Keyword Spotting with Few-shot examples

Yuan Junming

2023/12/04

Outline

- Why choose Few-Shot Keyword Spotting?
- Some Few-Shot Keyword Spotting Methods

  Review some papers

- Experimental Progress

  Model Pretraining based Mix training

  Extend Temporal feature to reduce the FAR

- Future work

Why choose Few-Shot Keyword Spotting?

- Modern KWS models are typically trained on large datasets and restricted to a small vocabulary of keywords, limiting their transferability to a broad range of unseen keywords.

- Learning to recognize new keywords with just a few-shot examples is essential for personalizing keyword spotting (KWS) models to a user’s choice of keywords.

Some Few-Shot Keyword Spotting Methods

- Data augmentation based.
- Meta-learning & Few-shot learning based.
- Transfer from labeled data of other keywords (model pre-training)

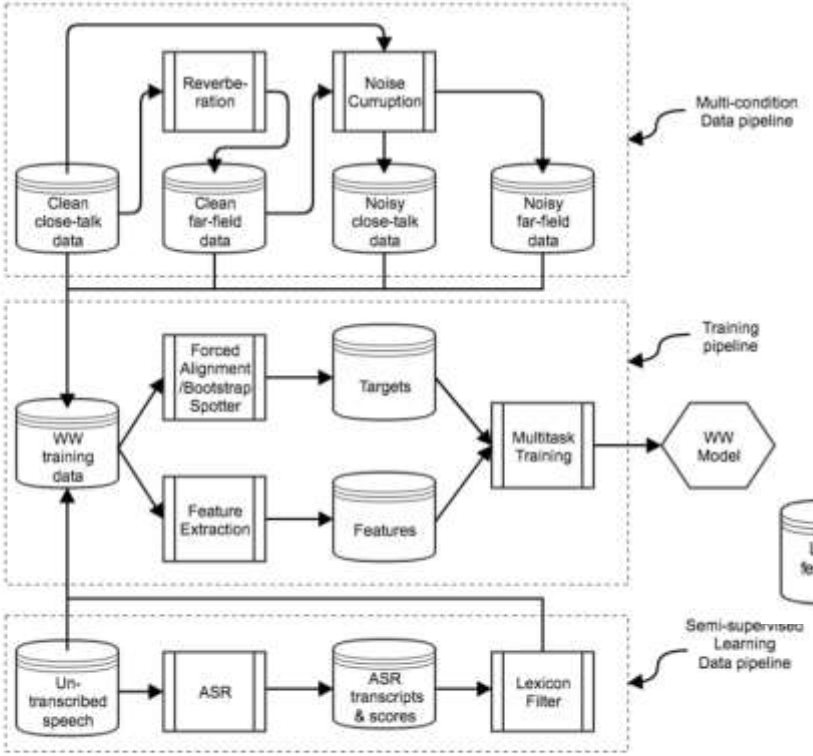
  Preparing large-scale KWS datasets using audios, transcription, and a forced aligner.

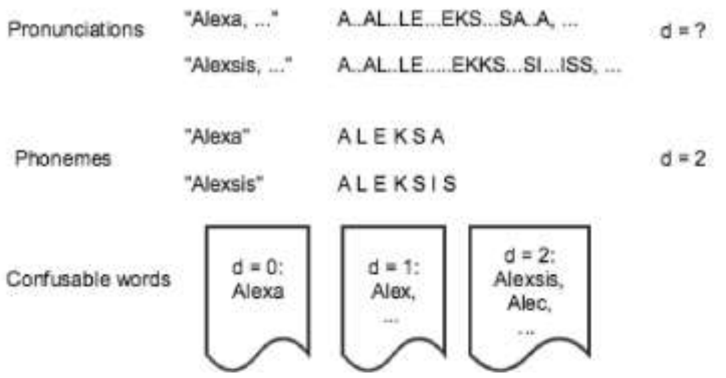
  Finetuning on the few keyword examples.
- Utilize unlabeled data.

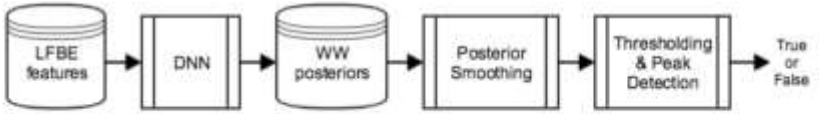
  Using self-supervised learning (SSL) method to learn feature extractors from unlabeled data.

TOWARDS DATA-EFFICIENT MODELING FOR WAKE WORD SPOTTING

*Alexa, Amazon.com Services LLC.

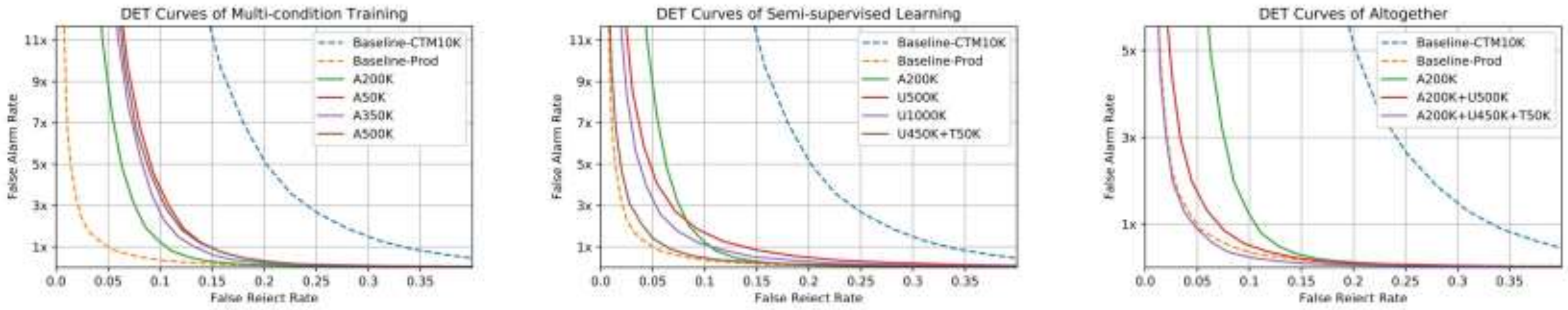






- **Motivation**
  - Propose data augmentation techniques such as the addition of reverberation and noise to simulate far-field speech.
- **Training architecture**
  - Multi-condition pipeline.
  - Semi-supervised learning pipeline.
  - Multi-task training pipeline.
- **Result**

size	CTM	CTM+R	CTM+N	CTM+RN
50K	10K	14K	14K	14K
200K	20K	60K	60K	60K
350K	35K	105K	105K	105K
500K	50K	150K	150K	150K



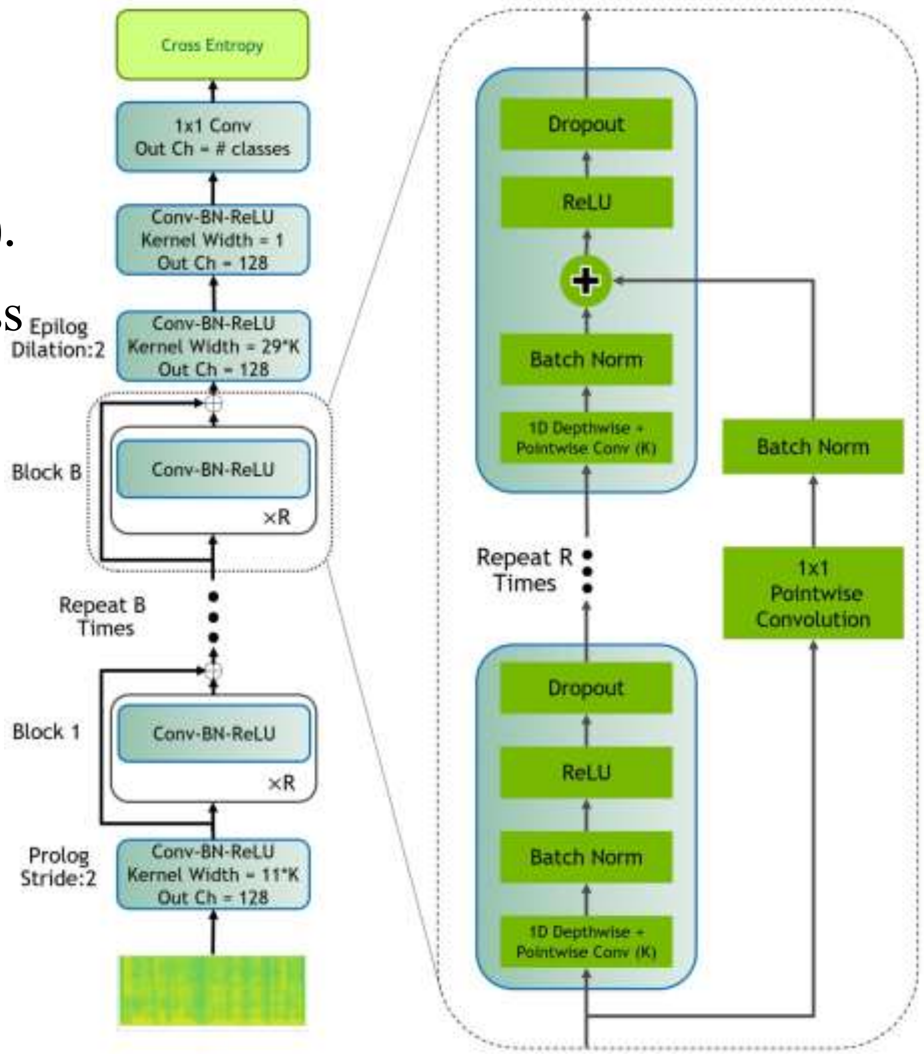
(a) (b) (c)

Gao Y, Mishchenko Y, Shah A, et al. Towards data-efficient modeling for wake word spotting[C]// (ICASSP 2020)

MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition

*NVIDIA, Santa Clara, USA.

- **Contribution**
  - Present a end-to-end neural network for KWS (MatchboxNet).
  - To improve the model's robustness by intensive data augmentation using an auxiliary noise dataset.
- **Model architecture**
  - A deep residual network
- **Training Methodology**
  - Applied time shift perturbations and SpecAugment
- **Result**



Model	# Parameters, K	Accuracy, %	Reference
ResNet-15	238	95.8 ± 0.351	[17]
DenseNet-BC-100	800	96.77	[32]
EdgeSpeechNet-A	107	96.80	[29]
MatchboxNet-3x1x64	77	97.21 ± 0.067	
MatchboxNet-3x2x64	93	97.48 ± 0.107	

Model	# Parameters, K	Accuracy, %	Reference
Attention RNN	202	94.30	[33]
Harmonic Tensor 2D-CNN	-	96.39	[30]
"Embedding + Head" Model	385	97.7	[31]
MatchboxNet-3x1x64	77	96.91 ± 0.101	
MatchboxNet-3x2x64	93	97.21 ± 0.072	
MatchboxNet-6x2x64	140	97.37 ± 0.110	

B	R	C	# Parameters, K	Accuracy,%
3	2	64	93	97.21
3	3	64	109	97.36
3	4	64	125	97.17
3	5	64	149	97.37
4	2	64	109	97.20
5	2	64	124	97.31
6	2	64	140	97.55
3	2	80	118	97.44
3	2	96	145	97.41
3	2	112	177	97.63

Majumdar S, Ginsburg B. Matchboxnet: 1d time-channel separable convolutional neural network architecture for speech commands recognition[J]. arXiv preprint arXiv:2004.08531, 2020.

MatchboxNet: 1D Time-Channel Separable Convolutional Neural Network Architecture for Speech Commands Recognition

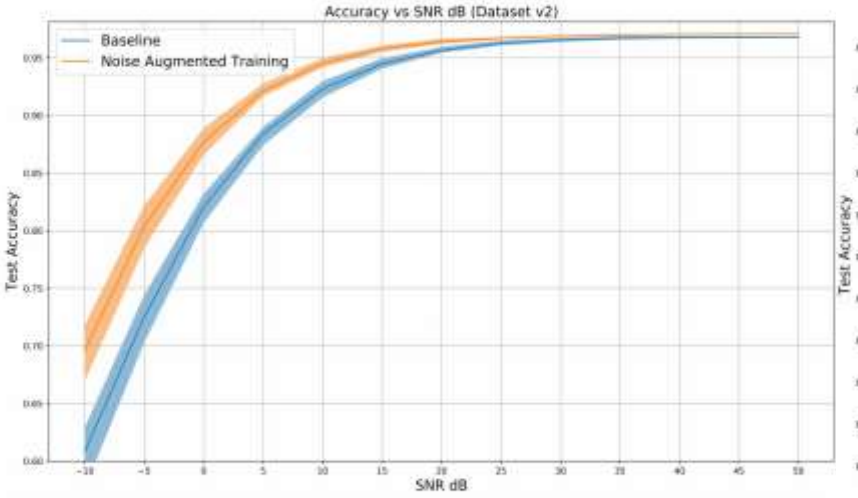
*NVIDIA, Santa Clara, USA.

