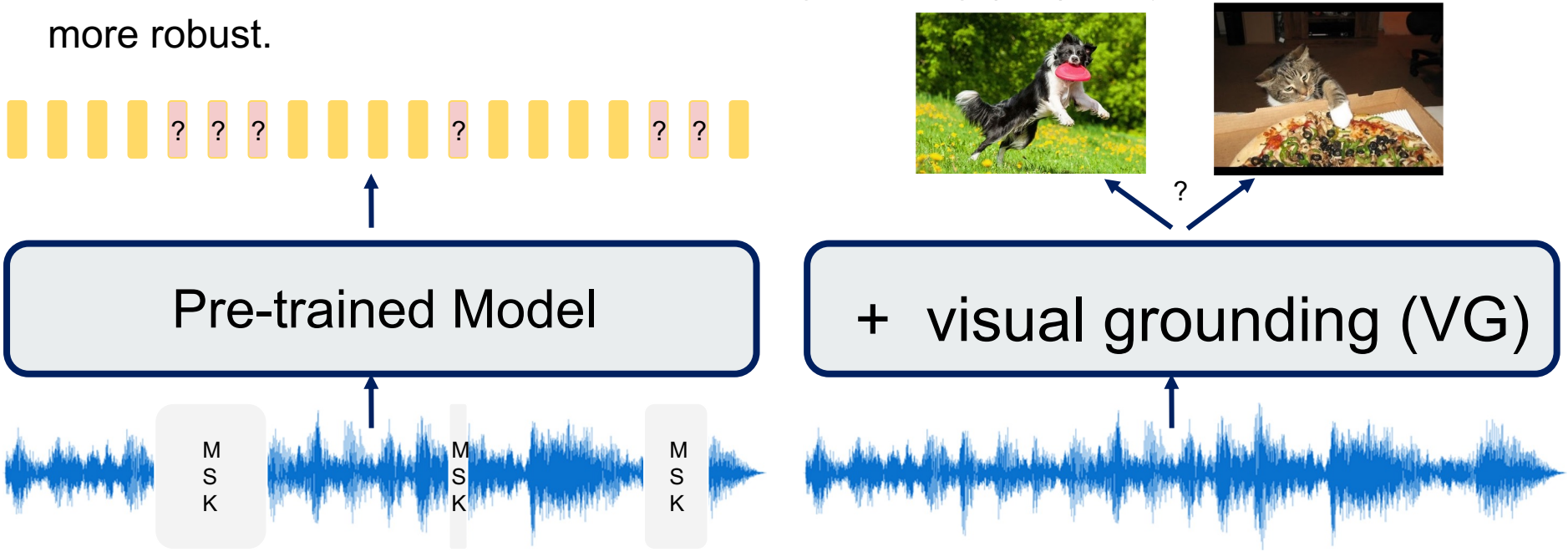
Visually-Enhanced Self-Supervised Speech Models

Team Lead: David Harwath (UT)
Student Team Members: Layne Berry (UT), Heng-Jui Chang (NTU), Po-Heng Chen (NTU), Vanya Cohen (UT), Wei-Yang Lin (NTU), Jason Peng (UT), Ian Shih (NTU), Xuan-Fu Wang (NTU)

