 utterances per language
- **Validation and test sets:** from CommonVoice (Ardila et al., 2019) and other corpora
  - ➢ multiple speakers per language
  - ➢ 500 samples per language each set
- The length of each speech utterance is 3..7 seconds.
- Audio signal represented via Mel-Frequency Cepstral Coefficients (MFCC).

Architecture: motivation

1. **Remove sound harmonics**
   - ***dense layer***
2. Recognize spectral line shape
   - ***1D-CNN (convolving by input feature vector index [sound tone])***
3. Recognize ``local'' temporal constructs
   - ***an optional stack of temporal LSTMs***
4. Reduce temporal patterns into single-vector representation
   - ***LSTM***
5. Finally, classify it into one of 16 languages
   - ***dense layer***

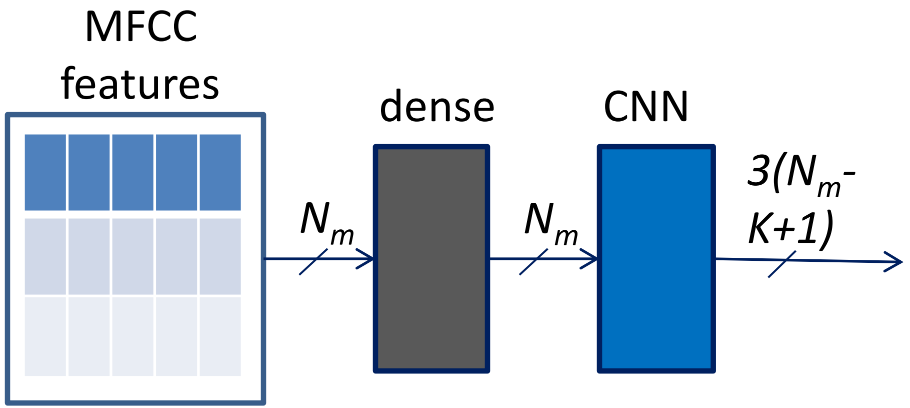
Architecture



Architecture: motivation

1. Remove sound harmonics
   - *dense layer*
2. **Recognize spectral line shape**
   - ***1D-CNN (convolving by input feature vector index [sound tone])***
3. Recognize ``local'' temporal constructs
   - ***an optional stack of temporal LSTMs***
4. Reduce temporal patterns into single-vector representation
   - ***LSTM***
5. Finally, classify it into one of 16 languages
   - ***dense layer***