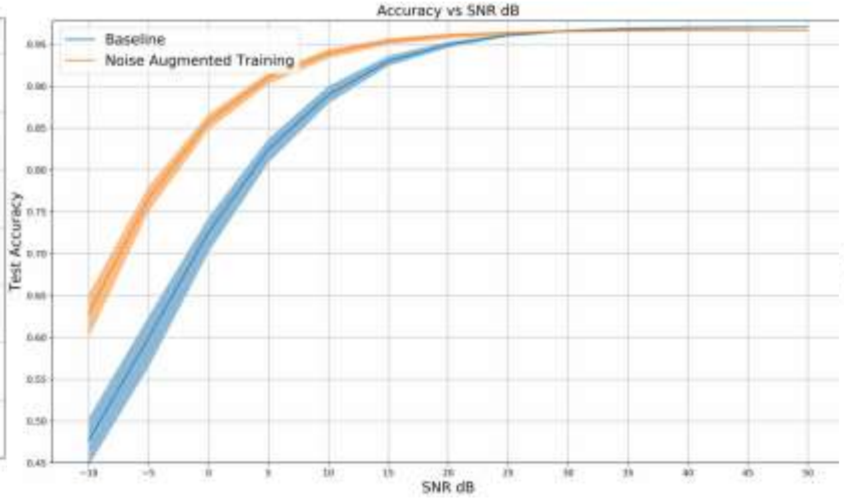
- **Result**

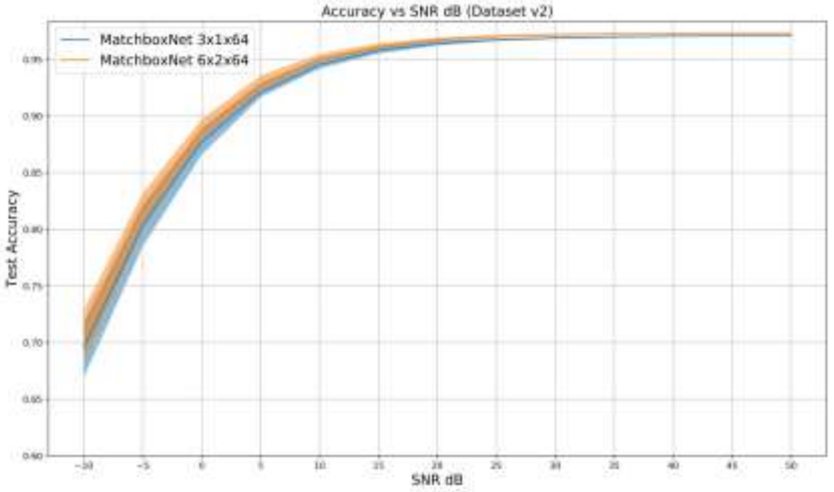
Model	Augmentation	Accuracy, %
MatchboxNet 3x1x64	basic	96.91 ± 0.101
MatchboxNet 3x1x64	+ background speech and noise	97.05 ± 0.099

Model	Dataset	# Parameters	Accuracy, %
MatchboxNet-3x1x64	v1	77K	96.88 ± 0.073
MatchboxNet-3x1x64	v2	77K	96.97 ± 0.071

Model	SNR (in dB)						
	-10	0	10	20	30	40	50
3x1x64	69.62	87.21	94.53	96.40	96.89	97.05	97.09
6x2x64	71.02	88.81	95.04	96.74	97.16	97.29	97.33







Majumdar S, Ginsburg B. Matchboxnet: 1d time-channel separable convolutional neural network architecture for speech commands recognition[J]. arXiv preprint arXiv:2004.08531, 2020.

An Investigation of Few-Shot Learning in Spoken Term Classification

*City University of Hong Kong. *Southern University of Science and Technology, Shenzhen, China. *Huawei Noah's Ark Lab. *The Hong Kong Polytechnic University.

- **Motivation**
  - Investigate the feasibility of applying few-shot learning algorithms to a speech task.
  - Investigate the performance of Model-Agnostic Meta-Learning (MAML).

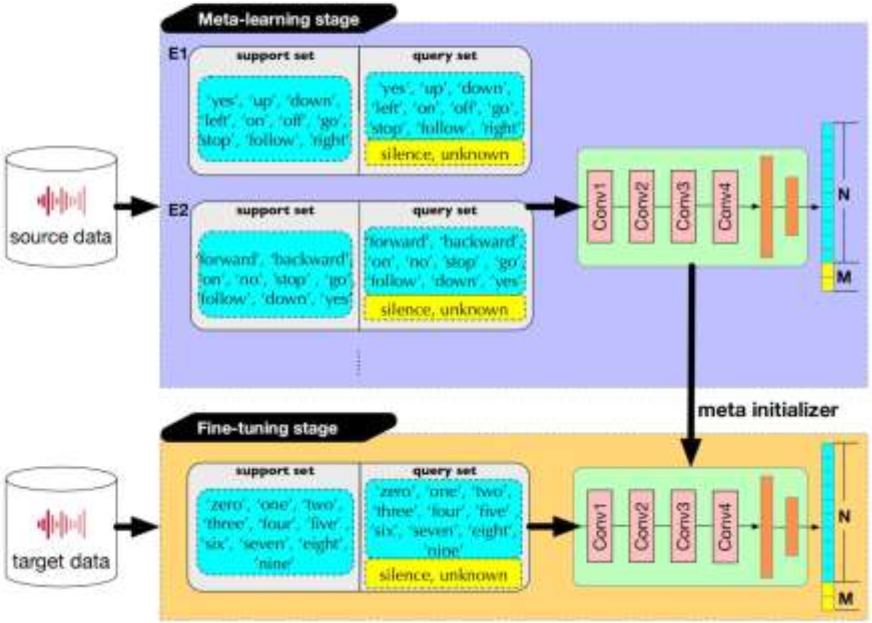
- **Methods**
  - Define a N+M-way problem where N and M are the number of new classes and fixed classes respectively.
  - Propose a modification to the MAML algorithm to solve the problem.

Chen Y, Ko T, Shang L, et al. An investigation of few-shot learning in spoken term classification[J]. arXiv preprint arXiv:1812.10233, 2018.

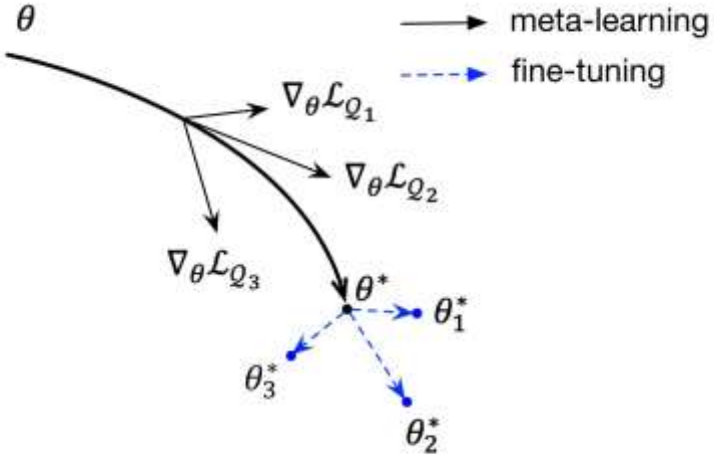
An Investigation of Few-Shot Learning in Spoken Term Classification

*City University of Hong Kong. *Southern University of Science and Technology, Shenzhen, China. *Huawei Noah's Ark Lab. *The Hong Kong Polytechnic University.

- **Framework**



- **Methodology**
  - fix the output positions of the fixed classes in the neural network classifier.
  - The fixed classes occur in every meta-task Ti in the meta-learning stage.
  - The adaptation of fixed classes is not needed in the fine-tuning stage



Algorithm 1 extended-MAML approach for few-shot spoken term classification

Require: $p(\mathcal{T})$: distribution over tasks
Require: $\mathcal{X}$: training keywords set
Require: $\mathcal{S}_{il}$: silence class set, $\mathcal{U}_{nk}$: unknown class set
Require: $\mathcal{S}_i$: support set, $\mathcal{Q}_i$: query set
Require: α, β: learning rates
1: Randomly initialize base model parameters $\boldsymbol{\theta}$
2: **while** not done **do**
3: Sample a batch of meta-tasks $\mathcal{T}_i \sim p(\mathcal{T})$
4: **for all** $\mathcal{T}_i$ **do**
5: Sample a support set $\mathcal{S}_i$ from $\mathcal{X}$
6: Compute the gradient $\nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$ using $\mathcal{S}_i$ and $\mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$ in Equation (1)
7: Update base model parameters with gradient descent: $\boldsymbol{\theta}'_i = \boldsymbol{\theta} - \alpha \nabla_{\boldsymbol{\theta}} \mathcal{L}_{\mathcal{S}_i}(f_{\boldsymbol{\theta}})$ ▷ step 6 and step 7 can be repeated for several times
8: Sample a query set $\mathcal{Q}_i$ from the union $\{\mathcal{X}, \mathcal{S}_{il}, \mathcal{U}_{nk}\}$ ▷ selected keywords from $\mathcal{X}$ in $\mathcal{Q}_i$ and $\mathcal{S}_i$ within $\mathcal{T}_i$ are the same
9: Compute the loss $\mathcal{L}_{\mathcal{Q}_i}(f_{\theta'_i})$ using $\mathcal{Q}_i$ and the updated model $f_{\theta'}$
10: **end for**
11: Update parameters $\boldsymbol{\theta}$ using each $\mathcal{Q}_i$ and $\mathcal{L}_{\mathcal{Q}_i}(f_{\theta'})$: $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} - \beta \nabla_{\boldsymbol{\theta}} \sum_i \mathcal{L}_{\mathcal{Q}_i}(f_{\theta'_i})$
12: **end while**

Chen Y, Ko T, Shang L, et al. An investigation of few-shot learning in spoken term classification[J]. arXiv preprint arXiv:1812.10233, 2018.

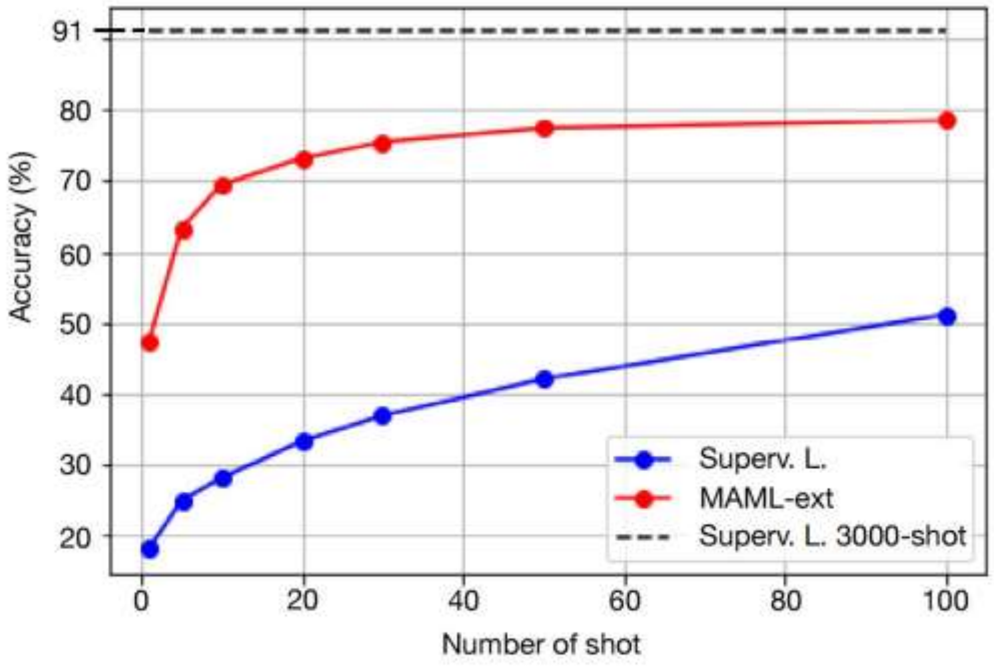
An Investigation of Few-Shot Learning in Spoken Term Classification

*City University of Hong Kong. *Southern University of Science and Technology, Shenzhen, China. *Huawei Noah's Ark Lab. *The Hong Kong Polytechnic University.

- **Result**

Methods	1-shot	5-shot	10-shot
Superv. L.	18.14 ± 0.44	24.83 ± 0.38	28.07 ± 0.34
MAML-ori	44.60 ± 0.98	60.88 ± 0.58	65.18 ± 0.62
MAML-ext	$\mathbf{47.42 \pm 0.96}$	$\mathbf{63.22 \pm 0.71}$	$\mathbf{69.48 \pm 0.47}$

Methods	1-shot	5-shot	10-shot
Superv. L.	17.03 ± 0.48	22.42 ± 0.33	25.6 ± 0.26
MAML-ori	33.35 ± 0.80	50.31 ± 0.50	57.34 ± 0.41
MAML-ext	$\mathbf{39.54 \pm 0.62}$	$\mathbf{52.20 \pm 0.51}$	$\mathbf{59.36 \pm 0.39}$



Chen Y, Ko T, Shang L, et al. An investigation of few-shot learning in spoken term classification[J]. arXiv preprint arXiv:1812.10233, 2018.

Few-Shot Keyword Spotting With Prototypical Networks

*The University of North Carolina at Charlotte.

- **Motivation**
  - There is a growing need for KWS systems

    (1) to recognize custom or new keywords on-device.

    (2) to quickly adapt from a small number of user samples
- **Methods**
  - Propose a few-shot KWS system(FS-KWS) with metric learning.
  - Propose a temporally dilated CNN architecture as a better embedding function for FS-KWS.
  - Release a FS-KWS dataset synthesized from Google's Speech command dataset.

Parnami A, Lee M. Few-shot keyword spotting with prototypical networks[C]//2022 7th International Conference on Machine Learning Technologies (ICMLT). 2022: 277-283.