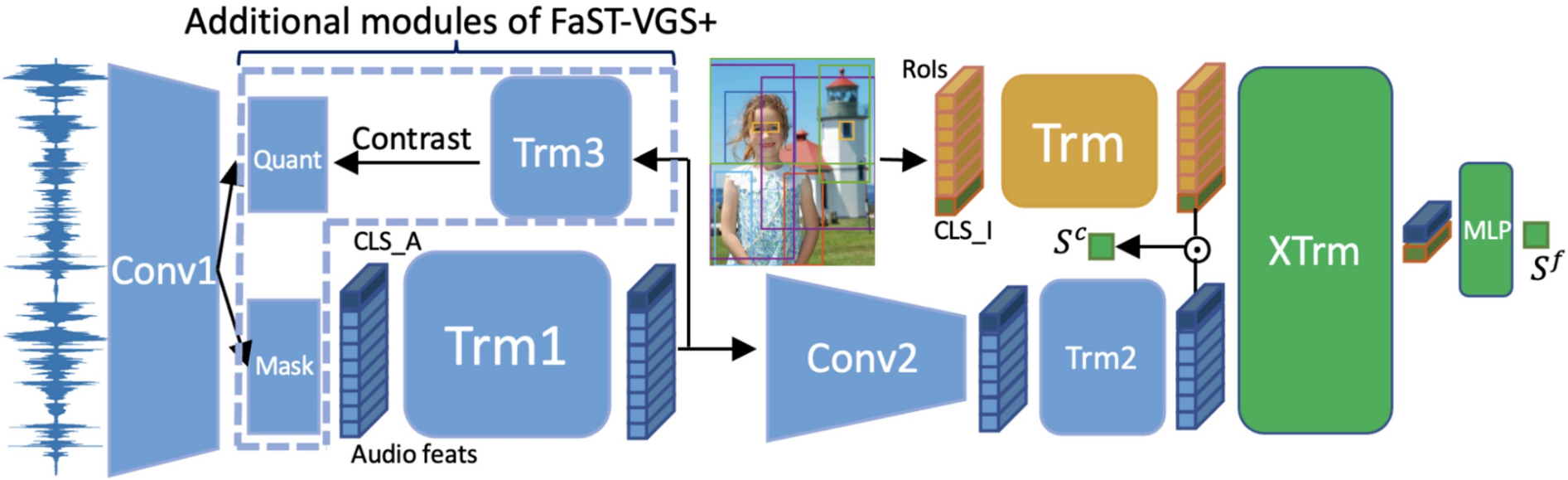
How about adding visual information?

The currently most successful pre-trained models for speech only leverage speech-only information.

However, other modalities, such as visual grounding (VG), may make the models more robust.



Current VG SotA: FaST-VGS



Speech-Image Retrieval

	Places Audio (test-seen)						Places Audio (test-unseen)					
	Speech → Image			Image → Speech			Speech → Image			Image → Speech		
Model	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
ResDAVEnet [4]	35.2	67.5	78.0	30.4	63.1	74.1	38.3	68.5	78.8	31.2	65.0	75.4
MILAN [23]	58.4	84.6	90.6	53.8	83.4	90.1	62.1	86.0	90.5	58.2	85.8	90.9
FaST-VGS$_{CO}$	60.0	86.1	92.3	60.2	85.1	**92.2**	62.8	88.4	92.9	62.3	89.0	93.2
FaST-VGS$_{CTF}$	**64.0**	**88.3**	**93.7**	**64.2**	**87.9**	**92.2**	**69.6**	**90.3**	**94.3**	**66.0**	**90.4**	**94.1**

Model	Speech → Image			Image → Speech		
	R@1	R@5	R@10	R@1	R@5	R@10
	Flickr8k (test 1k)					
[9]	5.5	16.3	25.3	-	-	-
[18]	9.6	-	-	-	-	-
[20]†	-	-	29.6	-	-	-
[6]	12.7	34.9	48.5	16.0	42.8	56.1
[40]†	21.8	49.9	63.1	-	-	-
FaST-VGS$_{CO}$	26.6	56.4	68.8	36.2	66.1	76.5
FaST-VGS$_{CTF}$	**29.3**	**58.6**	**71.0**	**37.9**	**68.5**	**79.9**
[22]	13.9	36.8	49.5	18.2	43.5	55.8
MILAN [23]	33.2	62.7	73.9	49.6	79.2	87.5
FaST-VGS$^{*}_{CO}$	41.2	70.4	81.5	53.6	81.1	89.5
FaST-VGS$^{*}_{CTF}$	45.5	73.8	83.7	59.8	84.1	90.7
	SpokenCOCO (test 5k)					
ResDAVEnet [4]	17.3	41.9	55.0	22.0	50.6	65.2
FaST-VGS$_{CO}$	31.8	62.5	75.0	42.5	73.7	84.9
FaST-VGS$_{CTF}$	**35.9**	**66.3**	**77.9**	**48.8**	**78.2**	**87.0**
	Text → Image			Image → Text		
DIME [39]	40.2	70.7	81.4	56.1	83.2	91.1
DIME$_{ASR}$	37.6	66.3	76.9	54.8	82.4	90.4

