Real-time HAR using Neural Architecture Search

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
12th October, 2021

실험 계획 추진 보고

- ✓ Search Space의 operator를 Backbone network와 통일하여 실험
- ✓ Flops의 제한만 두고 layer의 개수에는 제한을 두지 않음.

1주차 실험(10/12)

- 기존 실험에서 Layer의 개수 30까지 실험
- Layer 3 ~ 30
- 전체 데이터셋에 대하여 3회 반복 실험
- 비교 대상 논문 목록 정리 및 요약
- 10/15(금요일) [1] ~ [5] 비교 대상 논문 실험 결과 정리

2주차 실험(10/19)

- HAR NAS[6]논문 구현 및 OPPORTUNITY 이외의 데이터셋에 적용
- 재균이형 Operator 추가 실험.
- Real-time을 위한 bottleneck(LSTM) 개선 operator: Dilated Causal Convolution + Attention Mechanism[9]

3, 4주차 실험(10/26, 11/02)

- 교수님 피드백에 따른 추가 실험 예정.

➢ 다양한 실험에도 불구하고, 개선사항이 선명하지 않을 경우 NAS 시도 자체의 의도를 살려서 연구 보고하는 것도 고려하기 바람.

➢ 즉, S9, S10, A31의 디바이스별 latency와 최적 도출 모델들의 상관관계 등 예전 실험을 보다 심도 있게 추진하여 돌파구를 찾는 실험도 병행.

비교 대상 논문 요약

논문	목표	Method	Dataset
[1]	정확도	Conv + LSTM	OPP, Skoda
[2]	Real-time	CNN + Statistics	UCI, WIS
[3]	Real-time	CNN (3 Layer)	UCI, PAM2
[4]	정확도	Inception + GRU, Pooling	OPP, PAM2, UCI
[5]	정확도	Dual Attention Module	WIS, UniMiB, PAM2, OPP
[6]	정확도	RL-NAS, Performance Prediction	OPP

비교 대상 논문 요약

[1] Deep Convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition. 2016, Sensors

- DeepConvLSTM: 4 Conv Layer + 2 LSTM Layer + FC
- 기존 연구의 한계점: CNN 모델만으로는 temporal context를 명확히 인식하지 못한다.
- Approach: Raw sensor data에서 feature를 자동 추출하기 위한 CNN과 temporal dynamics of feature activities를 포착하기 위한 LSTM을 활용.

[2] Real-time human activity recognition from accelerometer data using Convolutional Neural Networks. 2018, Applied Soft Computing

- Shallow CNN + Statistics, Real-time
- 기존 연구의 한계점: Convolution 모델은 Global properties of the signal을 포착하지 못함.
- Approach: Local features는 convolution으로 추출하고 global features인 statistical features를 추가

[3] Deep Learning Models for Real-time Human Activity Recognition with Smartphones. 2020, Mobile Networks and Applications

- CNN, LSTM, BLSTM, MLP, SVM 비교를 통해 CNN 효과성 증명
- 기존 연구의 한계점: Real-time HAR에 있어서 multimodal and high-dimensional sensor data의 복잡성을 지적
- Approach: Smartphone의 built-in accelerometer만을 활용하여 real-time HAR 수행

비교 대상 논문 요약

[4] InnoHAR: A Deep Neural Network for Complex Human Activity Recognition. 2018, IEEE Access

- Inception Block Layer + GRU Layer

- 기존 연구의 한계점: Multi-scaled and noise-mixed signals를 효과적으로 처리하는 method 필요.

- Approach: Inception Block의 다양한 kernel size를 활용하여 multi-scale feature를 추출하고, Pooling layer를 활용하여 interference noise feature를 줄임.

[5] DanHAR: Dual Attention Network for multimodal human activity recognition using wearable sensors. 2021, Applied Soft Computing

- Attention Module

- 기존 연구의 한계점: SVM, KNN, 등과 같은shallow learning 모델들은 complex activities의 특성을 잘 포착하지 못함.

- Approach: Sensor modality를 포착하는 channel attention과 long sensor sequence를 포착하는 temporal attention을 모두 활용하여 모델의 representation power를 높임.

비교 대상 논문 요약

[6] Fast Deep Neural Architecture Search for Wearable Activity Recognition by Early Prediction of Converged Performance. 2021, International Symposium on Wearable Computers

- NAS, Performance Predictor

- 기존 연구의 한계점: HAR을 위한 DNN 설계에 있어서 layer type, layer size, number of layers, connection 등 다양한 hyperparameters들로 인해 설계의 복잡도가 높음.

- Approach: DeepConvLSTM에서 feature를 추출하는 CNN구조를 NAS를 통해 설계

1) Deep Q Learning(Reinforcement Learning)을 활용한 NAS

: Search space) Kernel size: {1,2,3,5,8}, # of kernels: {32,64,128,256}, pooling size: {2,3,5}, node connection

2) NAS의 computation time을 줄이기 위한 converged performance estimation 활용

: Deep Q Learning에 활용될 reward(탐색 중인 모델의 정확도)를 추정하는 Neural regression model 사용

: Neural regression model의 input으로는 탐색 중인 모델의 구조이고 target은 Training과 validation에서 얻은 statistics(정확도, Flops, Inference Time).

		NAS	
		Effective	Efficiency
Searched Model	Effective	O	O
	Efficiency		

참고 논문 요약

[7] Sensors-based Human Activity Recognition with Convolutional Neural Network and Attention Mechanism. 2020, IEEE International Conference on Software Engineering and Service Science

- Attention Mechanism

- 기존 연구의 한계: RNN 기반 구조는 순차적 계산 방식으로 인해 병렬처리가 안되고, LSTM Cell의 정보 수용력 한계로 인해 long sequence 정보를 저장하기 어렵다.

- Approach: DeepConvLSTM에서 LSTM Layer를 attention mechanism을 대체하여 병렬처리를 가능하게하고 전체 sequence를 보는 것이 가능.

- Dataset: OPPORTUNITY, Skota

[8] Deep ConvLSTM With Self-Attention for Human Activity Decoding Using Wearable Sensors. 2021, IEEE SENSORS JOURNAL

- Self-attention mechanism

- 기존 연구의 한계점: 기존 DeepConvLSTM에서는 CNN과 LSTM으로 얻어진 spatio-temporal 정보에서 어떤 time point의 정보가 중요한지를 포착하지 못함.

- Approach: CNN과 LSTM에서 얻은 features를 바탕으로 self-attention mechanism을 활용하여 feature들의 관계성을 파악하여 중요 time point를 포착하여 성능 개선.