Few-Shot Keyword Spotting With Prototypical Networks

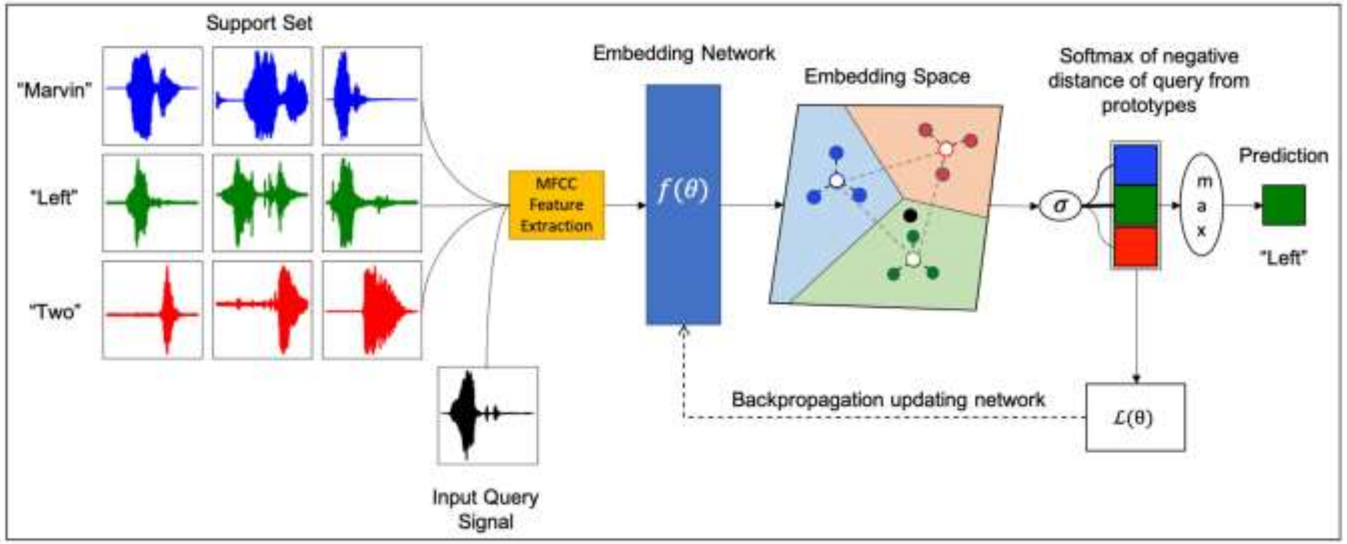
*The University of North Carolina at Charlotte.

- **Pipeline**

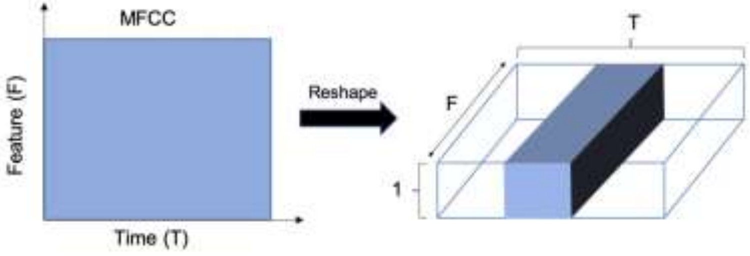
$$p_c = \frac{1}{|S_e^c|} \sum_{(s_i, y_i) \in S_e^c} f(s_i),$$

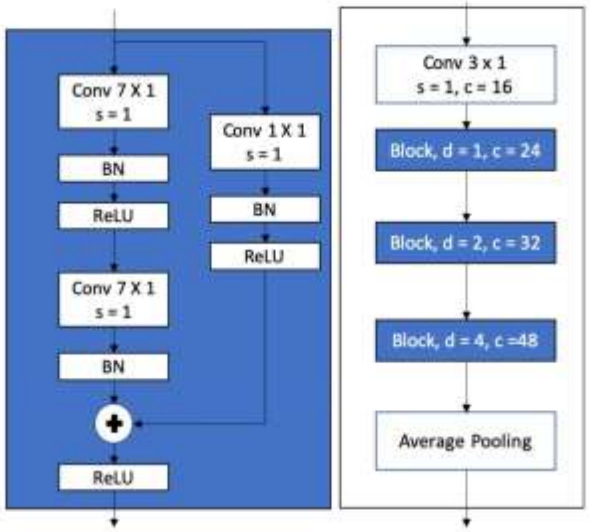
$$P(y = c \mid q_t, S_e, \theta) = \frac{exp(-d(f(q_t), p_c))}{\sum_n exp(-d(f(q_t), p_n))},$$

$$L(\theta) = -\sum_{t=1}^{|Q_e|} \log P_\theta(y_t \mid q_t, S_e),$$



- **Model architecture**
  - Audio Feature Extraction
  - Embedding Network





(a) *Block* (b) *TD-ResNet7*

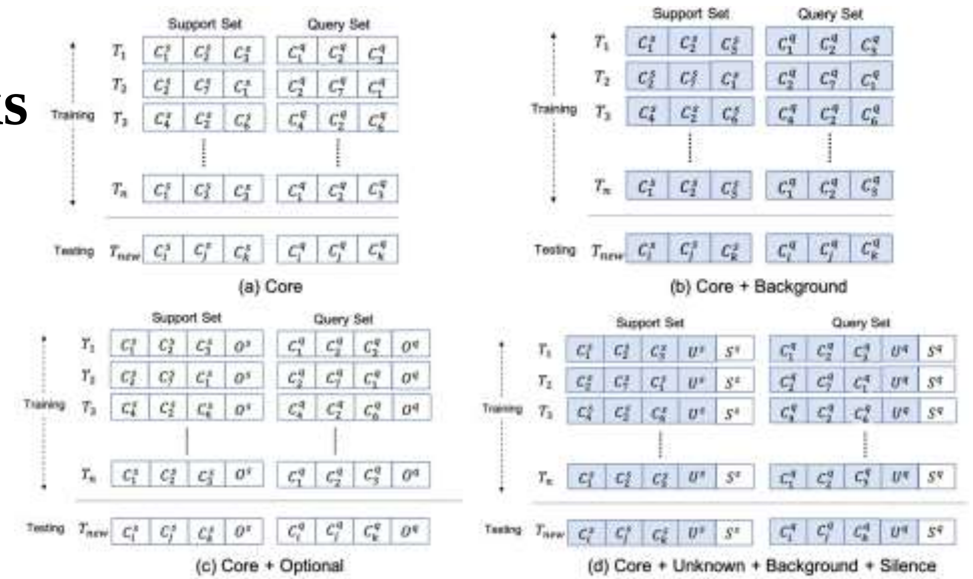
Parnami A, Lee M. Few-shot keyword spotting with prototypical networks[C]//2022 7th International Conference on Machine Learning Technologies (ICMLT). 2022: 277-283.

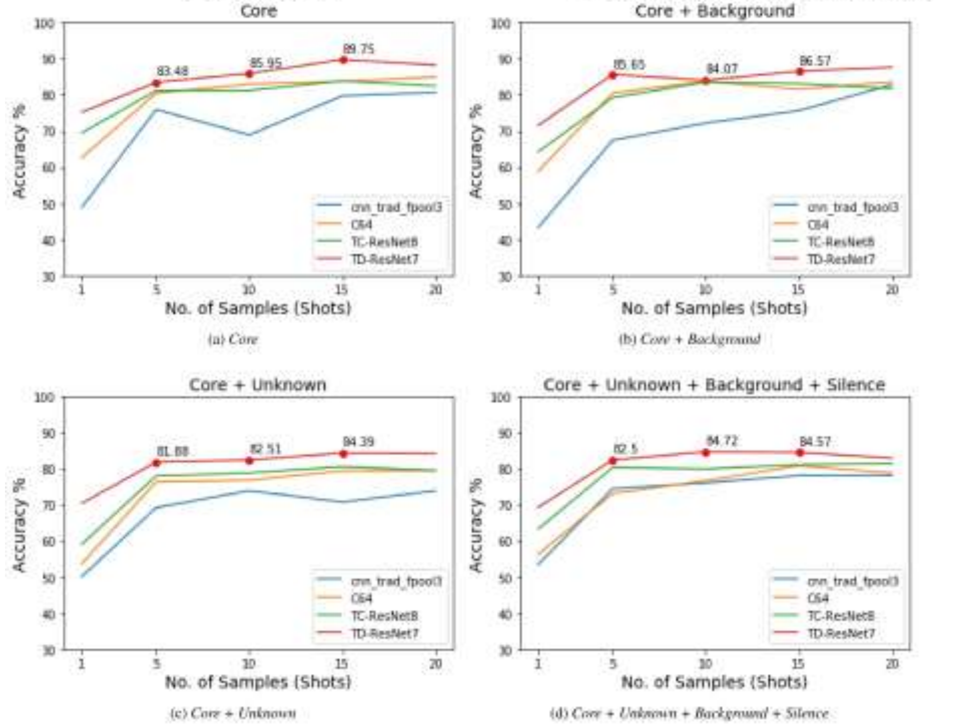
Few-Shot Keyword Spotting With Prototypical Networks

*The University of North Carolina at Charlotte.

- **Result**

Case	Embedding Network	2-way Acc.		4-way Acc.	
		1-shot	5-shot	1-shot	5-shot
core	cnn_trad_fpool3	69.23 ± 0.03	87.07 ± 0.02	48.83 ± 0.02	75.93 ± 0.01
	C64	77.20 ± 0.03	89.97 ± 0.02	62.63 ± 0.02	80.48 ± 0.01
	TC-ResNet8	82.70 ± 0.03	89.00 ± 0.02	69.47 ± 0.02	81.20 ± 0.01
	TD-ResNet7 (ours)	**85.43 ± 0.03**	**94.10 ± 0.01**	**75.22 ± 0.02**	**83.48 ± 0.02**
core + background	cnn_trad_fpool3	69.53 ± 0.04	86.8 ± 0.02	43.3 ± 0.02	67.42 ± 0.01
	C64	78.30 ± 0.03	90.03 ± 0.02	58.83 ± 0.02	80.52 ± 0.01
	TC-ResNet8	77.40 ± 0.03	**91.40 ± 0.02**	64.23 ± 0.02	79.25 ± 0.01
	TD-ResNet7 (ours)	**82.23 ± 0.03**	91.00 ± 0.02	**71.58 ± 0.02**	**85.65 ± 0.01**
core + unknown	cnn_trad_fpool3	58.33 ± 0.03	78.36 ± 0.02	50.15 ± 0.02	69.25 ± 0.01
	C64	63.42 ± 0.03	78.47 ± 0.02	53.69 ± 0.02	76.43 ± 0.01
	TC-ResNet8	68.84 ± 0.03	80.49 ± 0.02	59.08 ± 0.02	78.07 ± 0.01
	TD-ResNet7 (ours)	**77.24 ± 0.02**	**87.22 ± 0.01**	**70.45 ± 0.02**	**81.88 ± 0.01**
core + unknown + background + silence	cnn_trad_fpool3	67.43 ± 0.02	82.32 ± 0.01	53.51 ± 0.02	74.54 ± 0.01
	C64	65.83 ± 0.02	81.15 ± 0.01	56.38 ± 0.01	73.20 ± 0.01
	TC-ResNet8	78.63 ± 0.02	85.98 ± 0.01	63.37 ± 0.02	80.39 ± 0.01
	TD-ResNet7 (ours)	**82.77 ± 0.02**	**89.45 ± 0.01**	**69.34 ± 0.01**	**82.50 ± 0.01**





Parnami A, Lee M. Few-shot keyword spotting with prototypical networks[C]//2022 7th International Conference on Machine Learning Technologies (ICMLT). 2022: 277-283.

ON THE EFFICIENCY OF INTEGRATING SELF-SUPERVISED LEARNING AND META-LEARNING FOR USER-DEFINED FEW-SHOT KEYWORD SPOTTING

*Graduate Institute of Communication Engineering, National Taiwan University. *intelliGo Technology inc.

- **Motivation**
  - Previous works about User-defined keyword spotting try to incorporate self-supervised learning models or apply meta-learning algorithms.
  - It is unclear whether self-supervised learning and meta-learning are complementary and which combination of the two types of approaches is most effective for few-shot keyword discovery.

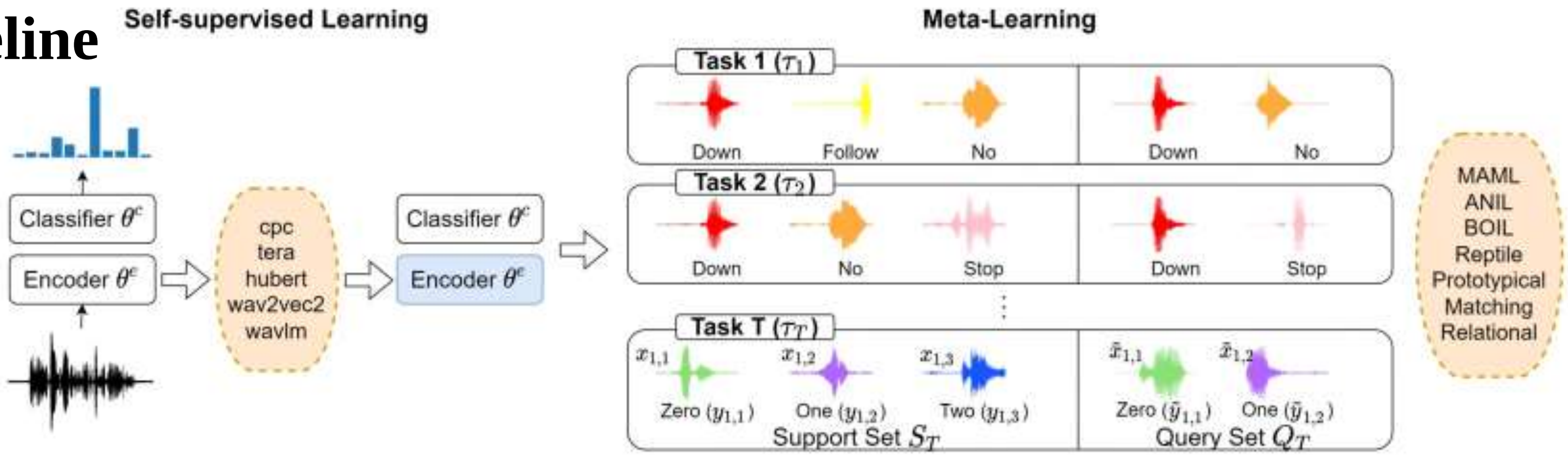
- **Methods**
  - Compare 5 widely used SSL models to answer which pre-trained model is the best for few-shot KWS.
  - Training the SSL models by 7 meta-learning algorithms to shed light on the effectiveness of combining the pre-training and meta-learning approaches.