ZeroSpeech 2021

		Phonetic ABX ↓				Lexical ↑		Syntactic ↑	Semantic (Further from 0)	
Model	Data	W-C	W-O	A-C	A-O	all	in vocab.		Syn.	Lib.
LSTM baseline	LS	3.43	4.81	4.31	7.92	60.55	66.22	52.89	7.35	2.38
BERT baseline	LS	3.43	4.84	4.17	7.59	67.66	75.55	56.13	6.25	4.35
(v. Niekerk et al. 2021)	LS	5.41	8.67	6.89	13.14	65.06	72.86	53.95	9.23	-1.14
(Maekaku et al. 2021)	LS	3.15	5.13	4.25	8.64	61.15	66.36	53.88	7.00	-1.47
(Chorowski et al. 2021)	LS	2.85	4.44	3.69	7.28	64.15	72.47	52.55	5.15	-0.85
VG baseline (l.b.)	LS+SC	8.39	10.66	10.59	15.03	52.86	54.93	53.02	9.71	0.16
VG baseline (h.b.)	LS+SC	5.36	7.35	6.71	11.92	67.20	74.85	54.53	9.99	-0.10
Kim et al.	LS+PS	6.50	9.95	9.04	15.44	51.36	51.91	50.43	8.23	16.76
FaST-VGS+	LS+SC	4.24	5.22	5.08	7.91	67.56	75.23	57.40	15.10	14.32

SUPERB

Method	#Params	Data	Speaker			Content					Semantics			ParaL
			SID	ASV	SD	PR	ASR (WER)		KS	QbE	IC	SF		ER
			Acc ↑	EER ↓	DER ↓	PER ↓	w/o ↓	w/LM ↓	Acc ↑	MTWV ↑	Acc ↑	F1 ↑	CER ↓	Acc ↑
FBANK	0	-	8.5E-4	9.56	10.05	82.01	23.18	15.21	8.63	0.0058	9.10	69.64	52.94	35.39
PASE+	7.83M	LS50	37.99	11.61	8.68	58.87	25.11	16.62	82.54	0.0072	29.82	62.14	60.17	57.86
APC	4.11M	LS360	60.42	8.56	10.53	41.98	21.28	14.74	91.01	0.0310	74.69	70.46	50.89	59.33
VQ-APC	4.63M	LS360	60.15	8.72	10.45	41.08	21.20	15.21	91.11	0.0251	74.48	68.53	52.91	59.66
NPC	19.38M	LS360	55.92	9.40	9.34	43.81	20.20	13.91	88.96	0.0246	69.44	72.79	48.44	59.08
Mockingjay	85.12M	LS360	32.29	11.66	10.54	70.19	22.82	15.48	83.67	6.6E-04	34.33	61.59	58.89	50.28
TERA	21.33M	LS360	57.57	15.89	9.96	49.17	18.17	12.16	89.48	0.0013	58.42	67.50	54.17	56.27
wav2vec	32.54M	LS960	56.56	7.99	9.9	31.58	15.86	11.00	95.59	0.0485	84.92	76.37	43.71	59.79
vq-wav2vec	34.15M	LS960	38.80	10.38	9.93	33.48	17.71	12.80	93.38	0.0410	85.68	77.68	41.54	58.24
wav2vec 2.0 Base	95.04M	LS960	75.18	6.02	6.08	5.74	6.43	4.79	96.23	0.0233	92.35	88.30	24.77	63.43
HuBERT Base	94.68M	LS960	81.42	5.11	5.88	5.41	6.42	4.97	96.30	0.0736	98.34	88.53	25.20	64.92
FaST-VGS	187.87M	LS960+SC742	41.49	6.54	6.50	16.30	13.46	9.51	96.85	0.0546	98.37	84.91	32.33	57.37
FaST-VGS+	217.23M	LS960+SC742	41.34	5.87	6.05	7.76	8.83	6.37	97.27	0.0562	98.97	88.15	27.12	60.96
modified CPC	1.84M	LL60k	39.63	12.86	10.38	42.54	20.18	13.53	91.88	0.0326	64.09	71.19	49.91	60.96
WavLM Base+	94.70M	Mix94k	86.84	4.26	4.07	4.07	5.64	—-	96.69	**0.0990**	**99.16**	89.73	21.54	67.98
wav2vec 2.0 Large	317.38M	LL60k	86.14	5.65	5.62	4.75	3.75	3.10	96.66	0.0489	95.28	87.11	27.31	65.64
HuBERT Large	316.61M	LL60k	90.33	5.98	5.75	3.53	3.62	2.94	95.29	0.0353	98.76	89.81	21.76	67.62
WavLM Large	316.62M	Mix94k	**95.25**	**4.04**	**3.47**	**3.09**	**3.51**	—-	**97.40**	0.0827	99.10	**92.25**	**17.61**	**70.03**

