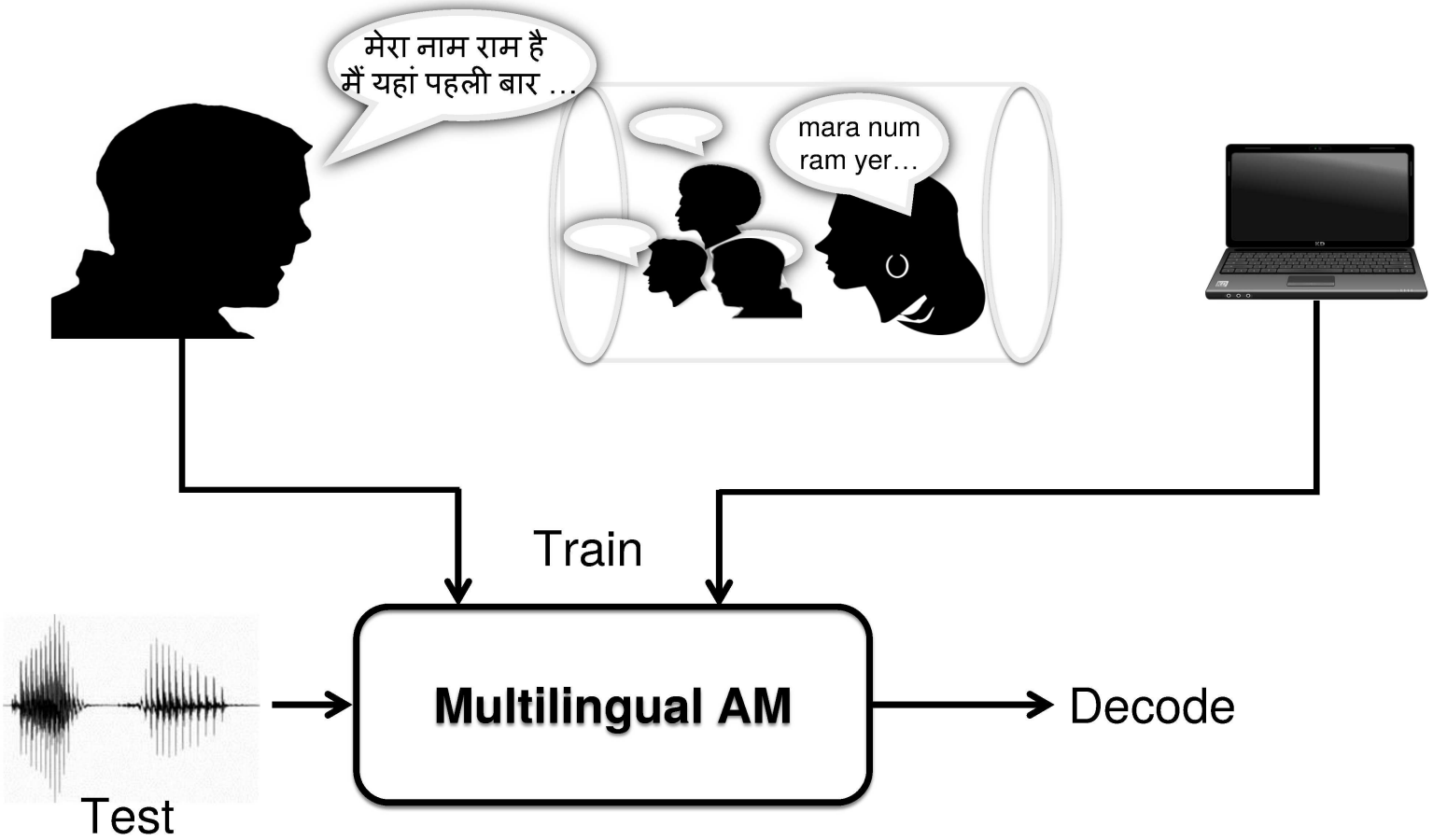
Training Deep Neural Networks Using Crowdsourced Transcriptions

Amit Das

Overview: Mismatched Crowdsourcing



Problems with mismatched transcripts

1. Type 1 Loss: Mismatched transcripts have inaccurate labels. Crowd worker not familiar with the language he/she is transcribing.
2. Type 2 Loss: Loss in information in converting English letters to IPA phones in target language (L).
   Audio in language L $\Rightarrow$ English letters $\Rightarrow$ IPA phones in L.