Kao W T, Wu Y K, Chen C P, et al. On the efficiency of integrating self-supervised learning and meta-learning for user-defined few-shot keyword spotting[C]//2022 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2023: 414-421.

ON THE EFFICIENCY OF INTEGRATING SELF-SUPERVISED LEARNING AND META-LEARNING FOR USER-DEFINED FEW-SHOT KEYWORD SPOTTING

*Graduate Institute of Communication Engineering, National Taiwan University. *intelliGo Technology inc.

- **Pipeline**

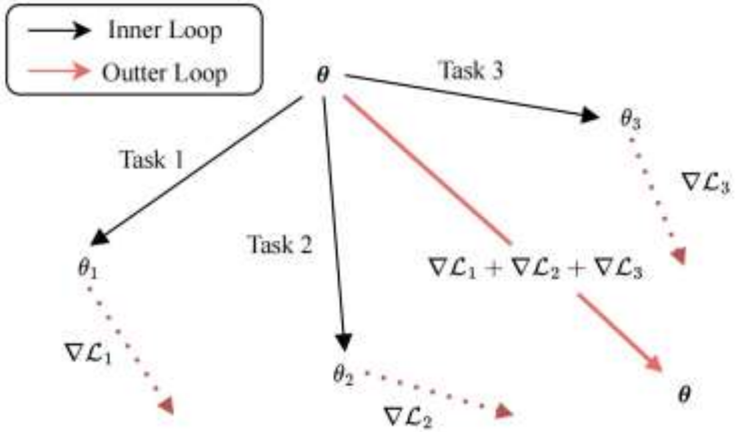


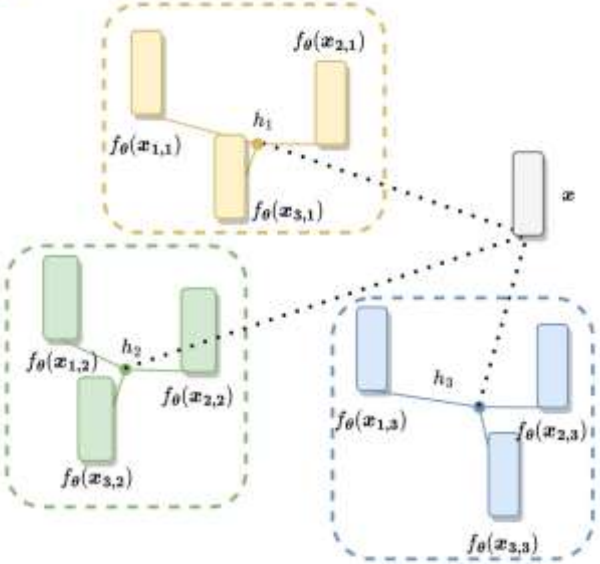
- **Meta-learning methods**
  - Optimization-based methods

    MAML, ANIL, BOIL, Reptile

  - Metric-based methods

    Prototypical network, Relational network, Matching network





Kao W T, Wu Y K, Chen C P, et al. On the efficiency of integrating self-supervised learning and meta-learning for user-defined few-shot keyword spotting[C]//2022 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2023: 414-421.

ON THE EFFICIENCY OF INTEGRATING SELF-SUPERVISED LEARNING AND META-LEARNING FOR USER-DEFINED FEW-SHOT KEYWORD SPOTTING

*Graduate Institute of Communication Engineering, National Taiwan University. *intelliGo Technology inc.

- **Result**

	SSL	MAML	ANIL	BOIL	Reptile	Prototypical	Matching	Relational	Trans-1	Trans-2
1-shot fine-tune	CPC	31.64	46.18	21.08	27.79	46.40	**46.98**	40.81	8.58	41.12
	TERA	44.66	39.93	43.97	37.84	48.12	**53.62**	42.16	44.88	
	HuBERT	50.00	63.13	38.53	53.78	67.99	**70.39**	49.34	63.33	
	Wav2Vec2	53.10	56.60	53.47	45.10	63.39	64.82	38.97	**65.71**	
	WavLM	39.12	53.88	46.34	38.81	69.90	**76.16**	42.83	58.26	
1-shot fix-encoder	CPC	33.97	-	-	23.48	39.73	41.63	35.71	**47.69**	56.58
	TERA	41.55	-	-	27.90	43.00	**48.18**	37.91	45.45	
	HuBERT	61.43	-	-	47.34	70.03	**79.30**	64.18	66.37	
	Wav2Vec2	57.41	-	-	35.04	56.69	**71.07**	57.99	66.5	
	WavLM	63.84	-	-	33.75	55.51	**75.27**	64.12	59.61	
5-shot fine-tune	CPC	32.02	58.49	21.68	52.05	**67.90**	64.55	59.39	9.06	79.95
	TERA	52.89	68.39	69.92	69.59	**75.40**	73.93	58.15	66.76	
	HuBERT	65.26	83.18	79.85	83.95	85.88	**88.98**	56.21	84.93	
	Wav2Vec2	60.58	78.76	70.84	82.45	80.49	**86.47**	52.89	84.82	
	WavLM	80.72	82.26	82.35	81.24	78.51	**87.30**	58.35	81.52	
5-shot fix-encoder	CPC	30.88	-	-	35.60	56.98	**58.32**	51.61	49.62	78.42
	TERA	45.56	-	-	44.67	60.55	62.71	50.93	**66.6**	
	HuBERT	70.80	-	-	38.02	85.84	**90.86**	73.60	85.03	
	Wav2Vec2	54.53	-	-	53.95	82.68	**85.52**	76.00	84.88	
	WavLM	70.24	-	-	49.02	83.06	**86.39**	67.75	81.16	

	ANIL	Matching	Trans-1
1-shot fine-tune	6.82	6.24	12.99
5-shot fine-tune	3.23	2.58	3.92

	Prototypical	Matching	Relational
1-shot HuBERT	67.99	70.39	49.34
1-shot scratch	38.95	40.80	41.23
5-shot HuBERT	85.88	88.98	56.21
5-shot scratch	61.56	60.26	50.91

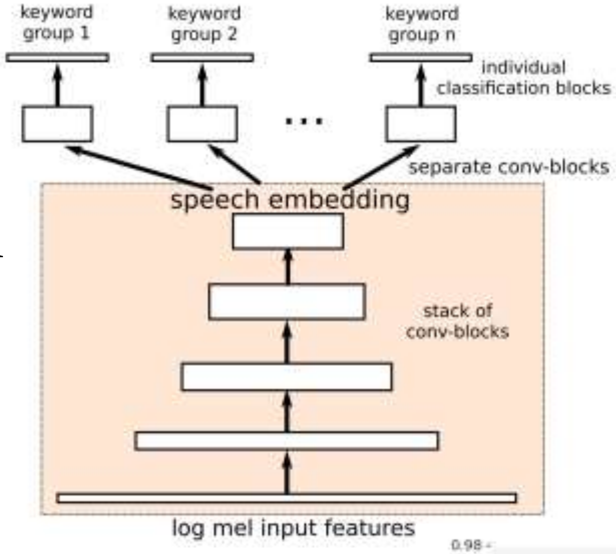
(a) meta+SSL (b) meta only (c) SSL only

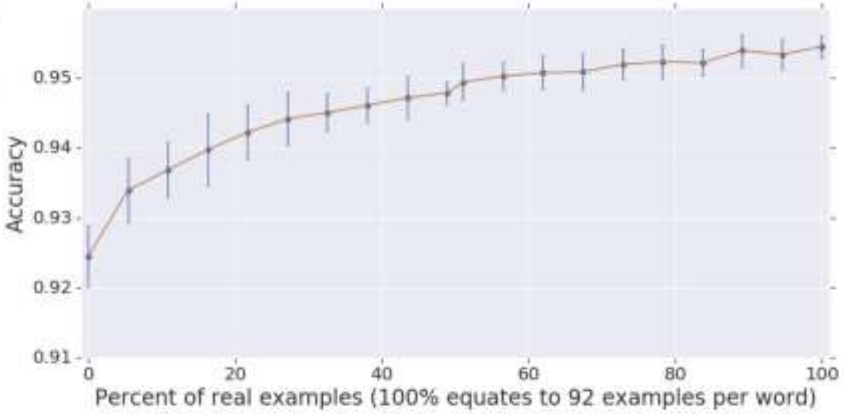
	ANIL	Matching	Trans-1
1-shot	63.62	80.41	22.77
5-shot	77.32	90.3	24.33

Kao W T, Wu Y K, Chen C P, et al. On the efficiency of integrating self-supervised learning and meta-learning for user-defined few-shot keyword spotting[C]//2022 IEEE Spoken Language Technology Workshop (SLT). IEEE, 2023: 414-421.

TRAINING KEYWORD SPOTTERS WITH LIMITED AND SYNTHESIZED SPEECH DATA

*Google Research.

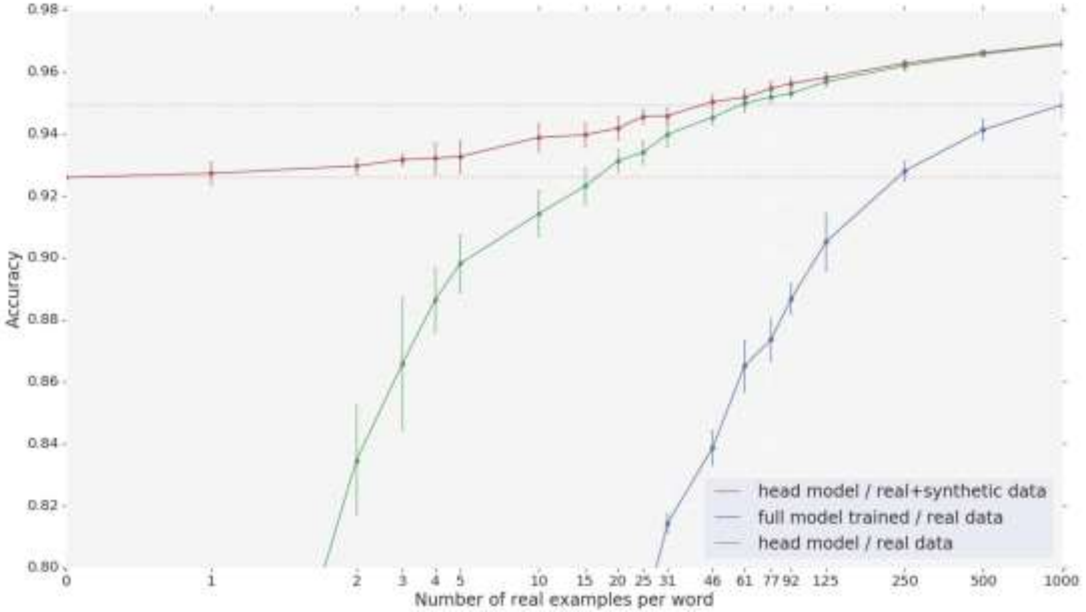




- **Motivation**
  - Explore the effectiveness of synthesized speech data in training small, spoken term detection models.
- **Model architecture**
  - Embedding model.

    Each Conv block consists of 5 layers
  - Head model.
- **Result**

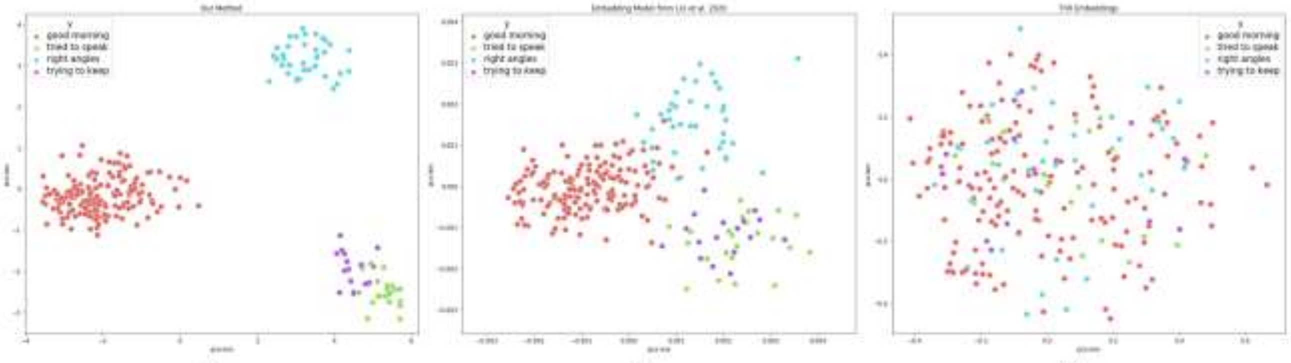
Training data	Size	Full model	Head model
speech commands	80k	97.4%	97.7%
synthetic data	3220	56.7%	92.6%
equivalent real data	3220	88.7%	95.3%



Lin J, Kilgour K, Roblek D, et al. Training keyword spotters with limited and synthesized speech data[C] (ICASSP2020).