Research Directions

SpeechCLIP: Vanya Cohen, Ian Shih, Layne Berry, Xuan-Fu Wang, Heng-Jui Chang

Word discovery + Star Temporal Classification for Self-Training ASR: Jason Peng

Training w/o negative examples: Po-Heng Chen

Improved grounding/segmentation: Wei-Yang Lin, Heng-Jui Chang

Timeline

March through May: Explore different directions in parallel to find those that are most promising

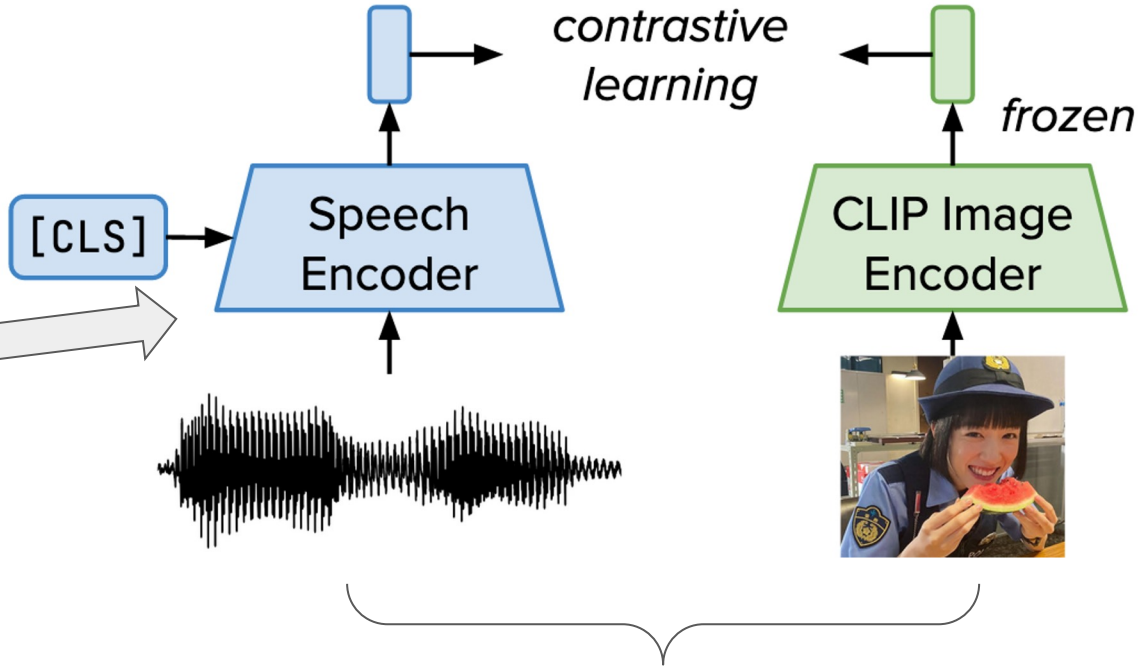
May: Consolidate around 1-2 most promising directions based on initial results