- Dataset: MHEALTH, UTD-MHAD, USC-HAD, WHARF, WISDM

참고 논문 요약

[9] Dilated causal convolution with multi-head self-attention for sensor human activity recognition. 2021, Neural Computing and Applications

- Dilate causal convolution, multi-head self-attention

- 기존 연구의 한계점: Long short-term memory를 가진 RNN 모델은 sequential 학습이 가능하지만, 연산의 병렬화가 안된다는 단점을 가짐.

- Approach: 1-D Convolution과 multi-head attention mechanism을 활용하여 연산의 병렬화를 가능하게 하고 sequential modeling이 가능하게 함.

- Dataset: Ordonez smart home, Kasteren smart home, UCIHAR

[10] Human Activity Recognition by Using Different Deep Learning Approaches for Wearable Sensors. 2021, Neural Processing Letters

- CNN, Convolutional LSTM, 3-D CNN fed by Convolutional LSTM

- 기존 연구의 한계: X

- Approach: 3-D CNN에 Convolutional LSTM에서 얻은 features를 활용하여 성능 개선.

- Dataset: Activity recognition from accelerometer data using wearable device dataset. 2011, Pers Ubiquitous Compute

참고 논문 요약

[11] A deep learning approach for human activities recognition from multimodal sensing devices. 2020, IEEE Access

- CNN + BLSTM

- 기존 연구의 한계: Large variability를 가진 complex activities를 잘 포착해내기 어려움

- Approach: Multi-channel CNN + BLSTM을 사용하여 input sensor data의 features를 추출하고 forward와 backward sequence 정보를 모두 고려하여 성능 개선.

- Dataset: WISDM2019, MHEALTH

실험 결과

✓ Metrics: ACC, Precision, Recall, F1-score, Confusion Matrix 포함할 것

Method	Dataset	ACC	Precision	Recall	F1-Score	Flops	Param Size
[1]	UCIHAR						
[2]	UCIHAR						
[3]	UCIHAR						
[4]	UCIHAR						
[5]	UCIHAR						
[6]	UCIHAR						
Proposed	UCIHAR						
[1]	UniMiB						
[2]	UniMiB						
[3]	UniMiB						
[4]	UniMiB						
[5]	UniMiB						
[6]	UniMiB						
Proposed	UniMiB						
[1]	WISDM						
[2]	WISDM						
[3]	WISDM						
[4]	WISDM						
[5]	WISDM						
[6]	WISDM						
Proposed	WISDM						

실험 결과

✓ Metrics: ACC, Precision, Recall, F1-score, Confusion Matrix 포함할 것

Method	Dataset	ACC	Precision	Recall	F1-Score	Flops	Param Size
[1]	WISDM2019						
[2]	WISDM2019						
[3]	WISDM2019						
[4]	WISDM2019						
[5]	WISDM2019						
[6]	WISDM2019						
Proposed	WISDM2019						
[1]	KU-HAR						
[2]	KU-HAR						
[3]	KU-HAR						
[4]	KU-HAR						
[5]	KU-HAR						
[6]	KU-HAR						
Proposed	KU-HAR						
[1]	OPPORTUNITY						
[2]	OPPORTUNITY						
[3]	OPPORTUNITY						
[4]	OPPORTUNITY						
[5]	OPPORTUNITY						
[6]	OPPORTUNITY						
Proposed	OPPORTUNITY						

Single Layer Neural Network for Real-time HAR

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
5th October, 2021

To do list

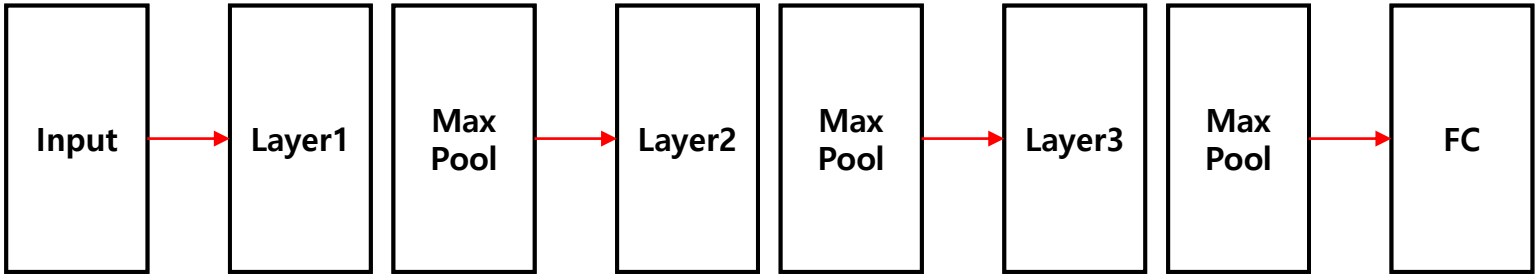
1. 분류 측면에서 다른 클래스와의 오분류 패턴을 확인하기 위해 Confusion Matrix 제공
2. DanHAR을 Backbone으로 하였을 때, 실험을 진행, Layer 컬럼은 생략

비교 모델

1. 기존 3 Layer 모델

2. DanHAR 모델

①



②

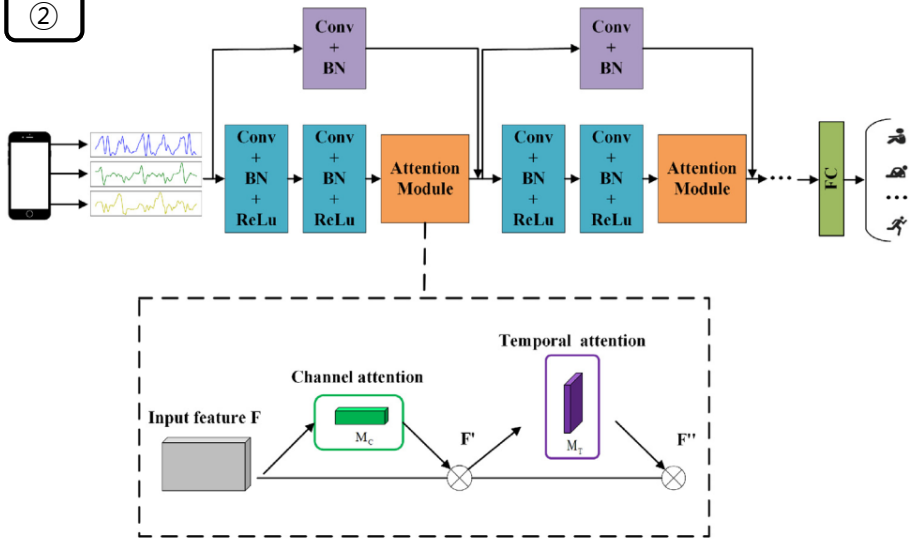


Fig. 1. Overview of the proposed Dual Attention Network.