Teaching keyword spotters to spot new keywords with limited examples

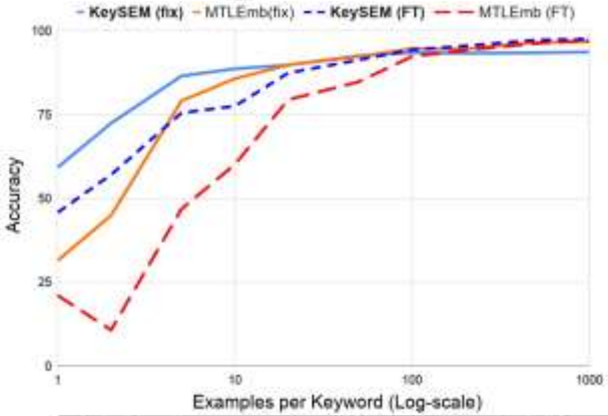
*Google Research, Switzerland. *Indian Institute of Technology Bombay, India.

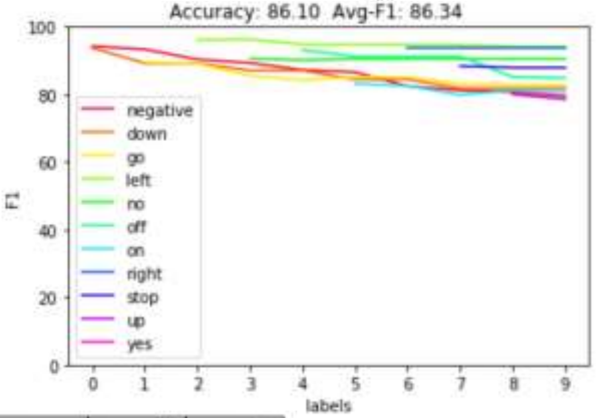


(a) (b) (c)

- **Motivation**
  - Present a speech embedding model (KeySEM) which allows for more accurate KWS models to be learned from fewer training examples.

- **Model architecture**
  - Similar architecture to Lin et al.





- **Result**

Method	SC	LK-c	LK-o	ja	eo	pl	pt
Matchbox	98.0	97.3	89.8	76.0	88.5	85.0	84.3
MHAtt-RNN	98.0	99.7	95.3	86.0	87.0	87.3	79.3
MTLEmb (fix)	96.6	95.1	87.2	86.7	87.4	89.5	82.6
MTLEmb (FT)	97.7	94.9	88.1	75.0	79.6	74.2	72.1
KeySEM (rand)	97.2	93.6	78.1	83.2	84.1	85.4	78.3
KeySEM (fix)	93.9	**99.8**	**97.8**	92.3	**91.2**	**90.3**	82.4
KeySEM (FT)	**98.2**	97.8	93.2	**92.9**	89.1	**90.3**	**84.7**

Method	SC	LK-c	LK-o	ja	eo	pl	pt
Matchbox	45.3	3.0	2.8	48.0	60.5	58.3	36.3
MHAtt-RNN	48.2	27.3	12.8	55.0	62.5	68.0	53.0
MTLEmb (fix)	79.1	43.9	33.1	76.5	74.5	76.3	71.3
MTLEmb (FT)	46.8	30.1	21.4	41.3	48.0	43.8	37.6
KeySEM (fix)	**86.5**	**98.5**	**94.5**	**86.2**	**87.4**	**86.3**	**80.8**
KeySEM (FT)	75.5	62.3	44.4	82.1	82.7	83.9	75.8

Awasthi A, Kilgour K, Rom H. Teaching keyword spotters to spot new keywords with limited examples[J]. arXiv preprint arXiv:2106.02443, 2021.

Few-Shot Keyword Spotting in Any Language

*Harvard University, USA. *Coqui, Germany. *Google, USA.

- **Motivation**
  - Training KWS models requires the manual collection and curation of thousands of target samples across a diverse pool of speakers and accents for each keyword of interest. It is a prohibitive requirement for under-resourced languages.

- **Methods**
  - Introduce a few-shot transfer learning method for keyword spotting in any language.

    Leveraging Common Voice corpora, by applying forced alignment to automatically extract 760 frequent words across nine languages and use it to train an embedding model.

    Then finetune this embedding model to classify a target keyword.

Mazumder M, Banbury C, Meyer J, et al. Few-shot keyword spotting in any language[J]. arXiv preprint arXiv:2104.01454, 2021.

Few-Shot Keyword Spotting in Any Language

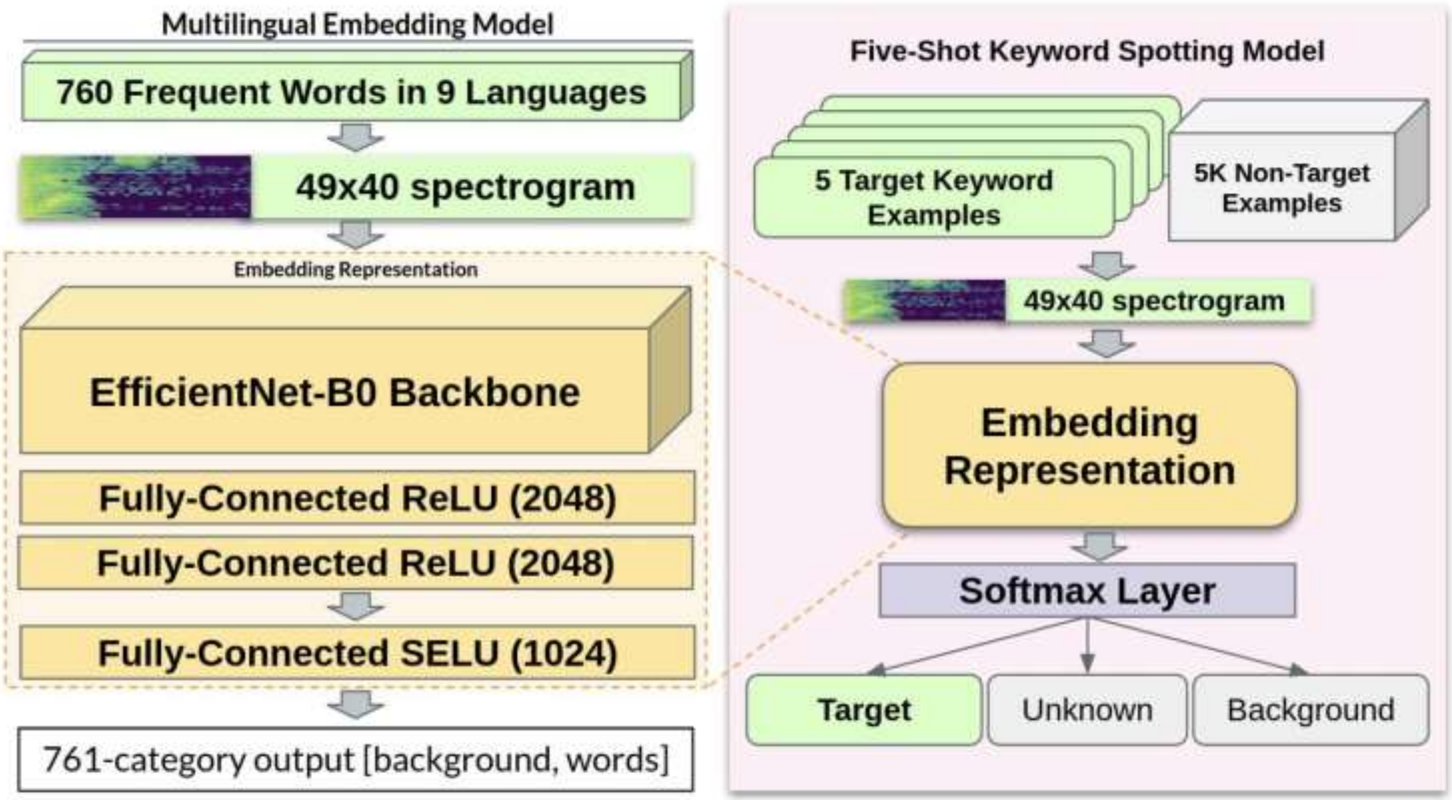
*Harvard University, USA. *Coqui, Germany. *Google, USA.

- **Model architecture**
  - Multilingual Embedding Model
  - 5-shot Transfer Learning Model

- **Result**

Language	# words	# train	# val	val acc
English	265	518760	57640	78.95
German	152	287100	31900	79.90
French	105	205920	22880	79.16
Kinyarwanda	68	134640	14960	73.64
Catalan	80	132660	14740	87.63
Persian	35	69300	7700	85.70
Spanish	31	61380	6820	79.65
Italian	17	31680	3520	81.16
Dutch	7	13860	1540	72.60
Model	760	1455300	161700	79.81

Training \ Test	GSC	Extracted
GSC	93.42%	90.49%
Extracted	78.07%	92.23%



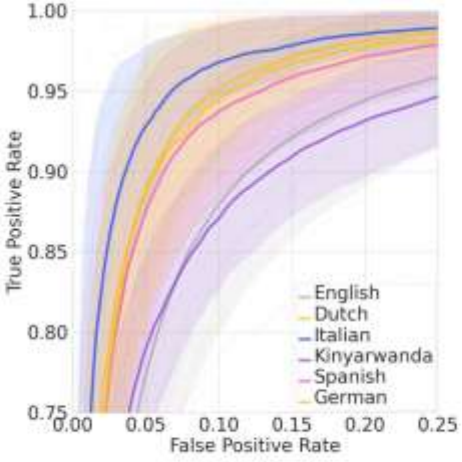
(a) *Multilingual embedding model* (b) *5-shot keyword spotting*

Mazumder M, Banbury C, Meyer J, et al. Few-shot keyword spotting in any language[J]. arXiv preprint arXiv:2104.01454, 2021.

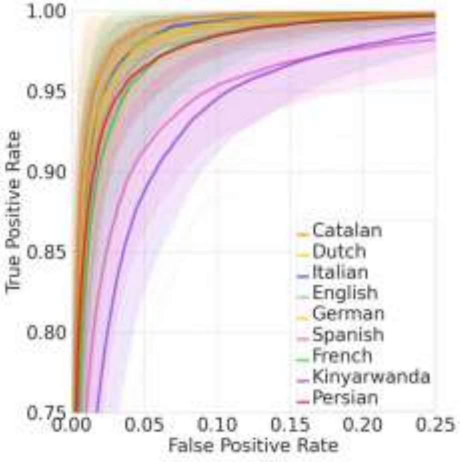
Few-Shot Keyword Spotting in Any Language

*Harvard University, USA. *Coqui, Germany. *Google, USA.

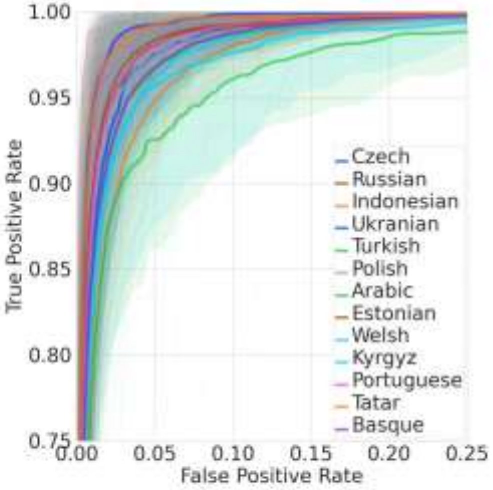
- **Result**



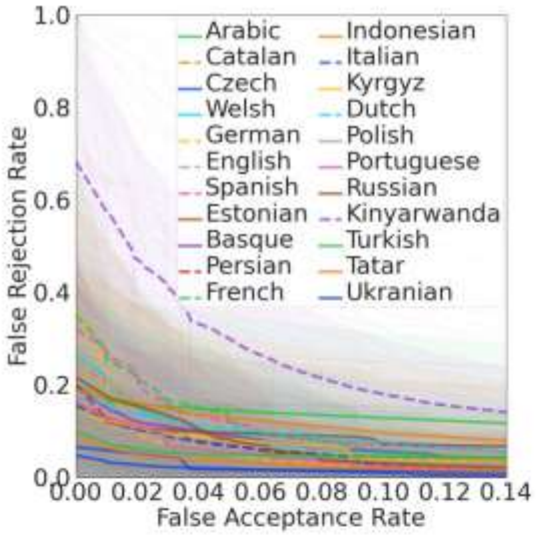
(a) *6 Monolingual Embeddings*



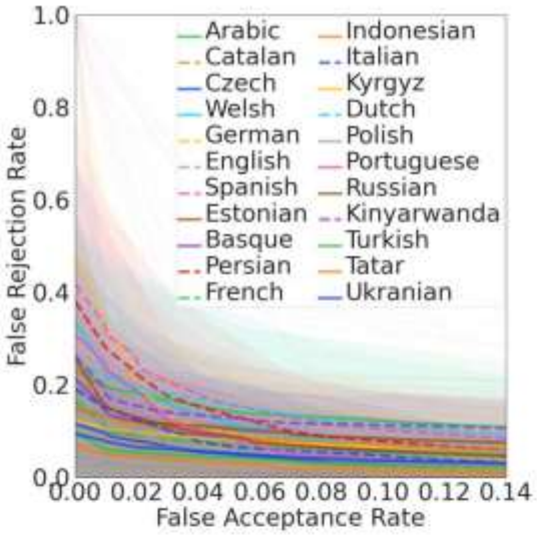
(b) *Multilingual Embedding*



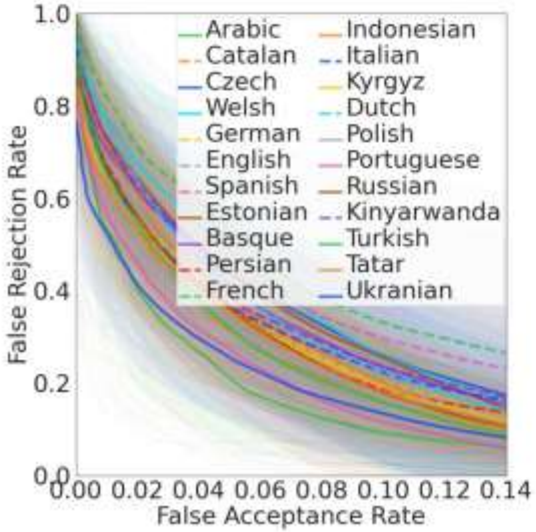
(c) *Generalization to New Languages*



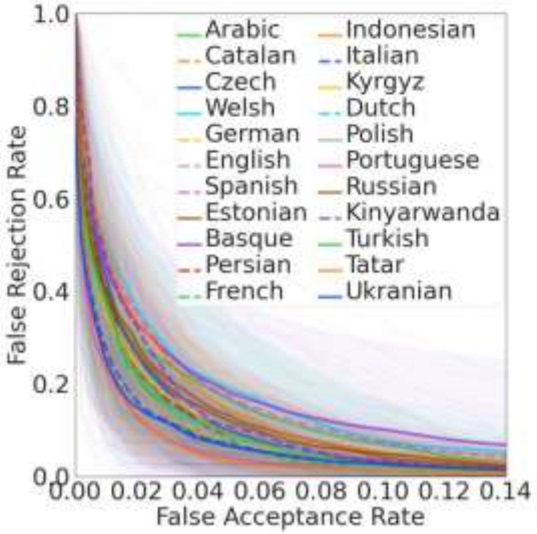
(a) *Baseline Keyword Spotting*



(b) *Context Keyword Spotting*



(c) *Baseline Keyword Search*



(d) *Context Keyword Search*

Mazumder M, Banbury C, Meyer J, et al. Few-shot keyword spotting in any language[J]. arXiv preprint arXiv:2104.01454, 2021.