Nature of Data

Probabilistic Transcripts (PT)

1. PTs in Target Language: From crowd workers who neither speak nor have any familiarity with the target language.
2. PTs are limited: 40 minutes of audio.

Deterministic Transcripts (DT)

1. DTs in Target Language: None. No natively transcribed DTs in the target language.
2. DTs in Source Languages: DTs from 5 other languages none of which are the target language.
3. DTs are limited: 40 minutes of audio per language $\times$ 5 languages $=$ 200 minutes.

Unsupervised Data

1. Unsupervised data in Target Language: At least 5 hours.

Multilingual Corpus

SBS Multilingual Corpus

Language	Utterances		Phones
	Train	Test	
Swahili (SW)	463	123	53
Hungarian (HG)	459	117	70
Cantonese (CA)	544	148	37
Mandarin (MD)	467	113	57
Arabic (AR)	468	112	51
Urdu (UR)	385	94	45
All	-	-	83

- Each phone is shared by at least two languages.
- E.g.: If SW is the target language, then the training set consists of all languages except SW, i.e., HG, CA, MD, AR, UR.

Monolingual (MONO) PER

- What would be the PER if we had access to DTs in the target language?
- Train and test an ASR in the same language (no language mismatch) with 40 minutes of DTs. This system is called "MONO".
- Establish lower bound of PER since this is the best case scenario.

PERs of HMM and DNN models trained on matched monolingual DTs. Dev set in parentheses.

Lang	PER (%)	
	HMM	DNN
SW	35.63 (47.00)	34.18 (39.49)
HG	38.72 (40.33)	35.62 (37.32)
MD	31.80 (26.14)	28.26 (25.16)

Multilingual (MULTI) PER

- What would be the PER if we had no access to DTs and PTs in the target language?
- Train using mismatched languages with 200 minutes of DTs. Test on target language. This system is called "MULTI".
- Establish upper bound PER since this is the worst case scenario.

PERs of HMM and DNN models trained on mismatched multilingual DTs. Dev set in parentheses.

Lang	PER (%)		
	HMM	DNN	# Senones
SW	65.73 (67.58)	61.17 (63.12)	1003
HG	67.55 (68.50)	63.25 (63.65)	1012
MD	71.09 (69.10)	64.68 (63.84)	994

Self-training (SELF) PER

- Decode 40 minutes of target audio using MULTI.
- A subset of frames are discarded if the best path state posterior probability was below a threshold.
- The remaining frames with their best path labels are retained for re-training the MULTI system. This is the SELF system.
- PER improves by 1.01%-2.20% compared to the MULTI system.

PERs of DNN models trained using the self-training algorithm. Dev set in parentheses.

Lang	PER %		
	MULTI	SELF	PER Improve
SW	61.17 (63.12)	60.14 (62.07)	1.03
HG	63.25 (63.65)	61.05 (62.26)	2.20
MD	64.68 (63.84)	63.67 (61.94)	1.01

Using PTs for training ASR

- Note that PTs have not been used yet in ASR training.
- Next few slides explore the effect of adding PTs during HMM and DNN training.
- In particular, maximum aposteriori (MAP) HMM and three variants of DNN were used.

Single softmax DNN (DNN-1)

- Step 1: Adapt MULTI HMM system using MAP adaptation.
- Step 2: From the MAP HMM system, generate state level alignments as labels for DNN training. Since the alignments are based on PTs, the labels are **soft** instead of the traditional 1-hot when DTs are used.

Example: Alignments for a single frame

1-hot label: [1.0 *a*]

Soft labels: [0.45 *a*, 0.25 *b*, 0.3 *c*]

- Step 3: Add a randomly initialized softmax layer and fine tune the DNN using the soft labels and cross-entropy training objective.