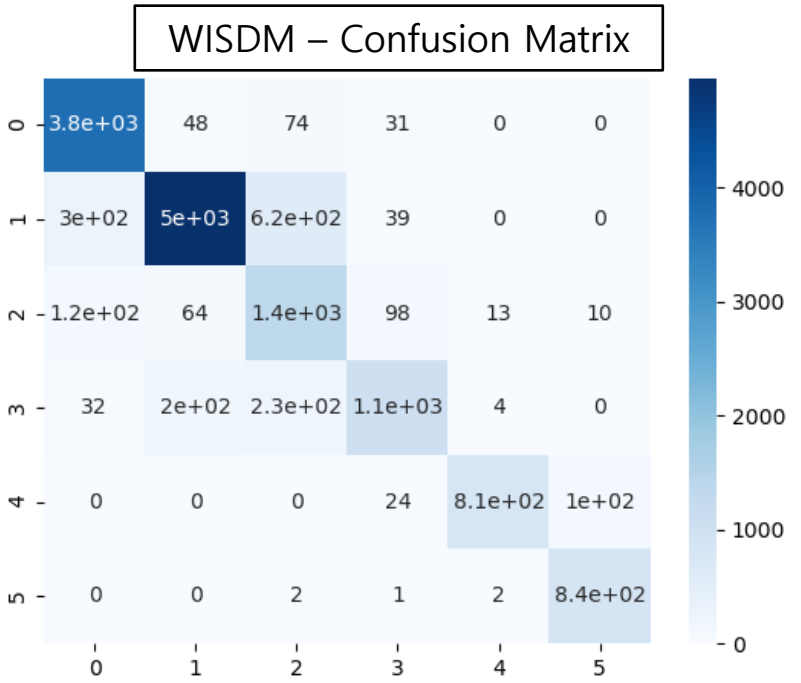
Table 2
Brief description for each backbone.

Layer	Backbone			
	Standard CNN	Att-based CNN	Residual network	Att-based residual network
Layer1	*Conv*, 128 *Conv*, 128	*Conv*, 128 *Conv*, 128 *AttModule*	[*Conv*, 128 *Conv*, 128]	[*Conv*, 128 *Conv*, 128 *AttModule*]
Layer2	*Conv*, 256 *Conv*, 256	*Conv*, 256 *Conv*, 256 *AttModule*	[*Conv*, 256 *Conv*, 256]	[*Conv*, 256 *Conv*, 256 *AttModule*]
Layer3	*Conv*, 384 *Conv*, 384	*Conv*, 384 *Conv*, 384 *AttModule*	[*Conv*, 384 *Conv*, 384]	[*Conv*, 384 *Conv*, 384 *AttModule*]
AvgPool, FC, Softmax				

실험 결과 정리

✓ DanHAR Backbone 비교 실험 결과 테이블

Method	Dataset	ACC	Flops	Param Size
기존 3-Layer	UCIHAR	92.77	2.7	0.24
DanHAR	UCIHAR	91.35	4.8	0.92
ProxylessNAS	UCIHAR	92.81	1.4	0.16
FBNet	**UCIHAR**	**93.01**	**2.7**	**0.46**
DARTs	UCIHAR	91.89	2.9	0.51
기존 3-Layer	UniMiB	88.84	2.3	0.26
DanHAR	**UniMiB**	**90.58**	**5.4**	**0.92**
ProxylessNAS	UniMiB	88.62	1.7	0.16
FBNet	UniMiB	88.48	3.5	0.77
DARTs	UniMiB	87.54	2.7	0.29
기존 3-Layer	WISDM	84.67	1	0.16
DanHAR	WISDM	86.39	1.8	0.44
ProxylessNAS	**WISDM**	**88.21**	**0.5**	**0.07**
FBNet	WISDM	86.62	1.2	0.22
DARTs	WISDM	86.10	0.6	0.08
기존 3-Layer	WISDM2019	88.42	3.4	0.33
DanHAR	WISDM2019	89.17	7.3	0.92
ProxylessNAS	WISDM2019	87.75	2.3	0.16
FBNet	WISDM2019	88.53	4.3	0.24
DARTs	**WISDM2019**	**90.89**	**3.5**	**0.26**



0 : WALKING
1 : WALKING_UPSTAIRS
2 : WALKING_DOWNSTAIRS
3 : SITTING
4 : STANDING
5 : LAYING

실험 결과 정리

1. 대부분의 데이터셋에서 Attention module 사용한 DanHAR 기반 모델이 기존 3-Layer 모델 보다 조금 더 좋은 성능 결과
2. UCI HAR 경우에는 기존 3-Layer 모델이 DanHAR보다 더 좋은 성능 결과
3. 따라서, 데이터셋에 따라 적합한 모델 구조가 다를 수 있음

Single Layer Neural Network for Real-time HAR

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
28th September, 2021

To do list

1. 위즈덤 데이터에 대한 오분류 패턴 파악

WISDM 데이터 구성

✓ 데이터 구성

1. X 축 : 다리의 수평적 움직임 포착
2. Y 축 : 상하 움직임 포착
3. Z 축 : 다리의 앞뒤 움직임 포착

✓ 클래스 별 정확도 결과 :