METRIC LEARNING FOR USER-DEFINED KEYWORD SPOTTING

*Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea. *Hyundai Motor Company. *42dot Inc., Seoul,Republic of Korea.

- **Motivation**
  - Detect new spoken terms defined by users.

- **Methods**
  - Propose a metric learning-based training strategy for user-defined keyword spotting.

    (1) Construct a large-scale keyword dataset with an existing speech corpus and propose a filtering method to remove data that degrade model training.

    (2)Propose a two-stage training strategy (pre-train + finetune).

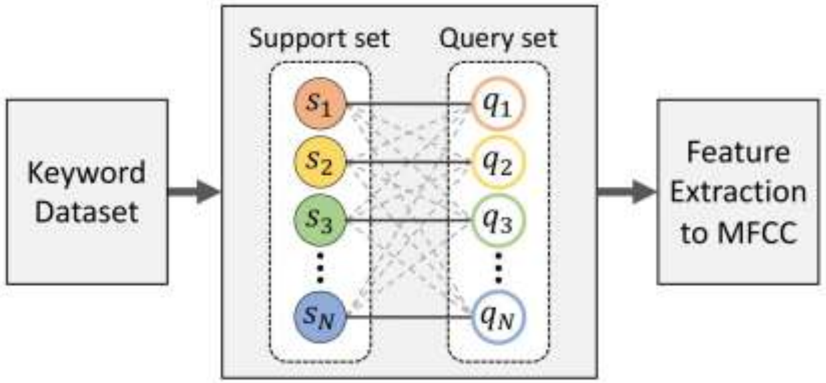
    (3)Propose unified evaluation protocol and metrics(FRR at given FAR).

Jung J, Kim Y, Park J, et al. Metric Learning for User-Defined Keyword Spotting[C] (ICASSP 2023)

METRIC LEARNING FOR USER-DEFINED KEYWORD SPOTTING

*Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea. *Hyundai Motor Company. *42dot Inc., Seoul,Republic of Korea.

- **LibriSpeech Keywords (LSK) Dataset**
  - Utilize a pre-trained wav2vec 2.0 model to force-align individual words from utterance-level labels.
  - Compute CER score on each keyword in dataset with the pre-trained wav2vec 2.0 model to filter misaligned examples.
  - The 13 most frequent words and one-letter words are removed, because they consist mostly of articles and prepositions.
  - 10 keywords in GSC dataset that are used as the user-defined keywords are removed.
- **Training Strategy**
  - Pre-train(LSK) + Finetune(25 keywords of GSC)
  - Compare the softmax loss, AM-Softmax and Angular Prototypical loss



Jung J, Kim Y, Park J, et al. Metric Learning for User-Defined Keyword Spotting[C] (ICASSP 2023)

METRIC LEARNING FOR USER-DEFINED KEYWORD SPOTTING

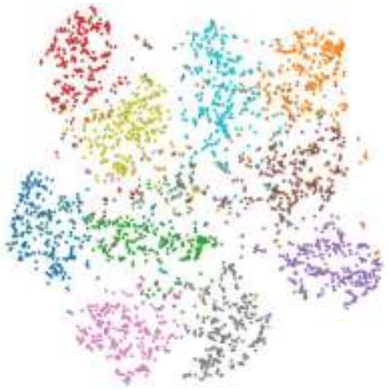
*Korea Advanced Institute of Science and Technology, Daejeon, Republic of Korea.   *Hyundai Motor Company.   *42dot Inc., Seoul,Republic of Korea.

- **Result**

Training loss		EER ↓			Acc ↑			F1-score ↑			FRR@FAR=2.5 ↓			FRR@FAR=10 ↓		
Pre-train	Fine-tune	1-shot	5-shot	10-shot	1-shot	5-shot	10-shot	1-shot	5-shot	10-shot	1-shot	5-shot	10-shot	1-shot	5-shot	10-shot
[19] w/ Inc. Training		-	-	9.0†	-	-	-	-	-	-	-	-	17.0†	-	-	8.3†
-	Softmax	17.31	9.52	7.79	69.57	84.13	84.10	0.68	0.84	0.84	44.47	24.30	19.83	24.17	8.83	6.03
	AM-Soft	17.43	8.91	7.20	63.43	84.60	86.97	0.63	0.85	0.87	55.10	21.33	18.30	26.57	7.73	5.10
	AP	20.47	9.33	8.50	61.37	80.30	80.13	0.60	0.80	0.80	56.53	26.60	23.00	35.47	8.57	6.93
Softmax	-	30.77	20.64	19.01	47.07	62.23	67.23	0.47	0.63	0.68	66.10	59.57	44.10	51.20	36.87	27.77
	Softmax	16.91	11.00	9.20	69.47	83.23	85.47	0.68	0.83	0.85	48.33	26.67	21.67	25.10	11.67	8.67
	AM-Soft	10.47	4.75	4.01	85.43	94.80	95.33	0.85	0.95	0.95	24.20	6.90	5.97	10.87	3.07	2.03
	AP	10.10	5.20	3.77	83.53	94.23	95.00	0.83	0.94	0.95	23.00	7.67	5.47	10.23	3.40	2.20
AM-Soft	-	34.78	26.87	22.65	41.73	56.83	63.30	0.43	0.58	0.64	75.77	75.00	61.53	61.80	51.23	38.87
	Softmax	23.60	15.38	13.80	53.17	70.50	77.93	0.54	0.70	0.78	65.23	44.07	37.87	41.73	22.73	18.27
	AM-Soft	10.88	6.54	5.64	85.13	92.57	93.63	0.85	0.93	0.94	26.40	11.87	9.60	11.63	4.80	3.50
	AP	11.80	6.57	4.80	80.27	92.40	93.07	0.79	0.92	0.93	28.20	12.43	7.63	13.70	4.70	3.13
AP	-	32.70	24.81	21.01	41.23	60.03	69.37	0.45	0.61	0.70	80.50	74.03	57.67	59.60	50.77	36.23
	Softmax	15.81	10.97	8.87	70.07	80.77	83.33	0.71	0.81	0.84	55.10	29.60	20.77	25.83	12.07	7.57
	AM-Soft	8.08	5.31	4.27	88.53	**94.03**	95.53	0.88	**0.94**	**0.96**	17.10	7.50	5.67	6.90	3.37	2.60
	AP	**7.77**	**4.49**	**3.24**	**89.97**	93.93	**95.97**	**0.90**	**0.94**	**0.96**	**16.77**	**6.67**	**4.20**	**5.93**	**2.47**	**1.20**

Dataset	# Classes	# Samples	EER ↓	Acc ↑	F1-score ↑	FRR@FAR=2.5 ↓	FRR@FAR=10 ↓
LSK	500	500	4.13	94.50	0.95	6.07	2.13
	500	1,000	3.94	94.60	0.95	5.23	1.97
	1,000	500	3.63	95.07	0.95	5.13	1.47
	1,000	1,000	3.24	**95.97**	**0.96**	4.20	**1.20**
LSK+KSK	2,000	1,000	**3.07**	95.63	**0.96**	**3.73**	**1.20**

Filtering	EER ↓	Acc ↑	F1-score ↑	FRR@FAR=2.5 ↓	FRR@FAR=10 ↓
✗	3.47	95.67	**0.96**	4.83	1.47
✓	**3.24**	**95.97**	**0.96**	**4.20**	**1.20**



(a) Trained only on the GSC dataset.

(b) Pre-trained only on the LSK dataset.

(c) Pre-trained on the LSK dataset, then fine-tuned on the GSC dataset.

Jung J, Kim Y, Park J, et al. Metric Learning for User-Defined Keyword Spotting[C] (ICASSP 2023)

Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems

*PSI, KU Leuven, Belgium.

- **Motivation**
  - The design of a custom KWS algorithm typically demands the training of a model on a dataset of collected user-defined keywords, preventing users from obtaining a custom solution in a short time.

- **Methods**
  - Contributes an evaluation framework for FSL architectures composed by a feature encoder and a prototype-based open-set classifier initialized with few-shot samples.

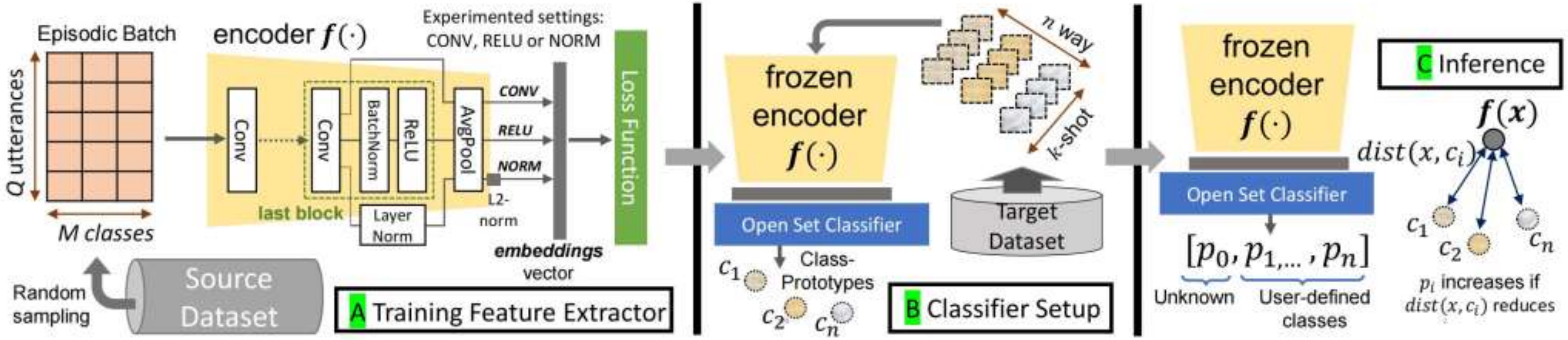
    Train a feature extractor using the prototypical loss, its angular variant or the triplet loss.

Rusci M, Tuytelaars T. Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems[J].(INTERSPEECH2023)

Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems

*PSI, KU Leuven, Belgium.

- **Framework**



- **Model architecture**
  - Feature encoder

    DSCNN: composed of a stacked sequence of depthwise and pointwise convolution blocks)
  - Open-set classifier

    Open Nearest Class Mean (openNCM)

    OpenMAX

    Dummy Proto (DProto)

$$c_j = \frac{1}{S}\sum_{i=1}^{S} f(x^S_{i,j}) \quad (1)$$

$$L_{PN} = -\frac{1}{Q \cdot M}\sum_{i=1}^{Q}\sum_{j=1}^{M}\log\frac{\exp(\mathbf{s}(x^Q_{i,j}, j))}{\sum_{k=1}^{M}\exp(\mathbf{s}(x^Q_{i,j}, k))} \quad (2)$$

where

$$\mathbf{s}(x, j) = -d_{L2}(f(x), c_j) \quad (3)$$

$$\mathbf{s}(x, j) = w \cdot (\cos(f(x), c_j) - m) + b \quad (4)$$

$$L_{TL} = \frac{1}{N_t}\sum_{i=1}^{N_t}\max(0, d_{L2}(x_i, x_i^+) - d_{L2}(x_i, x_i^-) + m) \quad (5)$$

Rusci M, Tuytelaars T. Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems[J].(INTERSPEECH2023)

Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems

*PSI, KU Leuven, Belgium.