June through August (workshop): and finish experimental work

Parallel SpeechCLIP

Use learned representations from this model on downstream tasks (SUPERB, ZeroSpeech)

Also evaluate this model's ability to do speech-to-image retrieval vs. models trained from scratch



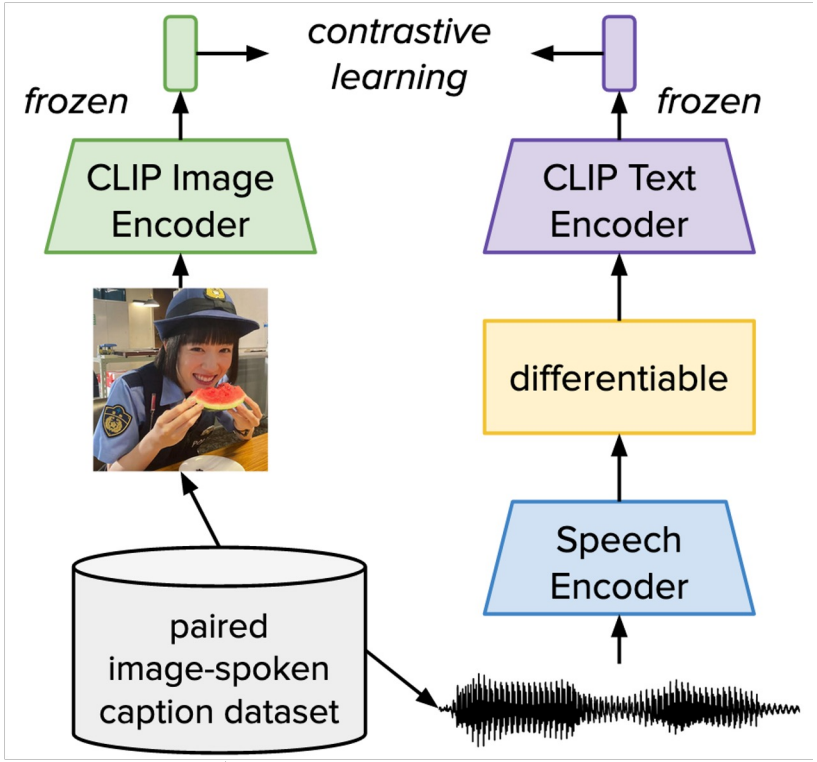
Training examples from SpokenCOCO, Places Audio, etc…

Cascaded SpeechCLIP

Again, we can use the learned representations from the speech encoder on downstream tasks like SUPERB, ZeroSpeech

This model also opens the door to some new possibilities:

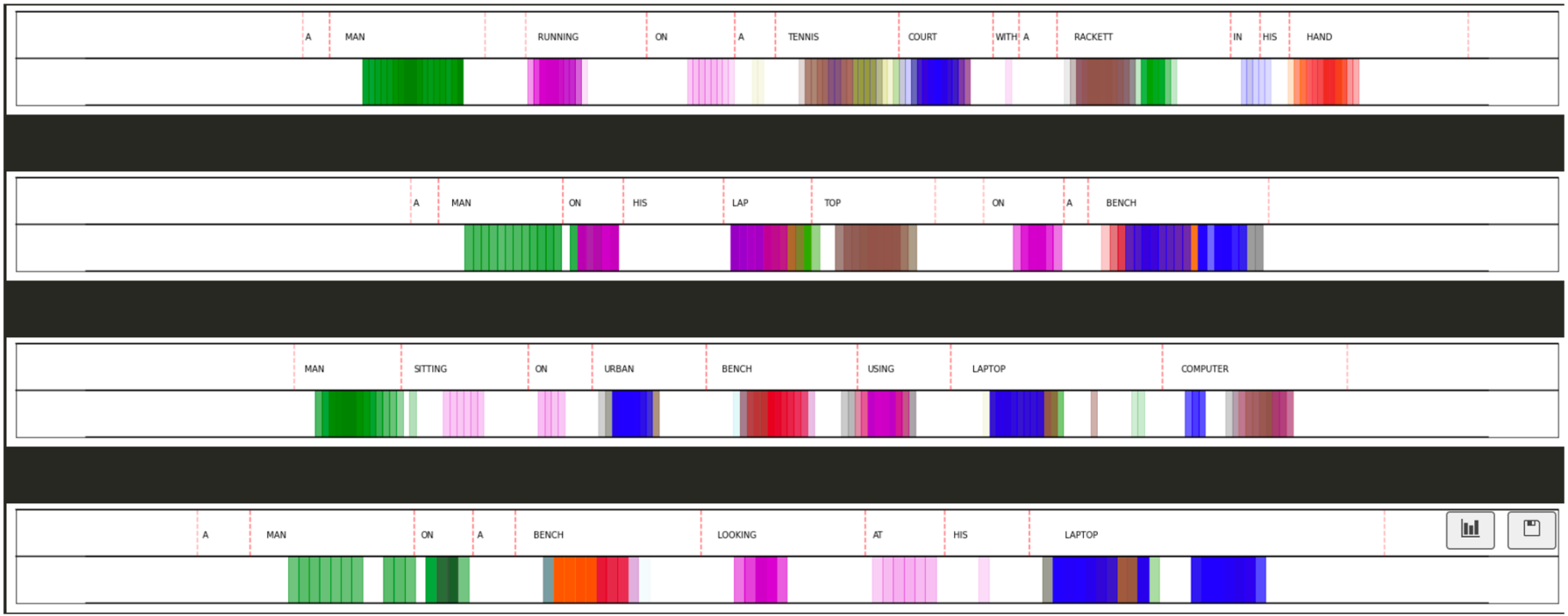
- Unsupervised speech recognition: Learn to map English speech to English text using the image as a guide
- Unsupervised speech-to-text translation: Use Hindi input speech, learn to map to English text using the image as a guide



This can be English (SpokenCOCO, Places Audio) or Hindi (Places Audio Hindi)

Efficient Self-Training for ASR (Jason Peng)

Our latest Transformer-based models for visually grounding speech learn to segment/discover words:



Efficient Self-Training for ASR (Jason Peng)

Rank	Code	Word	F1	Prec	Recall	Occ
2	74	trail	100.00	100.00	100.00	34
3	1045	chewing	100.00	100.00	100.00	18
4	3830	harbor	100.00	100.00	100.00	19
8	211	smart	100.00	100.00	100.00	22
11	3570	shirtless	100.00	100.00	100.00	12
13	1082	garage	100.00	100.00	100.00	19
16	3918	locomotive	100.00	100.00	100.00	20
24	2300	soup	100.00	100.00	100.00	25
31	1367	dirt	98.67	99.33	98.01	148
32	4052	christmas	98.51	97.06	100.00	33
33	2331	boxes	98.36	100.00	96.77	30
34	1931	hydrant	98.17	97.92	98.43	188
35	4077	public	98.00	98.00	98.00	49
36	1269	mouth	97.90	97.22	98.59	70
37	1116	polar	97.73	95.56	100.00	43
38	536	buildings	97.65	99.05	96.30	104
39	505	wine	97.51	96.08	98.99	98
40	2762	batter	97.48	95.08	100.00	58
			...			
180	3181	rural	90.32	93.33	87.50	14
181	1755	jeans	90.32	87.50	93.33	14
182	3492	carrots	90.10	86.67	93.81	91
183	3849	stack	90.00	81.82	100.00	18
186	2522	cakes	90.00	90.00	90.00	18
187	2264	apple	89.92	96.67	84.06	58
188	1089	distance	89.86	83.78	96.88	31
189	2276	team	89.80	84.62	95.65	22
190	1189	colorful	89.74	97.22	83.33	70
191	490	leash	89.66	81.25	100.00	13
192	1372	shopping	89.66	81.25	100.00	13
193	1972	eyes	89.55	90.91	88.24	30
194	1778	kid	89.52	97.92	82.46	47
195	653	tabby	89.47	94.44	85.00	17
196	2378	meal	89.43	82.09	98.21	55
197	934	commuter	89.36	80.77	100.00	21
198	1498	highway	89.36	80.77	100.00	21
199	1859	bottom	89.36	91.30	87.50	21
200	2856	screen	89.31	84.52	94.67	71

We can extract the discovered segments and cluster them (e.g. with K-means) to recover extremely pure word clusters that also have a very high coverage

Current models discover > 1000 different words on SpokenCOCO with high accuracy (greater than 0.5 F1)

Big idea: what if you had an untranscribed speech dataset and could cluster words this way, and then have a human annotator listen to just a few examples of each cluster and assign it a label?