1. Upstairs와 Downstairs를 잘 구분하지 못함

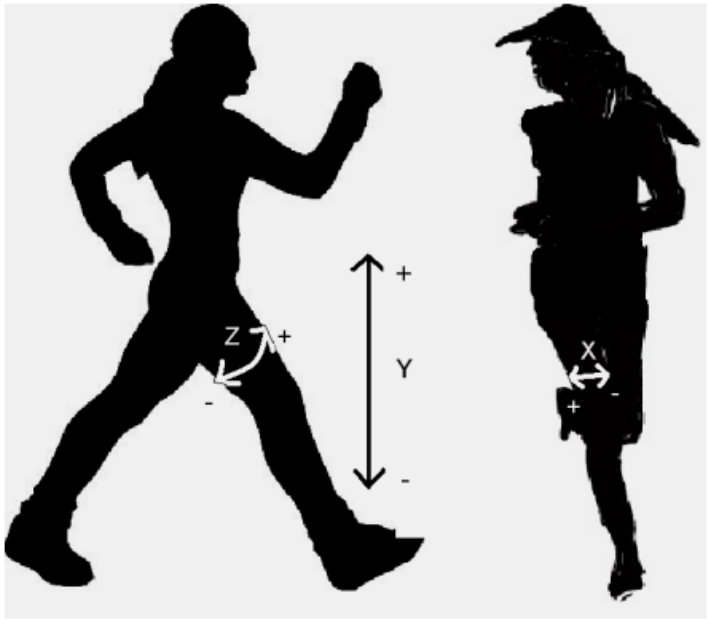


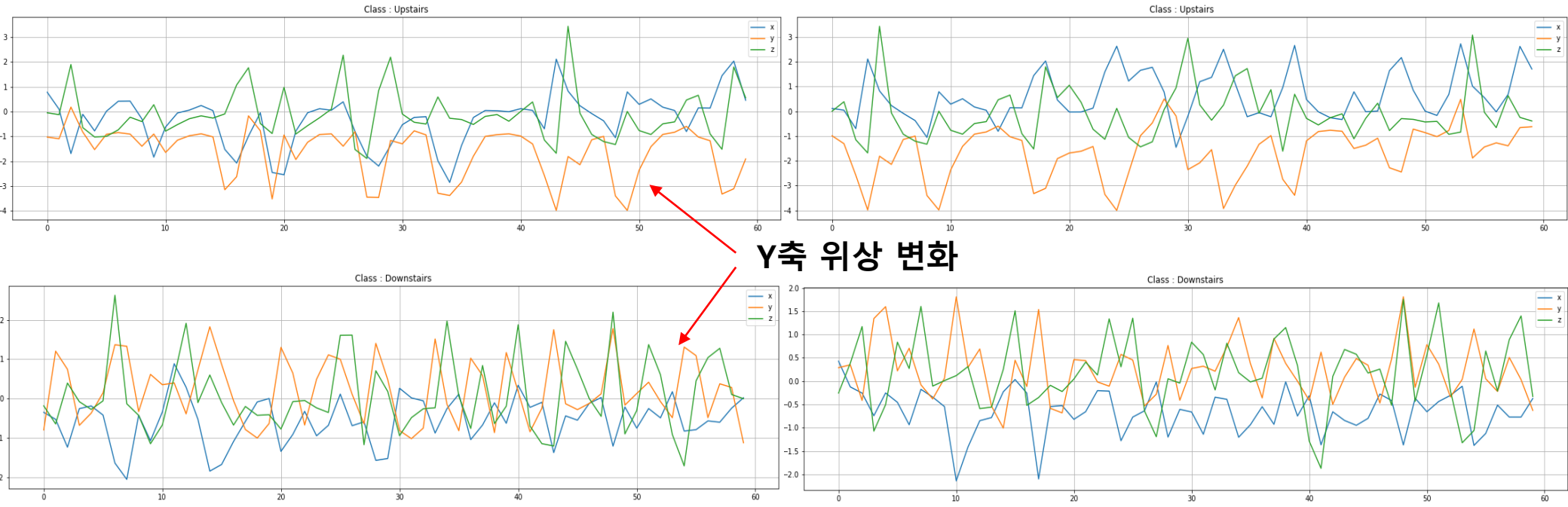
Figure 1: Axes of Motion Relative to User

Class	Accuracy	Num of Instance
Jogging	0.87	3918
Walking	0.85	5918
Upstairs	**0.62**	1679
Downstairs	**0.68**	1545
Sitting	0.89	932
Standing	0.89	847

WISDM 데이터 구성

1. **Upstairs & Downstairs 데이터 차이**

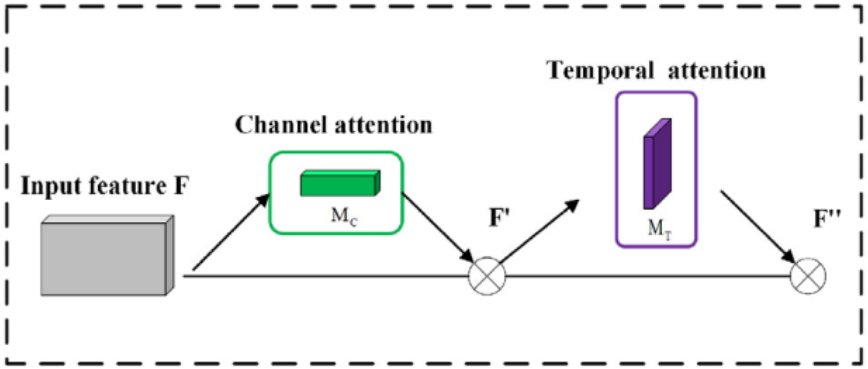
- 두 가지 패턴의 데이터에서 Y 축 위상의 차이가 두드러짐
- Upstairs : Y축이 X축보다 낮게 분포
- Downstairs : Y축이 X축보다 높게 분포
- 데이터의 채널 축에서의 변화를 포착하는 것이 중요



실험

[Applied Soft Computing 2021] DanHAR: Dual Attention Network for multimodal human activity recognition using wearable sensors

1. **채널 축과 Temporal 축에서 피쳐를 잘 포착하기 위해 attention module 활용.**
2. **Attention module을 기반으로 성능 개선의 여지 포착**



Class	Accuracy	Accuracy	Num of Instance
Jogging	0.87	0.89	3918
Walking	0.85	0.89	5918
Upstairs	**0.62**	**0.70**	1679
Downstairs	**0.68**	**0.76**	1545
Sitting	0.89	0.88	932
Standing	0.89	0.89	847

Single Layer Neural Network for Real-time HAR

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
14th September, 2021

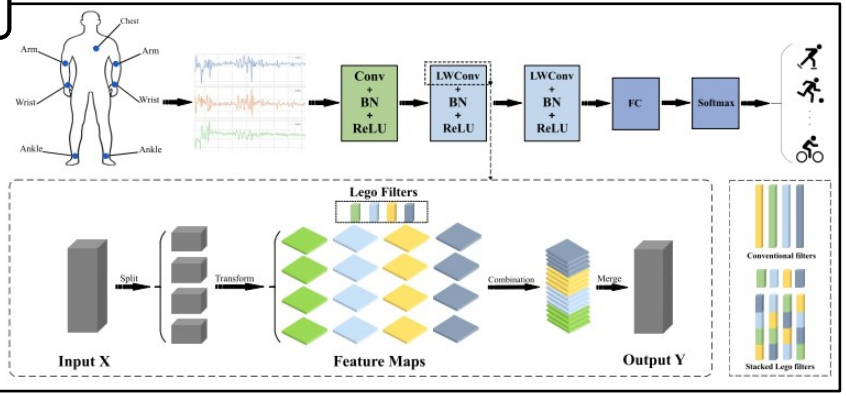
To do list

1. 기존 논문에서 HAR 모델을 보통 3~5개의 layer를 사용한다는 내용 추가
2. Hyperparameter를 늘리는 실험으로 operator의 수를 늘려서 실험
3. 핸드폰의 latency는 포기
4. Search space의 각 operator에 대한 Complexity를 제공