- **Result**

Loss	Feature Extractor	openNCM 5-shot $ACC^{+}_{5\%}$	openNCM 5-shot AUROC	openNCM 5-shot $FRR^{+}_{5\%}$	OpenMAX 5-shot $ACC^{+}_{5\%}$	OpenMAX 5-shot AUROC	OpenMAX 5-shot $FRR^{+}_{5\%}$	Dproto 5-shot $ACC^{+}_{5\%}$	Dproto 5-shot AUROC	Dproto 5-shot $FRR^{+}_{5\%}$
PN	DSCNN-L-NORM	0.21	0.66	0.78	0.23	0.79	0.77	0.21	0.64	0.79
	DSCNN-L-CONV	0.54	0.86	0.46	0.12	0.87	0.87	0.64	0.91	0.35
	DSCNN-L-RELU	**0.56**	**0.87**	**0.43**	0.14	0.91	0.85	**0.66**	**0.92**	**0.32**
AP	DSCNN-L-NORM	**0.66**	**0.92**	**0.29**	0.44	0.94	0.54	**0.65**	**0.93**	**0.30**
TL	DSCNN-L-NORM	**0.71**	**0.93**	**0.26**	0.37	0.94	0.62			
	DSCNN-L-CONV	0.58	0.88	0.41	0.25	0.95	0.74			
	DSCNN-L-RELU	0.66	0.90	0.33	0.20	0.96	0.80			
PN	DSCNN-S-NORM	0.14	0.57	0.85	0.14	0.72	0.85	0.14	0.54	0.85
	DSCNN-S-CONV	**0.40**	**0.81**	**0.60**	0.14	0.83	0.84	**0.40**	**0.81**	**0.59**
	DSCNN-S-RELU	0.39	0.80	0.60	0.20	0.86	0.77	0.39	0.80	0.60
AP	DSCNN-S-NORM	**0.39**	**0.83**	**0.60**	0.34	0.87	0.64	0.31	0.81	0.68
TL	DSCNN-S-NORM	**0.51**	**0.87**	**0.46**	0.38	0.91	0.59			
	DSCNN-S-CONV	0.39	0.80	0.60	0.26	0.92	0.70			
	DSCNN-S-RELU	0.42	0.82	0.57	0.28	0.92	0.69			

Loss	Feature Extractor	openNCM 10-shot $ACC^{+}_{5\%}$	openNCM 10-shot AUROC	openNCM 10-shot $FRR^{+}_{5\%}$	OpenMAX 10-shot $ACC^{+}_{5\%}$	OpenMAX 10-shot AUROC	OpenMAX 10-shot $FRR^{+}_{5\%}$	Dproto 10-shot $ACC^{+}_{5\%}$	Dproto 10-shot AUROC	Dproto 10-shot $FRR^{+}_{5\%}$
PN	DSCNN-L-NORM	0.22	0.68	0.78	0.23	0.75	0.77	0.22	0.67	0.78
	DSCNN-L-CONV	0.62	0.89	0.37	0.48	0.89	0.50	0.71	0.93	0.28
	DSCNN-L-RELU	**0.63**	**0.89**	**0.37**	**0.56**	**0.92**	**0.40**	**0.71**	**0.93**	**0.28**
AP	DSCNN-L-NORM	**0.71**	**0.93**	**0.25**	**0.66**	**0.93**	**0.30**	**0.70**	**0.94**	**0.25**
TL	DSCNN-L-NORM	**0.76**	**0.94**	**0.21**	**0.71**	**0.94**	**0.24**			
	DSCNN-L-CONV	0.63	0.89	0.36	0.67	0.95	0.29			
	DSCNN-L-RELU	0.71	0.91	0.28	0.64	0.96	0.32			
PN	DSCNN-S-NORM	0.17	0.59	0.83	0.17	0.69	0.83	0.15	0.55	0.84
	DSCNN-S-CONV	**0.48**	**0.85**	**0.51**	0.38	0.86	0.60	0.43	0.83	0.57
	DSCNN-S-RELU	0.45	0.84	0.54	**0.44**	**0.87**	**0.54**	**0.44**	**0.81**	**0.56**
AP	DSCNN-S-NORM	**0.41**	**0.84**	**0.57**	**0.36**	**0.86**	**0.63**	0.33	0.82	0.66
TL	DSCNN-S-NORM	**0.56**	**0.89**	**0.42**	0.54	0.91	0.42			
	DSCNN-S-CONV	0.42	0.82	0.57	0.56	0.92	0.39			
	DSCNN-S-RELU	0.49	0.85	0.50	**0.58**	**0.93**	**0.37**			

DSCNN-L — params: 407k	$ACC^{+}_{5\%}$	AUROC	Train Data	Extra Params
openNCM+*Classif* [22]+ NORM	0.52	0.89	*source*	-
openNCM+TL+NORM	0.76	0.94	*source*	-
dProto [10]+RELU	0.71	0.93	*source*	-
PEELER [20]	0.76	0.94	*source*	+6.3M
end-to-end [4]	0.76	0.93	*target*	-

DSCNN-S — params: 22k	$ACC^{+}_{5\%}$	AUROC	Train Data	Extra Params
openNCM+*Classif* [22]+ NORM	0.47	0.85	*source*	-
openNCM+TL+NORM	0.56	0.89	*source*	-
dProto [10]+RELU	0.44	0.82	*source*	-
PEELER [20]	0.60	0.88	*source*	+341k
end-to-end [4]	0.72	0.93	*target*	-

Rusci M, Tuytelaars T. Few-Shot Open-Set Learning for On-Device Customization of KeyWord Spotting Systems[J].(INTERSPEECH2023)

Experimental Progress

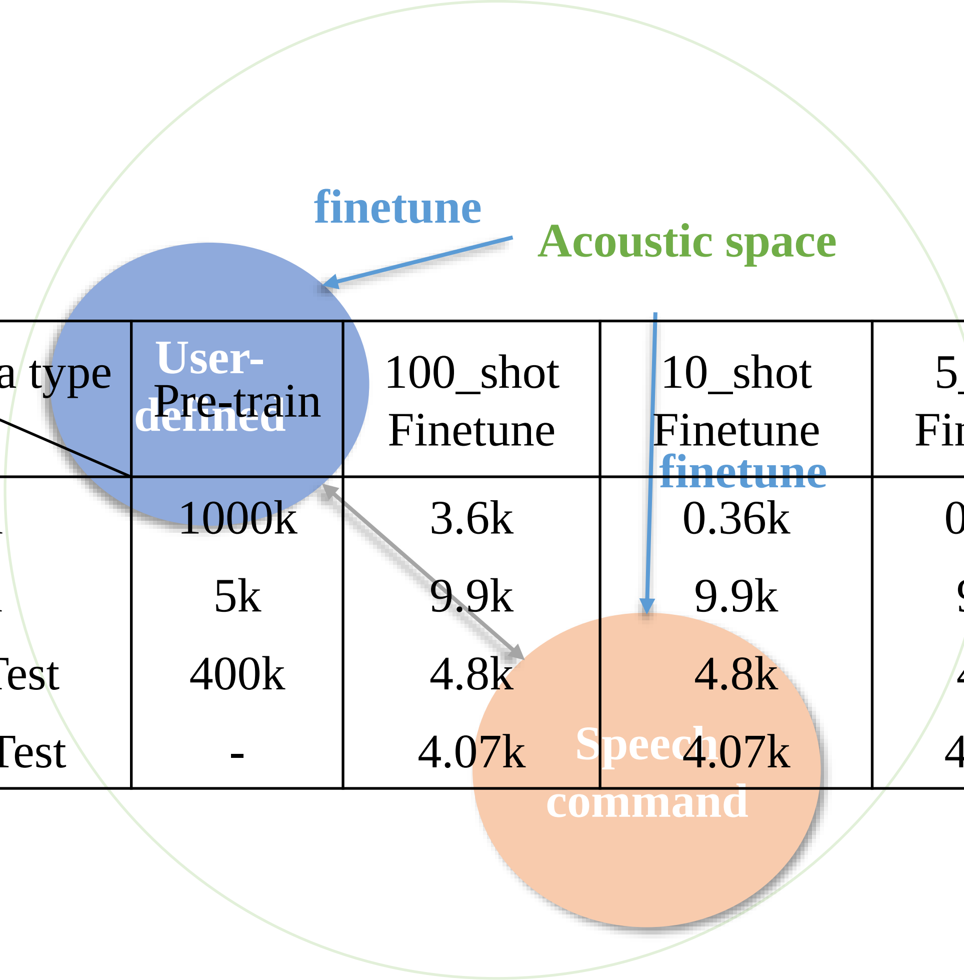
- Model Pretraining based Mix training

Motivation:

(1)Learning a whole Acoustic space by large-scale KWS dataset, Finetuning on the few keyword examples.

(2)Verifying the effectiveness of Few-shot mix training in mixed scenario.

(3)Verifying whether the pattern learned by mix training outperforms the base-based training strategy in clean scenario.



Preparing Pre-train Dataset:

(a)Utilizing MFA to force-align individual words

Word utterance-level labels in Librispeech 960.

(b)The words with lengths of less than 5 and overlapping with GSC are removed.

Set type \ Data type	Pre-train	100_shot Finetune	10_shot Finetune	5_shot Finetune
Train	1000k	3.6k	0.36k	0.18k
Valid	5k	9.9k	9.9k	9.9k
Clean_Test	400k	4.8k	4.8k	4.8k
Mixed_Test	-	4.07k	4.07k	4.07k

Preparing Finetuning Dataset:

(a)5 training subsets are randomly sampled from the training set of GSC_v2.

(b)Each subset contains 36 classes, and 100/10/5 audios are randomly sampled from each class.

Experimental Progress

- Model Pretraining based Mix training

Result

(1)Pretrain model performance

(2)Without model pretraining performance in clean scenario

The performance on pre-train model

Strategy \ metric	Acc.(%)	EER
Base(CE+softmax)	90.93	0.0025
MT(BCE+sigmoid)	91.47	0.0035

The performance in Top-1 Accuracy(%) of different training strategies when detecting clean keywords. All the models are trained without model pretraining.

	GSC_v2	GSC_100_shot	GSC_10_shot	GSC_5_shot
	Top-1 Acc.(%)	Top-1 Acc.(%)	Top-1 Acc.(%)	Top-1 Acc.(%)
Base(CE+softmax)	96.81	84.56±0.62	overfit	overfit
MT(BCE+sigmod)	97.03	**87.71±0.54**	23.96±2.74	16.03±2.12

Experimental Progress

- Model Pretraining based Mix training

Result

(3)Without model pretraining performance in mixed scenario.

The performance in Top-2 Accuracy(%) of different training strategies when detecting mixed keywords. All the models are trained without model pretraining.

	GSC_v2	GSC_100_shot	GSC_10_shot	GSC_5_shot
	Top-2 Acc.(%)	Top-2 Acc.(%)	Top-2 Acc.(%)	Top-2 Acc.(%)
Base(CE+softmax)	60.59	42.64±0.93	overfit	overfit
MT(BCE+sigmod)	90.35	**69.65±0.49**	16.45±1.56	10.69±2.01

Experimental Progress

- Model Pretraining based Mix training

Result

(4)Model pretraining performance in clean scenario.

Finetune: start update layer = the 7th MBConvBlock

Pre-train		Finetune		GSC_10_shot	GSC_10_shot_Aug
MT	base	MT	base	Top-1 Acc.(%)	Top-1 Acc.(%)
✗	✓	✗	✓	60.67±2.32	**66.98±1.23**
✗	✓	✓	✗	**61.45±0.82**	63.18±1.36
✓	✗	✗	✓	61.46±2.25	66.51±1.37
✓	✗	✓	✗	59.11±1.31	61.41±1.58

Pre-train		Finetune		GSC_v2	GSC_100_shot
MT	base	MT	base	Top-1 Acc.(%)	Top-1 Acc.(%)
✗	✓	✗	✓	96.36	91.82±0.38
✗	✓	✓	✗	**97.14**	**92.60±0.31**
✓	✗	✗	✓	96.22	91.84±0.28
✓	✗	✓	✗	97.12	91.91±0.33

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
MT	base	MT	base	Top-1 Acc.(%)	Top-1 Acc.(%)
✗	✓	✗	✓	33.50±4.24	38.15±5.06
✗	✓	✓	✗	**43.05±3.08**	41.75±4.03
✓	✗	✗	✓	31.37±3.80	37.23±3.07
✓	✗	✓	✗	42.24±3.31	**42.30±4.17**

Experimental Progress

- Model Pretraining based Mix training

Result

(5)Model pretraining performance in mixed scenario.

Finetune: start update layer = the 7th MBConvBlock

Pre-train		Finetune		GSC_10_shot	GSC_10_shot_Aug
MT	base	MT	base	Top-2 Acc.(%)	Top-2 Acc.(%)
✗	✓	✗	✓	39.70±0.64	34.94±1.81
✗	✓	✓	✗	42.80±0.67	41.84±1.34
✓	✗	✗	✓	42.03±0.77	39.19±1.95
✓	✗	✓	✗	**44.76±1.92**	**42.78±1.15**

Pre-train		Finetune		GSC_v2	GSC_100_shot
MT	base	MT	base	Top-2 Acc.(%)	Top-2 Acc.(%)
✗	✓	✗	✓	60.86	52.95±1.14
✗	✓	✓	✗	87.82	74.32±0.69
✓	✗	✗	✓	64.31	55.24±0.34
✓	✗	✓	✗	**90.61**	**78.06±0.59**

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
MT	base	MT	base	Top-2 Acc.(%)	Top-2 Acc.(%)
✗	✓	✗	✓	20.34±2.74	25.15±4.09
✗	✓	✓	✗	30.40±2.57	29.44±3.17
✓	✗	✗	✓	21.34±2.99	27.26±0.95
✓	✗	✓	✗	**32.08±3.21**	**31.80±3.31**

Experimental Progress

- Model Pretraining based Mix training

  Result

  (6)Model pretraining performance in clean scenario.

  (a)start update layer: the 7th MBConvBlock

  (b)start update layer: the last FC layer

The performance in Top-1 Accuracy(%) on (a).

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
MT	base	MT	base	Top-1 Acc.(%)	Top-1 Acc.(%)
✗	✓	✗	✓	33.50±4.24	38.15±5.06
✗	✓	✓	✗	**43.05±3.08**	41.75±4.03
✓	✗	✗	✓	31.37±3.80	37.23±3.07
✓	✗	✓	✗	42.24±3.31	**42.30±4.17**

The performance in Top-1 Accuracy(%) on (b).