Single softmax DNN (DNN-1) PER

PERs of HMM and DNN models trained on PTs of the target language. First element in parentheses is the PER of the dev set. Second element is the absolute improvement in PER of the test set over MAP HMM.

Lang	PER (%)	
	MAP HMM	DNN-1
SW	44.77 (50.97,0.0)	45.14 (47.83,-0.37)
HG	56.85 (57.69,0.0)	56.13 (57.21,0.72)
MD	59.23 (58.05,0.0)	54.95 (54.35,4.28)

- Clearly, PER improvements are not consistent. PER worse for SW.

Two softmax DNN (DNN-2) PER

- Use 2 softmax layers, both sharing the same hidden layers. "Y" configuration.
- Softmax # 1: Trained on PT of target language only.
- Softmax # 2: Trained on DT of all other languages.
- **Both** softmax layers trained **simultaneously**.

PERs of HMM and DNN models trained on PTs of the target language. First element in parentheses is the PER of the dev set. Second element is the absolute improvement in PER of the test set over MAP HMM.

Lang	PER (%)		
	MAP HMM	DNN-1	DNN-2
SW	44.77 (50.97,0.0)	45.14 (47.83,-0.37)	43.03 (45.87,1.74)
HG	56.85 (57.69,0.0)	56.13 (57.21,0.72)	55.53 (56.08,1.32)
MD	59.23 (58.05,0.0)	54.95 (54.35,4.28)	53.70 (53.94,5.53)

Two softmax DNN (DNN-2): Why does it help?

- Reduce entropy of posterior probability outputs at the softmax layer.
  - Training DTs and PTs on two separate softmax layers ensure the weights in the target softmax layer are not corrupted.
- Fix non-linear transformation at the hidden layers.
  - Non-linear transformation in the shared hidden layers become weak if we train the DNN using PT labels (noisy) only. We fix that using DTs.

Three softmax DNN (DNN-3)

- Step 1: Use DNN-2 to decode unsupervised data and generate soft labels. Motivation: Make use of the unsupervised data as well.
- Step 2: Train DNN using 3 softmax layers this time.
  - Softmax # 1: Trained on PT of target language only. (PT from crowd workers)
  - Softmax # 2: Trained on DT of all other languages.
  - Softmax # 3: Trained on PT of the target language. (PT from the ASR system in Step 1)

Three softmax DNN (DNN-3) PER

PERs of HMM and DNN models trained on PTs of the target language. First element in parentheses is the PER of the dev set. Second element is the absolute improvement in PER of the test set over MAP HMM.

Lang	PER (%)			
	MAP HMM	DNN-1	DNN-2	DNN-3
SW	44.77 (50.97,0.0)	45.14 (47.83,-0.37)	43.03 (45.87,1.74)	43.50 (45.95, 1.27)
HG	56.85 (57.69,0.0)	56.13 (57.21,0.72)	55.53 (56.08,1.32)	55.69 (56.85, 1.16)
MD	59.23 (58.05,0.0)	54.95 (54.35,4.28)	53.70 (53.94,5.53)	53.05 (53.59, 6.18)

Summarizing PER Improvements

PER Improvements

Lang	PER (%)		
	MULTI	Best PT Adapted DNN	MONO
SW	61.17 (63.12)	43.03 (45.87)	34.18 (39.49)
HG	63.25 (63.65)	55.53 (56.08)	35.62 (37.32)
MD	64.68 (63.84)	53.05 (53.59)	28.26 (25.16)

Recall:

- MULTI: Multilingual system, not trained using target language data. Worst case PER (upper bound).
- MONO: Monolingual system, trained using DT in target language. Best case PER (lower bound).
- Best PT Adapted DNN: Either DNN-2 or DNN-3. Trained using PT in target language. PER in between the bounds.

Conclusions

1. Crowdsourced PTs are not as useful as DTs. The gap between MONO and DNN-2/DNN-3 is still large.
2. Proposed DNN-2/DNN-3 systems are able to close between 28% and 67% (relative) of the gap between MULTI and MONO systems. Thus, PTs are between one and two thirds as useful as DTs.
3. Crowdsourced PTs are more useful than PTs generated from an ASR system.

Thank You