기존 HAR 연구의 모델 구조

1. [IEEE Sensors, 2021] TANG et al. : 3 Conv Layer
2. [Applied Science, 2020]Peppas. et al. : 2 Conv Layer
3. [IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, 2020]Eunji Kim
   : 3 Conv Layer
4. [IEEE Access, 2019]C. Xu et al. : 4 Conv Layer
5. [IEEE Access, 2018]Rui Xi. et al. : 3 Conv Layer + 2 SRU Layer

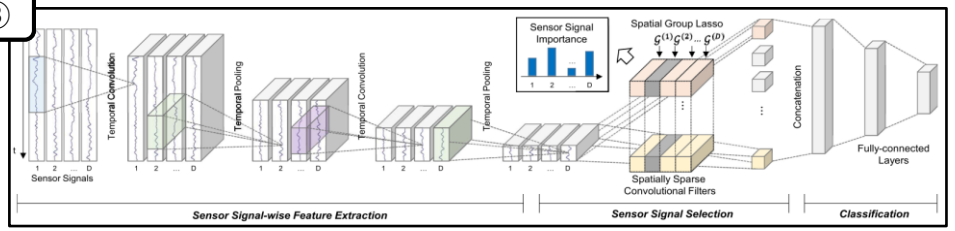
①



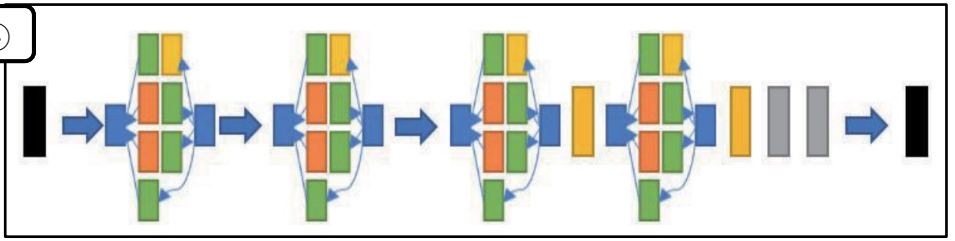
②

Layer Name	Kernel Size	Stride	Hidden Units
1D conv ReLU 1	12 × 3	1	-
1D Max Pooling 1	3 × 1	3	-
1D conv ReLU 2	12 × 192	1	-
1D Max Pooling 1	3 × 1	3	-
Flatten	-	-	-
Concat	-	-	-
FC ReLU	-	-	512
Softmax	-	-	6

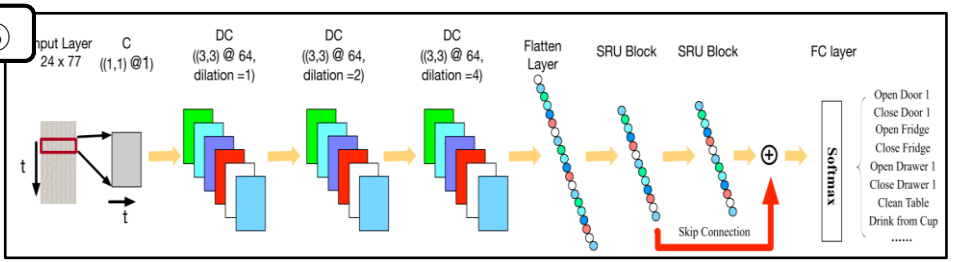
③



④



⑤



Search Space의 Operators

✓ **Complexity 비교 : Module) ④ ≥ ⑤ , Conv) ① > ③ ≥ ②**

Input Channel : M, Output Channel : N, Kernel Size : K, Input & Output Length : T_I & T_O

1. **Standard Convolution**
   - **Computational Complexity : $O(K * M * T_O * N)$**
2. **Depthwise Seperable Convolution**
   - **Computational Complexity : $O((K+N) * T_O * M)$**
3. **Mixed Convolution**
   - **j-th Group Input & Output Channel Size : M_j & N_j, j-th Group Kernel Size: K_j {3, 5, 7, 9}**
   - **Computational Complexity : $O(\sum(K_j + N_j) * T_O * M_j)$**
4. **Dynamic Convolution(구현 이슈로 보류) : Same Output but Big Model size**
   - **Temperature Constant : r**
   - **Computational Complexity : $O(K * M * T_O * N) + O(\pi)$**
   - **Attention Module : $O(\pi)$: $T_I * M + M^2/r + M*K/r$**
5. **Convolutional Block Attention Module(CBAM)**
   - **Computational Complexity : $O(\pi_c) + O(\pi_s)$**
   - **Channel Attention Module : $O(\pi_c)$**
   - **Spatial Attention Module : $O(\pi_s)$**

Layer 별 탐색 공간

✓ **3번 Mixed Convolution과 5번 Convolutional Attention Block Module(CBAM)은 연산자의 특성으로 인해 1 Layer에 적용 불가하여 2~3 Layer부터 탐색**

1. **1-Layer**
   - Opertors : {Standard Conv, Depthwise Seperable Conv}
   - Dilation : {1, 2}
   - Kernel Size : {3, 5, 7, 9}
   - 총 16 개
2. **2~3 Layer**
   - Opertors : {Standard Conv, Depthwise Seperable Conv, CBAM}
   - Dilation : {1, 2}
   - Kernel Size : {3, 5, 7, 9}
   - Operator : {3x5_Mixed Conv, 3x5x7_Mixed Conv, 3x5x7x9_Mixed Conv, 3x5x7x9x11_Mixed Conv}
   - 총 31 개

✓ **Flops 계산상의 오류 가능**
- 논문에서 제공한 Computation Cost에 비해 낮은 Flops 결과
- Customized 연산자로 인해 기존 Flops 계산 라이브러상으로는 누락되는 값 존재 가능.
- CBAM operator