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
MT	base	MT	base	Top-1 Acc.(%)	Top-1 Acc.(%)
✗	✓	✗	✓	48.29±2.23	51.32±3.32
✗	✓	✓	✗	50.08±2.57	51.29±3.07
✓	✗	✗	✓	**51.59±3.38**	**53.02±3.03**
✓	✗	✓	✗	51.54±3.08	51.73±3.34

Experimental Progress

- Model Pretraining based Mix training

  Result

  (7)Model pretraining performance in mixed scenario.

    (a)start update layer: the 7th MBConvBlock

    (b)start update layer: the last FC layer

The performance in Top-2 Accuracy(%) on (a).

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
				Top-2 Acc.(%)	Top-2 Acc.(%)
MT	base	MT	base		
✗	✓	✗	✓	20.34±2.74	25.15±4.09
✗	✓	✓	✗	30.40±2.57	29.44±3.17
✓	✗	✗	✓	21.34±2.99	27.26±0.95
✓	✗	✓	✗	**32.08±3.21**	**31.80±3.31**

The performance in Top-2 Accuracy(%) on (b).

Pre-train		Finetune		GSC_5_shot	GSC_5_shot_Aug
				Top-2 Acc.(%)	Top-2 Acc.(%)
MT	base	MT	base		
✗	✓	✗	✓	30.03±3.25	33.51±2.38
✗	✓	✓	✗	34.10±2.83	34.43±2.79
✓	✗	✗	✓	35.83±4.08	38.22±3.24
✓	✗	✓	✗	**37.46±3.36**	**39.02±3.33**

Experimental Progress

- Model Pretraining based Mix training

Conclusion

(1) In the Few-shot case, model pretraining demonstrates greater effectiveness, with its advantages diminishing gradually as the number of shots increases.

Only in the 5-shot, finetuning the final layer yield optimal results.

(2) In the Few-shot case, mix training(MT) proves superior to base-based methods in mixed scenario, with the optimal performance achieved through MT+MT.

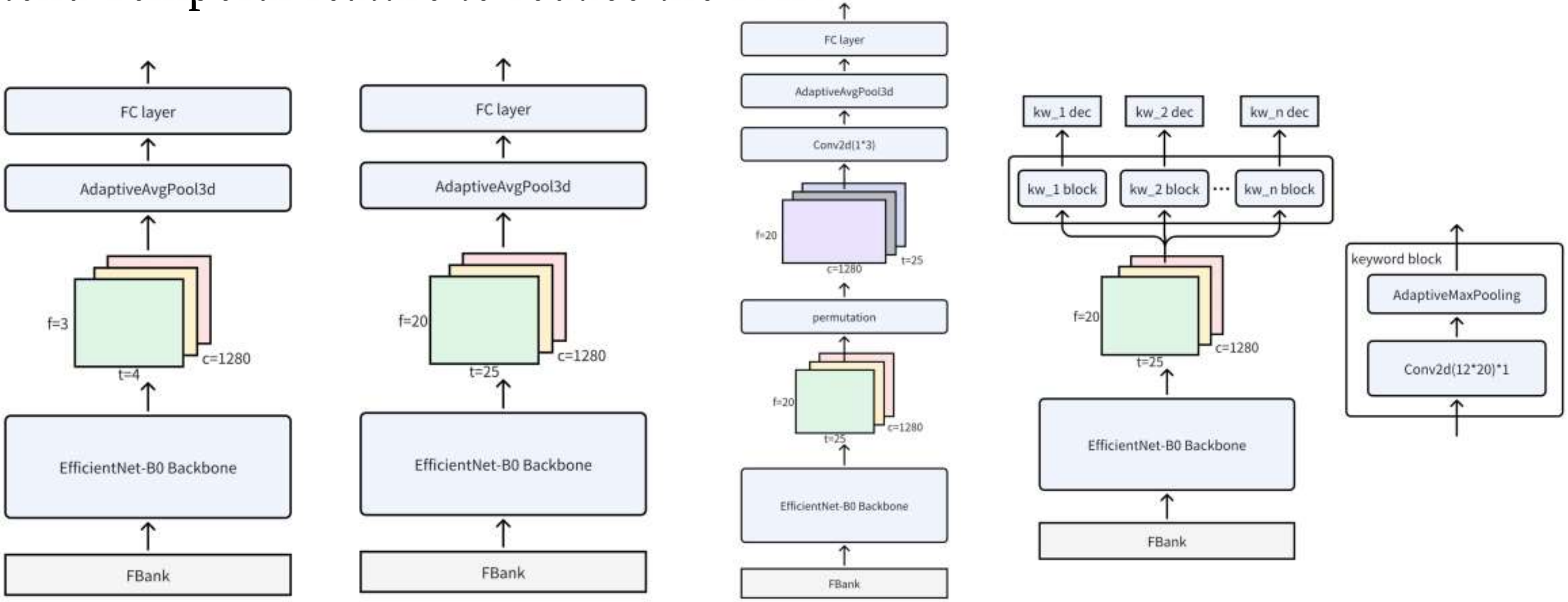
In extreme scenarios(10-shot/5-shot), the overall performance is consistently poor.

(3) Mix training shows a slight advantage over base-based methods in clean scenario, possibly due to the better learning of patterns.

In extreme scenarios(10-shot/5-shot), it is challenging to discern this trend.

Experimental Progress

- Extend Temporal feature to reduce the FAR



(a) (b) (c) (d)

Experimental Progress

- Extend Temporal feature to reduce the FAR

Dataset

Sampled on fyt dataset (pretrain: 5000 keywords, finetune: 51 keywords + unknown).

Result

(1)Model pretraining(pretrain: base, finetune: MT) performance on fyt clean scenario.

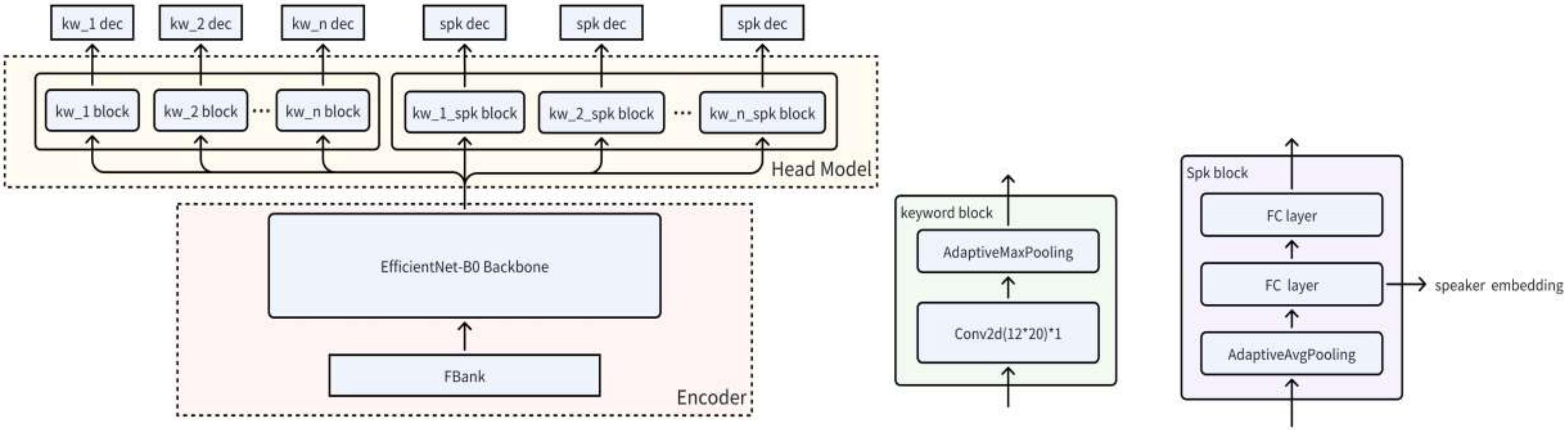
Set type \ Data type	Pre-train	100_shot Finetune
Train	1480k	5.2k
Valid	8.24k	3.28k
ACC_Test	-	1.55k
FAR_Test	-	8k

Model structure	Top-1 Acc.(%)	FAR(%)
(a) raw	**97.07±0.37**	**1.21±0.32**
(b) CHG str	96.81±0.25	1.32±0.66
(c) CHG str + trans + 2d-conv	96.64±0.37	7.82±7.12
(d) CHG str + 52 2d-conv	95.91±0.31	6.22±2.26
(d) CHG str + 52 2d-conv+SE	96.36±0.34	7.62±2.61

Feature Work

- Multi-task(KWS+SID/ASV) based mix training

  Structure



Feature Work

- Multi-task(KWS+SID/ASV) based mix training

Result

(1)Model performance in clean scenario.

Training strategy / dataset	KWS+SID task			KWS+ASV task	
	KW Top-1 Acc.(%)	Spk Top-1 Acc.(%)	Total Acc.(%)	KW Top-1 Acc.(%)	Spk EER.(%)
base / data_v1	94.65	73.49	71.86	86.45	4.67
base / data_v1_aug	95.12	90.23	86.51	86.04	4.14
MT(self corruption) / data_v1	**100**	**91.39**	**91.39**	**93.38**	**3.22**
MT(self corruption) / data_v1_aug	**99.53**	**90.93**	**90.69**	**92.77**	**2.86**

Feature Work

- Multi-task(KWS+SID/ASV) based mix training

  Result

  (1)Model performance in mixed scenario.

Training strategy / dataset	KWS+SID task			KWS+ASV task	
	KW Top-2 Acc.(%)	Spk Top-1 Acc.(%)	Total Acc.(%)	KW Top-2 Acc.(%)	Spk EER.(%)
base / data_v1	58.02	10.58	0.00	52.51	24.69
base / data_v1_aug	61.16	30.23	0.46	54.79	22.02
MT(self corruption) / data_v1	94.53	47.44	17.67	83.35	10.57
MT(self corruption) / data_v1_aug	**95.58**	**78.14**	**56.74**	**84.56**	**9.04**

Feature Work

- Multi-task(KWS+SID/ASV) based mix training

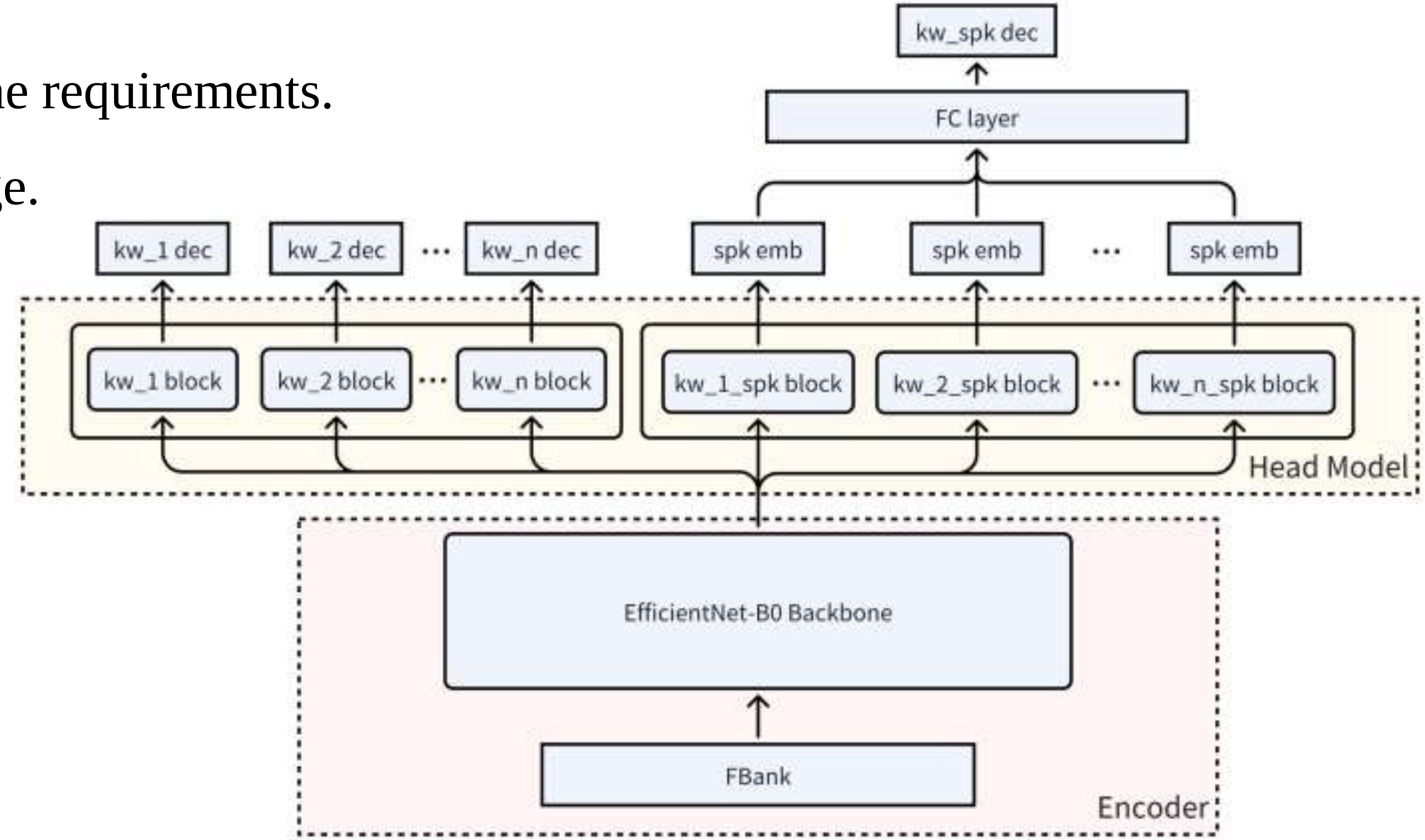
  Defect

  (1) The available data does not quite fit the requirements.

  (2) The number of parameters is very large.

  Shared FC layer

  ….



请大家批评指正

Thank You！