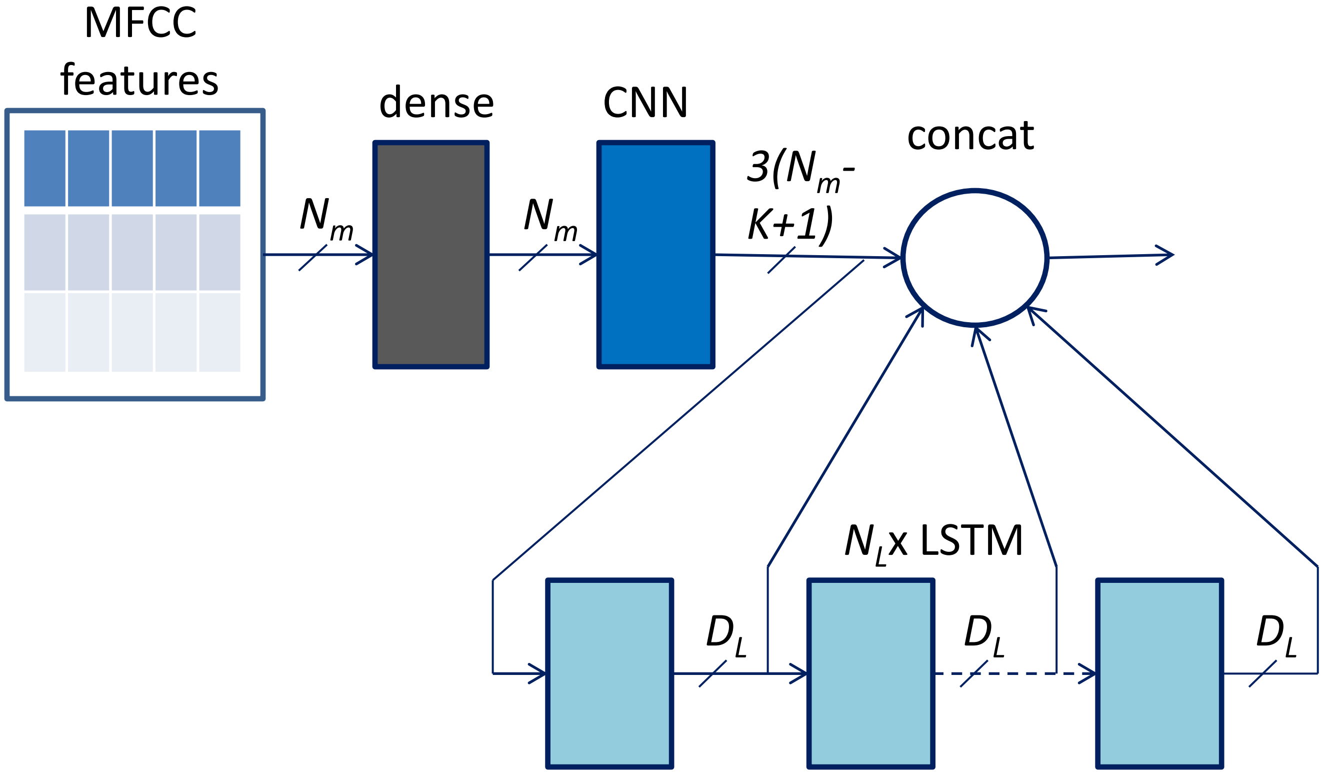
Architecture



Architecture: motivation

1. Remove sound harmonics
   - *dense layer*
2. Recognize spectral line shape
   - *1D-CNN (convolving by input feature vector index [sound tone])*
3. **Recognize ``local'' temporal constructs**
   - *an optional stack of temporal LSTMs*
4. Reduce temporal patterns into single-vector representation
   - *LSTM*
5. Finally, classify it into one of 16 languages
   - *dense layer*