실험 결과 정리

✓ 3 Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Flops	Param Size	Layer1	Layer2	Layer3
Manual	UCIHAR	92.77	2.7	0.24	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	UCIHAR	91.55	0.3	0.17	9x1_DepthConv	3x1_CBAMBlock	3x1_DilatedDepthConv
RL(MnasNet)	UCIHAR	89.41	0.2	0.17	9x1_DepthConv	7x1_CBAMBlock	7x1_CBAMBlock
FBNet	UCIHAR	91.75	0.8	0.2	7x1_DilatedDepthConv	5x1_DilatedDepthConv	3x1_DilatedCBAMBlock
DARTs	UCIHAR	**93.89**	0.9	0.19	**9x1_DilatedConv**	**7x1_DilatedCBAMBlock**	**5x1_DilatedConv**
Manual	UniMiB	**94.27**	2.3	0.26	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	UniMiB	90.94	0.3	0.21	9x1_DepthConv	7x1_CBAMBlock	3x5x7x9x11_MixedConv
RL(MnasNet)	UniMiB	91.88	0.3	0.21	7x1_DilatedDepthConv	5x1_CBAMBlock	3x5x7_MixedConv
FBNet	UniMiB	91.05	1.9	0.24	7x1_DilatedDepthConv	3x1_DepthConv	7x1_DilatedConv
DARTs	UniMiB	90.94	0.5	0.21	5x1_DepthConv	3x5_MixedConv	3x1_DepthConv
Manual	WISDM	84.67	1	0.16	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	WISDM	83.38	0.3	0.11	7x1_DilatedConv	3x5x7x9x11_MixedConv	7x1_DilatedCBAMBlock
RL(MnasNet)	WISDM	**85.29**	0.2	0.11	**9x1_DilatedDepthConv**	**3x5_MixedConv**	**3x1_CBAMBlock**
FBNet	WISDM	84.28	0.8	0.16	7x1_DilatedConv	5x1_DilatedDepthConv	5x1_DilatedDepthConv
DARTs	WISDM	82.43	0.3	0.11	5x1_DilatedConv	7x1_DilatedDepthConv	3x5x7x9_MixedConv
Manual	WISDM2019	88.42	3.4	0.33	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	WISDM2019	78.99	0.4	0.27	3x1_DepthConv	3x1_DilatedCBAMBlock	9x1_CBAMBlock
RL(MnasNet)	WISDM2019	83.6	0.5	0.27	5x1_DilatedDepthConv	9x1_DilatedCBAMBlock	3x5x7x9_MixedConv
FBNet	WISDM2019	**88.85**	1.8	0.29	**9x1_DepthConv**	**7x1_CBAMBlock**	**5x1_DilatedCBAMBlock**
DARTs	WISDM2019	87.89	1.1	0.28	3x1_Conv	3x5_MixedConv	3x1_Conv

Single Layer Neural Network for Real-time HAR

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
7th September, 2021

To do list

1. Single, 2, 3 Layer에 대한 실험 결과 완성
2. 목표 설정

실험 결과 정리

✓ Single Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Latency	Flops	Param Size	Layer1
Manual	UCIHAR	89.61	2.438	1.8	1.42	7x1_Conv
ProxylessNAS	UCIHAR	90.8	2.356	1.8	1.42	7x1_DilatedConv
RL(MnasNet)	**UCIHAR**	**92.29**	**2.36**	**1.9**	**1.42**	**9x1_DilatedConv**
FBNet	**UCIHAR**	**92.29**	**2.36**	**1.9**	**1.42**	**9x1_DilatedConv**
DARTs	UCIHAR	91.38	2.356	1.8	1.42	7x1_DilatedConv
Manual	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv
ProxylessNAS	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv
RL(MnasNet)	**UniMiB**	**84.28**	**2.685**	**1.8**	**1.68**	**5x1_DilatedConv**
FBNet	UniMiB	80.07	2.729	1.8	1.68	3x1_DilatedConv
DARTs	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv
Manual	**WISDM**	**85.95**	**1.128**	**0.7**	**0.66**	**7x1_Conv**
ProxylessNAS	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv
RL(MnasNet)	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv
FBNet	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv
DARTs	WISDM	85.58	1.124	0.8	0.66	9x1_Conv
Manual	**WISDM2019**	**82.74**	**4.025**	**2.7**	**2.2**	**7x1_Conv**
ProxylessNAS	WISDM2019	78.03	4.063	2.3	2.2	5x1_DepthConv
RL(MnasNet)	WISDM2019	78.03	4.063	2.3	2.2	5x1_DepthConv
FBNet	WISDM2019	77.82	4.063	2.3	2.2	5x1_DepthConv
DARTs	WISDM2019	78.67	4.006	2.3	2.2	9x1_DepthConv

실험 결과 정리

✓ 2 Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Latency	Flops	Param Size	Layer1	Layer2
Manual	UCIHAR	**93.92**	1.179	2.1	0.53	**7x1_Conv**	**7x1_Conv**
ProxylessNAS	UCIHAR	93.62	**0.932**	1.6	0.52	**5x1_DilatedConv**	**5x1_DilatedConv**
RL(MnasNet)	UCIHAR	93.72	1.136	1.1	0.5	9x1_DilatedConv	9x1_DepthConv
FBNet	UCIHAR	93.28	1.056	1.1	0.5	9x1_DilatedConv	5x1_DepthConv
DARTs	UCIHAR	91.68	1.052	1	0.5	7x1_DilatedConv	5x1_DepthConv
Manual	UniMiB	92.1	1.368	2.2	0.59	7x1_Conv	7x1_Conv
ProxylessNAS	UniMiB	87.24	1.064	0.7	0.56	5x1_Conv	Zero
RL(MnasNet)	UniMiB	**92.75**	1.115	0.8	0.57	**9x1_DilDepthConv**	**5x1_DilDepthConv**
FBNet	UniMiB	90.43	1.312	2.5	0.6	9x1_DilDepthConv	9x1_Conv
DARTs	UniMiB	84.49	**1.042**	0.8	0.56	**7x1_Conv**	**Zero**
Manual	WISDM	**85.67**	0.903	0.9	0.26	**7x1_Conv**	**7x1_Conv**
ProxylessNAS	WISDM	83.31	**0.665**	**0.3**	**0.23**	**9x1_Conv**	**Zero**
RL(MnasNet)	WISDM	84.77	0.825	0.5	0.25	7x1_DepthConv	3x1_Conv
FBNet	WISDM	**85.67**	0.903	0.9	0.26	7x1_Conv	7x1_Conv
DARTs	WISDM	83.31	**0.665**	**0.3**	**0.23**	**9x1_Conv**	**Zero**
Manual	WISDM2019	**89.82**	1.876	1.1	0.76	**7x1_Conv**	**7x1_Conv**
ProxylessNAS	WISDM2019	83.49	**1.314**	**0.8**	0.76	**5x1_DilDepthConv**	**Zero**
RL(MnasNet)	WISDM2019	86.28	1.571	1.7	0.77	9x1_DilDepthConv	3x1_DilatedConv
FBNet	WISDM2019	84.46	1.545	1.7	0.77	5x1_DilDepthConv	3x1_DilatedConv
DARTs	WISDM2019	86.28	1.69	1.1	0.76	9x1_DepthConv	7x1_DilDepthConv