You could:

1. Propagate the label to all instances of the cluster across the entire dataset
2. Use these partial transcriptions to train a full ASR system using self-training

Efficient Self-Training for ASR (Jason Peng)

Problem: self-training usually assumes we have a subset of training utterances that are *fully transcribed* and other utterances that are untranscribed. But we only have a bunch of *partially* transcribed utterances.

Solution:

Star Temporal Classification:
Sequence Classification with Partially Labeled Data

Vineel Pratap [1] Awni Hannun [2] Gabriel Synnaeve [1] Ronan Collobert [3]

Original Label: seen from an airplane the island looks like a big spider

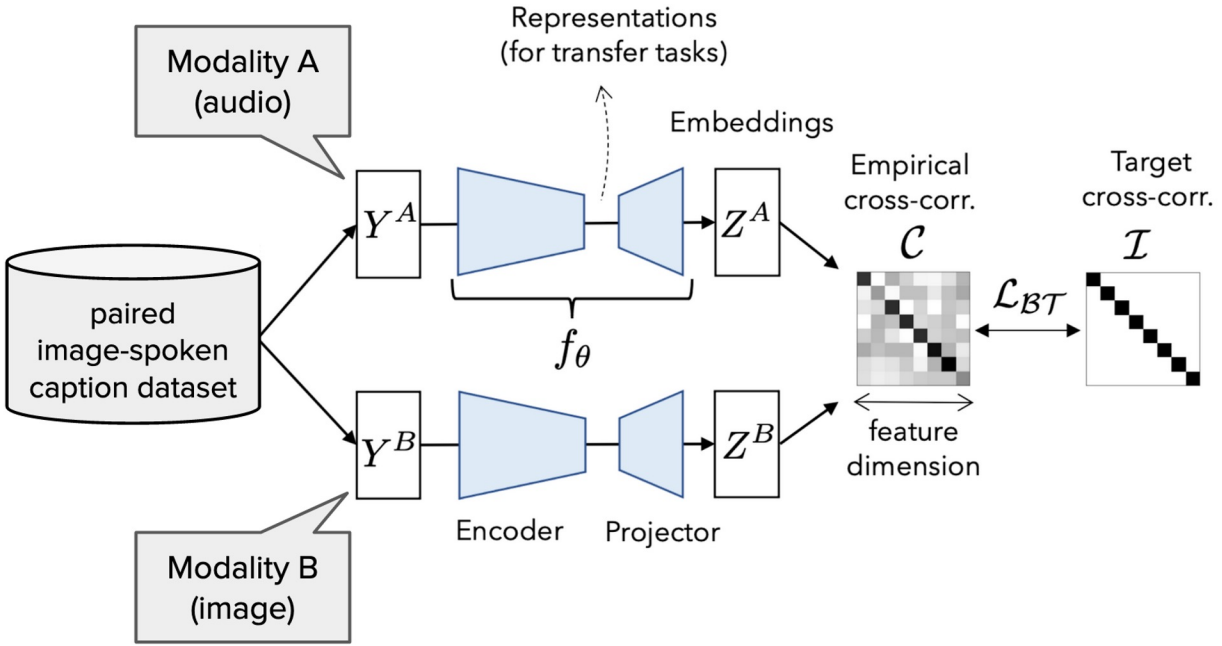
Partial Label: from airplane the a spider

Figure 1. An example of speech with a complete and a partial label.