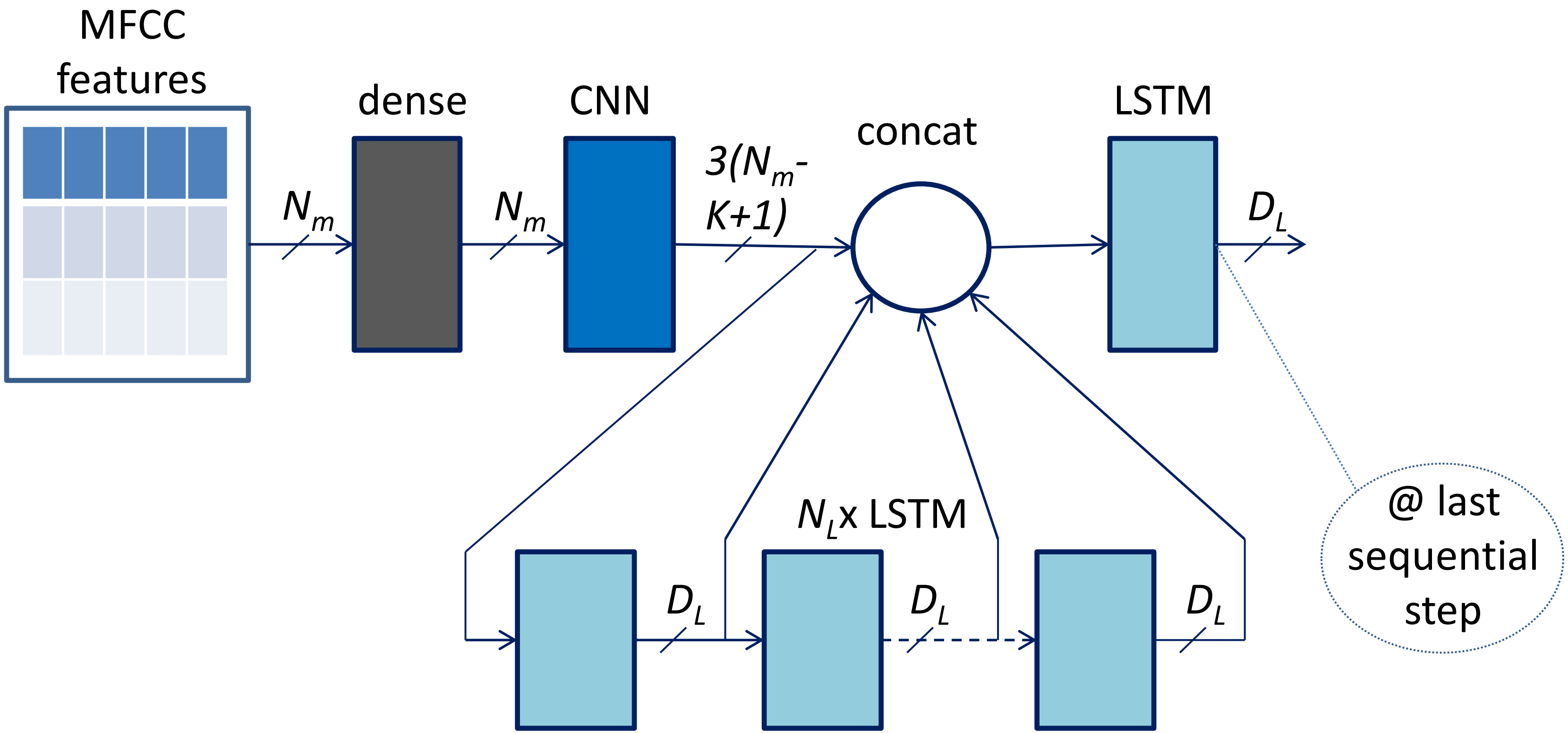
Architecture



Architecture: motivation

1. Remove sound harmonics
   - *dense layer*
2. Recognize spectral line shape
   - *1D-CNN (convolving by input feature vector index [sound tone])*
3. Recognize ``local'' temporal constructs
   - *an optional stack of temporal LSTMs*
4. **Reduce temporal patterns into single-vector representation**
   - *LSTM*
5. Finally, classify it into one of 16 languages
   - *dense layer*