실험 결과 정리

✓ 3 Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Latency(S9)	Flops	Param Size	Layer1	Layer2	Layer3
Manual	UCIHAR	92.77	0.936	2.7	0.24	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	UCIHAR	93.38	**0.598**	**0.6**	**0.17**	**9x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UCIHAR	92.67	1.068	2.4	0.22	9x1_DilatedConv	9x1_Conv	3x1_DilatedConv
FBNet	UCIHAR	**93.82**	0.984	0.9	0.18	**9x1_DilatedConv**	**5x1_DepthConv**	**9x1_DilDepthConv**
DARTs	UCIHAR	93.18	0.953	1.6	0.2	9x1_DilatedConv	5x1_Conv	5x1_DilDepthConv
Manual	UniMiB	**94.27**	1.267	2.3	0.26	**7x1_Conv**	**7x1_Conv**	**7x1_Conv**
ProxylessNAS	UniMiB	90.79	**0.587**	**0.5**	**0.2**	**9x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UniMiB	92.97	1.004	1.7	0.24	9x1_DilatedConv	5x1_Conv	3x1_DilatedConv
FBNet	UniMiB	91.59	0.982	1.9	0.24	9x1_DilDepthConv	7x1_DilatedConv	3x1_DilatedConv
DARTs	UniMiB	86.66	1.06	0.9	0.21	3x1_DepthConv	3x1_Conv	Zero
Manual	WISDM	84.67	1.244	1	0.16	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	WISDM	83.79	**0.789**	**0.2**	**0.1**	**7x1_Conv**	**Zero**	**Zero**
RL(MnasNet)	WISDM	**85.08**	0.912	0.4	0.12	**9x1_DilatedConv**	**3x1_DilatedConv**	**Zero**
FBNet	WISDM	84.8	1.108	0.4	0.12	7x1_Conv	5x1_DepthConv	9x1_Conv
DARTs	WISDM	79.86	0.951	0.2	0.11	7x1_DepthConv	3x1_DepthConv	Zero
Manual	WISDM2019	**88.42**	1.366	3.4	0.33	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	WISDM2019	87.56	0.805	1	0.29	9x1_DilDepthConv	Zero	7x1_DilatedConv
RL(MnasNet)	WISDM2019	85.42	**0.793**	**0.6**	**0.28**	**7x1_DilDepthConv**	**Zero**	**3x1_DilatedConv**
FBNet	WISDM2019	87.67	1.27	1.5	0.29	5x1_DepthConv	9x1_DilDepthConv	3x1_DilatedConv
DARTs	WISDM2019	86.82	0.975	0.8	0.27	3x1_Conv	5x1_DilDepthConv	Zero

실험 결과 분석 및 정리

1. 정확도 성능 지표상 최소 2개 이상의 layer를 구성하였을 때, 성능의 차이가 적음.
2. Layer의 개수가 많아질수록 operator에 따른 latency의 차이가 나타남.

   Ex) Single Layer Model에서는 각 모델 간의 latency의 차이가 작지만, 3 Layer Model에서는 모델 간의 latency 차이가 큼.
3. Layer의 개수에 따라 전체 latency에서 Fully Connected Layer의 latency 비중이 다름.

   Ex) Single Layer : FC 66%~90%

   2 Layer : FC 30% ~ 60%

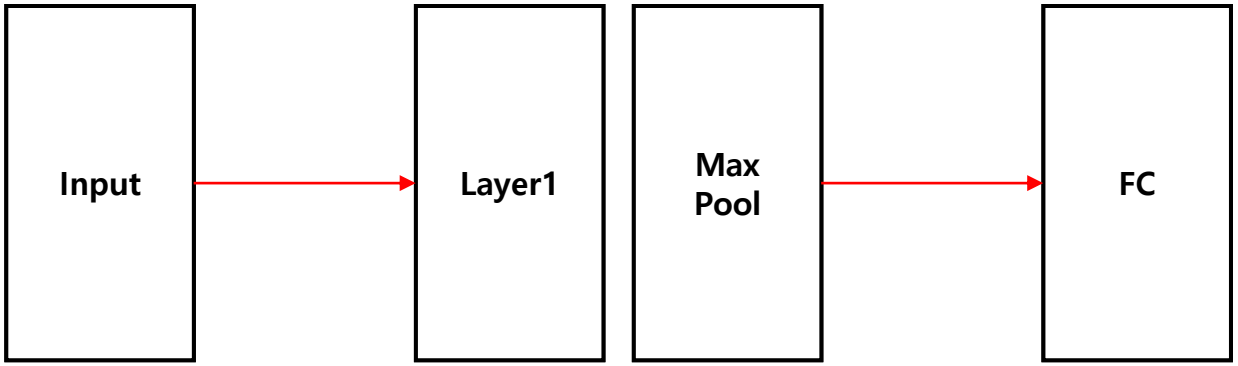
   3 Layer : FC 15% ~ 30%

Single Layer Neural Network for Real-time HAR

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com
31th August, 2021

To do list

1. Single Layer에 대한 실험 결과
2. 기존 3가지와 추가적으로 유명한 2가지 알고리즘 실험(Fbnet, DARTs)
3. 그 중 Baseline 설정
4. Search Space 늘려서 실험 8 -> 16



실험 결과 정리

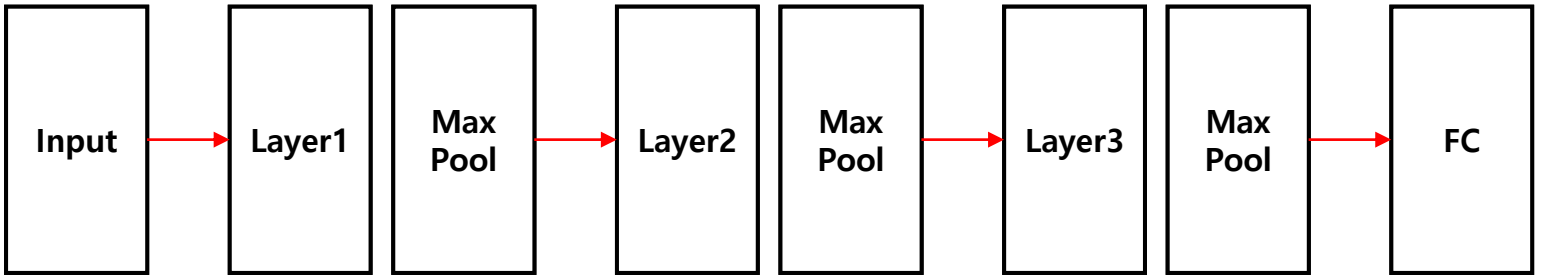
✓ Single Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Latency	Flops	Param Size	Layer1, Probability
Manual	UCIHAR	89.61	2.438	1.8	1.42	7x1_Conv
ProxylessNAS	UCIHAR	90.8	2.356	1.8	1.42	7x1_DilatedConv, 0.101
RL(MnasNet)	**UCIHAR**	**92.29**	**2.36**	**1.9**	**1.42**	**9x1_DilatedConv, 0.895**
FBNet	**UCIHAR**	**92.29**	**2.36**	**1.9**	**1.42**	**9x1_DilatedConv, 0.076**
DARTs	UCIHAR	91.38	2.356	1.8	1.42	7x1_DilatedConv, 0.096
Manual	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv
ProxylessNAS	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv, 0.080
RL(MnasNet)	**UniMiB**	**84.28**	**2.685**	**1.8**	**1.68**	**5x1_DilatedConv, 0.136**
FBNet	UniMiB	80.07	2.729	1.8	1.68	3x1_DilatedConv, 0.076
DARTs	UniMiB	82.89	2.729	1.9	1.68	7x1_Conv, 0.080
Manual	**WISDM**	**85.95**	**1.128**	**0.7**	**0.66**	**7x1_Conv**
ProxylessNAS	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv, 0.252
RL(MnasNet)	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv, 0.936
FBNet	WISDM	85.57	1.127	0.7	0.66	7x1_DepthConv, 0.097
DARTs	WISDM	85.58	1.124	0.8	0.66	9x1_Conv, 0.162
Manual	**WISDM2019**	**82.74**	**4.025**	**2.7**	**2.2**	**7x1_Conv**
ProxylessNAS	WISDM2019	78.03	4.063	2.3	2.2	5x1_DepthConv, 0.090
RL(MnasNet)	WISDM2019	78.03	4.063	2.3	2.2	5x1_DepthConv, 0.481
FBNet	WISDM2019	77.82	4.063	2.3	2.2	5x1_DepthConv, 0.093
DARTs	WISDM2019	78.67	4.006	2.3	2.2	9x1_DepthConv, 0.090