Basic idea: modify CTC training objective to allow wildcards, e.g. training label sequence for above example becomes

```
* from * airplane * the * a * spider *
```

Training Without Negative Examples (Po-Heng Chen)

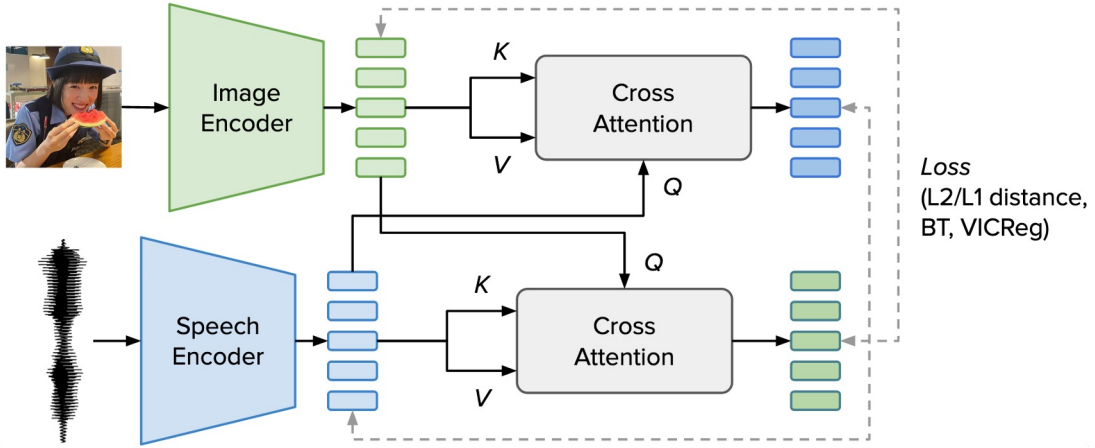


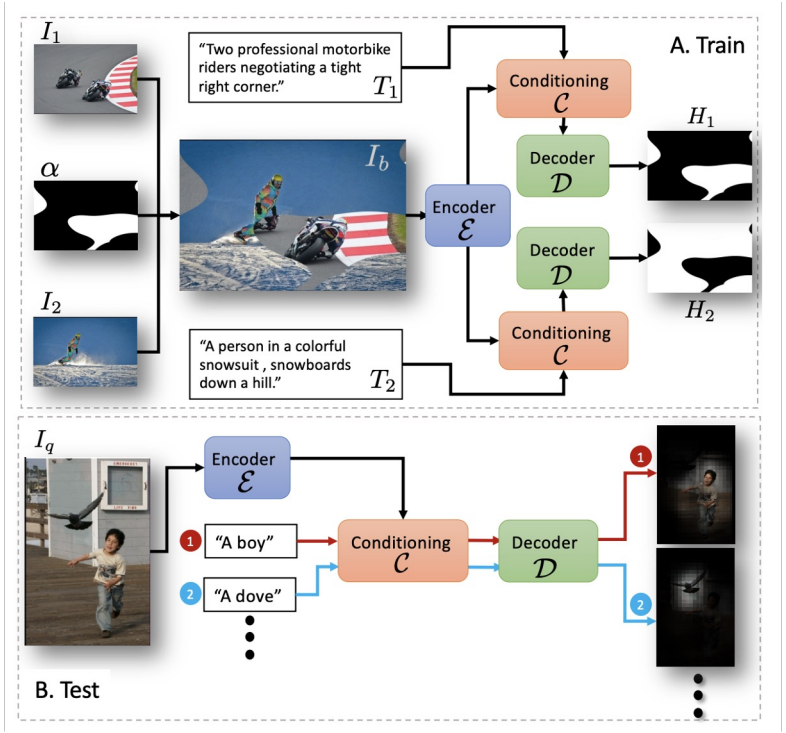
Use learned representations on downstream tasks (SUPERB, ZeroSpeech)

Also evaluate this model's ability to do speech-to-image retrieval vs. models trained from scratch

Compare against versions of the model that use negative example-based contrastive learning objectives

Improving Fine-Grained Cross-Modal Alignment (Heng-Jui Chang, Wei-Yang Lin)





Goal: force the cross-attention modules in image-speech grounding models to segment words/objects

Downstream tasks: word segmentation (COCO, ZeroSpeech), visual object detection on COCO