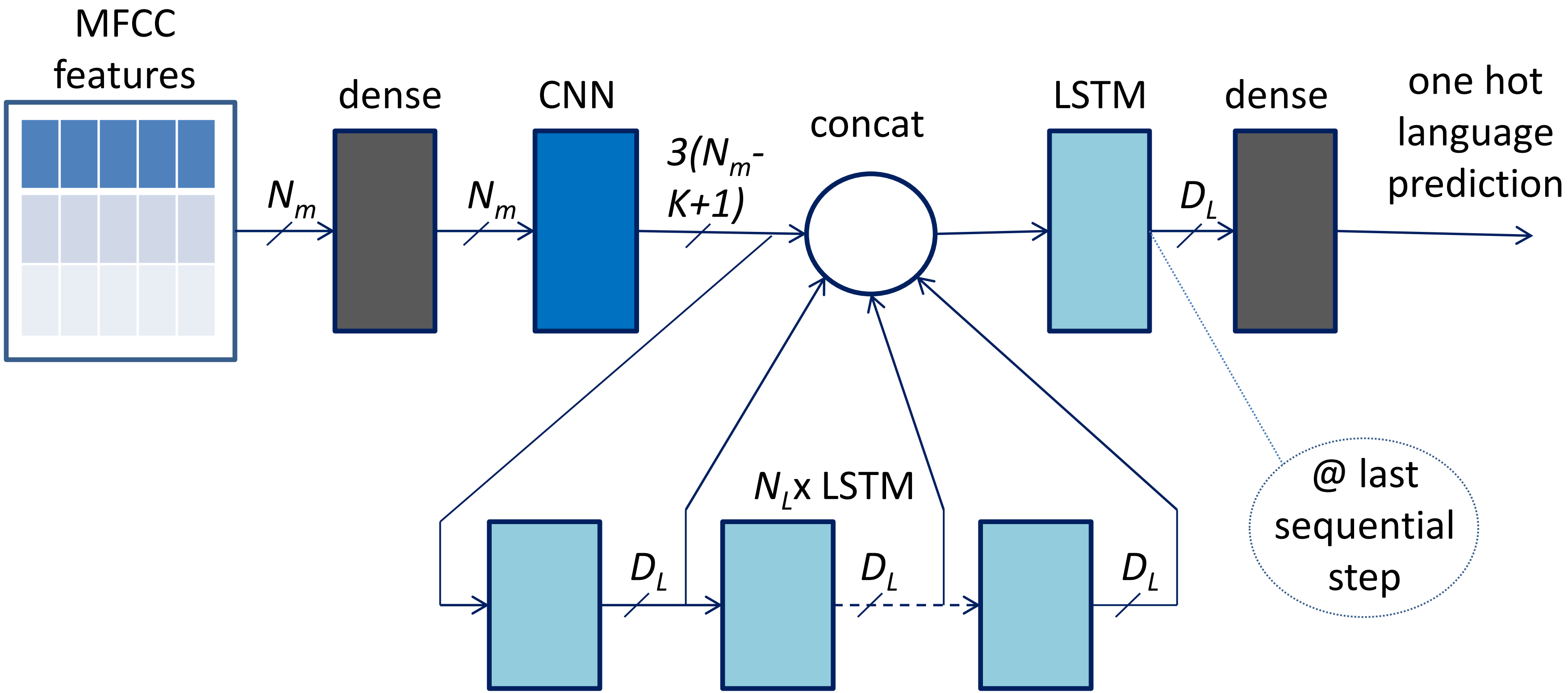
Architecture



Architecture: motivation

1. Remove sound harmonics
   - *dense layer*
2. Recognize spectral line shape
   - *1D-CNN (convolving by input feature vector index [sound tone])*
3. Recognize ``local'' temporal constructs
   - *an optional stack of temporal LSTMs*
4. Reduce temporal patterns into single-vector representation
   - *LSTM*
5. **Finally, classify it into one of 16 languages**
   - *dense layer*

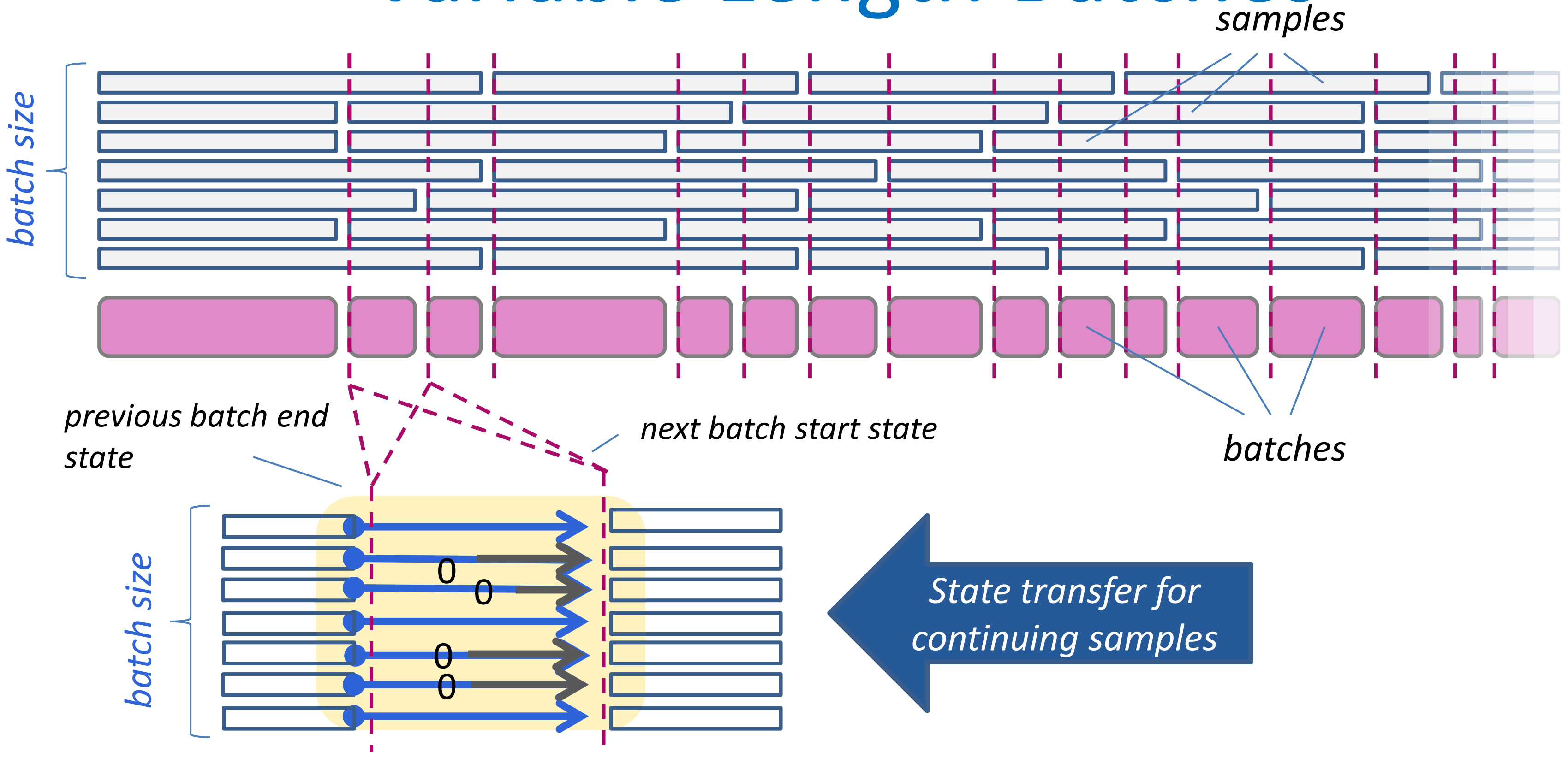
Architecture



Variable Length Batches

- A batched learning process with
  - fixed number of processed samples per batch (64)
  - variable number of time steps per batch
    - determined by the shortest sample within a given batch.
- Samples which do not fit within a batch length, are passed to the next batch for further processing, having their already-processed prefixes removed.
- Drawback: temporal depth of backpropagation through time is constrained

Variable Length Batches



Tuning of hyperparameters

- N_L – number of extra LSTM layers
  - A choice of N_L=2 was found to be producing the highest accuracy.
- D_L – output size of LSTM layers
  - Values of 200 and 300 were tried: no significant difference in performance was observed.

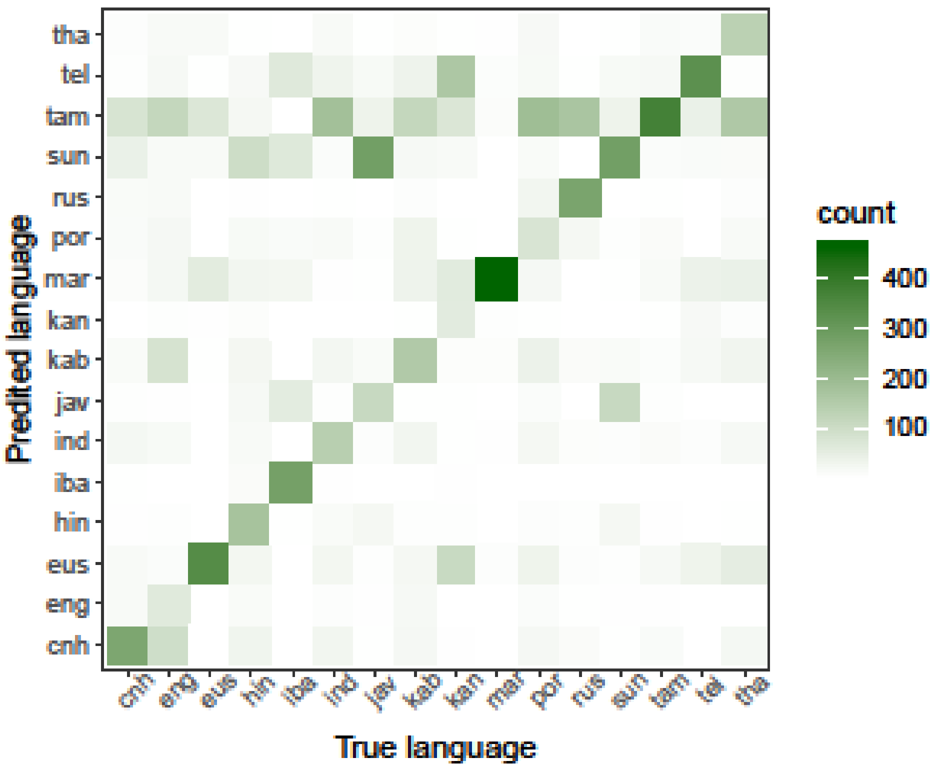
Augmenting train set

- **Using the original train set:**
  - slow learning dynamic; fails to converge at learning rates above $4{\cdot}10^{-4}$
  - Accuracy is below 12% at validation set.
- **Augmenting training data with validation set samples:**
  - A much superior accuracy of 74% on (cross-) validation set was achieved.

➢ Generalization across speakers yet remains too challenging for the system

Confusion matrix

(cross validation on augmented train set)



- Frequently overpredicts Tamil (tam) and Sundanese (sun)
- Surprisingly, fails to predict English.

Shared task submission

Set	Accuracy	Micro Avg	Micro Avg
Valid.	43.6%	43.6%	42.1%
Test	29.9%	29.8%	28.2%

- Trained on an augmented set.

Conclusion & future work

- To address the task of language classification in speech samples, we implemented and explored a neural network model inspired by an idea of phoneme sequence recognition.
- Our experiments are yet in progress, still it is clear that the generalization across domains appears to be an extremely challenging problem.
- A hypothesis to explore: Phonetic generalization may be enforced by insertion of “bottlenecks” (layers with low output size).


Thank you!
8192
4096
2048
1024
512
0