실험 결과 정리

✓ 3 Layer 비교 실험 결과 테이블

Method	Dataset	ACC	Latency	Flops	Param Size	Layer1	Layer2	Layer3
Manual	UCIHAR	92.09	1.22	2.2	0.23	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**UCIHAR**	**92.56**	**0.59**	**0.5**	**0.17**	**7x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UCIHAR	92.73	0.87	1.4	0.2	9x1_DilatedConv	3x1_DilatedConv	5x1_DilatedConv
Manual	UniMiB	93.98	1.267	2.3	0.26	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**UniMiB**	**84.56**	**0.573**	**0.34**	**0.2**	**5x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UniMiB	93.04	0.871	1.9	0.25	5x1_DilatedConv	5x1_DilatedConv	7x1_DilatedConv
Manual	WISDM2019	88.42	1.366	3.4	0.33	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**WISDM2019**	**83.17**	**0.7**	**0.6**	**0.27**	**5x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	**WISDM2019**	**83.17**	**0.7**	**0.6**	**0.27**	**5x1_DilatedConv**	**Zero**	**Zero**



실험 결과 분석 및 정리

1. Single Layer 상황에서는 Operation에 따른 latency 차이가 크지 않고, 정확도의 차이가 드러남.
2. 연구 목표 : Latency는 고정된 factor이고 Accuracy를 목표
3. Single layer 상황에서 NAS의 필요성을 주장하기 부족
   - 그러나 NAS가 효과적인 operator를 찾는다는 것은 기존 연구들에서 실험적 증거가 제시됨
4. 효과적인 Operator를 제안
5. Proposed operator의 효과성을 NAS 알고리즘을 통해 실험적 증거 제시
   - NAS 알고리즘을 통해 proposed operator가 선택된다는 것을 실험적 증거 제시

Searching Efficient HAR Neural Networks

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com

24th August, 2021

실험 결과 정리

✓ 비교 실험 알고리즘

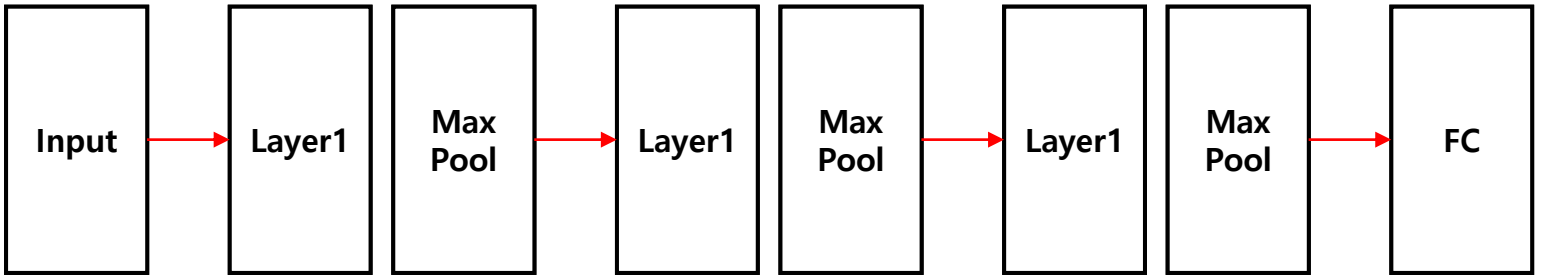
1. Manaul : 수동으로 설계된 모델
2. ProxylessNAS : Differentiable NAS기반으로 Latency와 정확도를 목적 함수로 설계하여 탐색된 모델
3. RL(MnasNet) : 강화학습 기반으로 Latency와 정확도를 목적 함수로 설계하여 탐색된 모델 (* MnasNet의 강화학습 알고리즘 부분만을 가져와서 적용.)

실험 결과 정리

✓ 비교 실험 결과 테이블

- **Zero Layer :** 입력 받은 input을 다음 layer로 그대로 전달하는 layer

Method	Dataset	Target	ACC	Latency	Flops	Param Size	Layer1	Layer2	Layer3
Manual	UCIHAR	s9	92.09	1.22	2.2	0.23	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**UCIHAR**	**s9**	**92.56**	**0.59**	**0.5**	**0.17**	**7x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UCIHAR	s9	92.73	0.87	1.4	0.2	9x1_DilatedConv	3x1_DilatedConv	5x1_DilatedConv
Manual	UniMiB	s9	93.98	1.267	2.3	0.26	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**UniMiB**	**s9**	**84.56**	**0.573**	**0.34**	**0.2**	**5x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	UniMiB	s9	93.04	0.871	1.9	0.25	5x1_DilatedConv	5x1_DilatedConv	7x1_DilatedConv
Manual	WISDM2019	s9	88.42	1.366	3.4	0.33	7x1_Conv	7x1_Conv	7x1_Conv
ProxylessNAS	**WISDM2019**	**s9**	**83.17**	**0.7**	**0.6**	**0.27**	**5x1_DilatedConv**	**Zero**	**Zero**
RL(MnasNet)	**WISDM2019**	**s9**	**83.17**	**0.7**	**0.6**	**0.27**	**5x1_DilatedConv**	**Zero**	**Zero**



Searching Efficient HAR Neural Networks

CAU Artificial Intelligence LAB
임원선
costor107@gmail.com

6th August, 2021

1. 효율적으로 동작하는 모델을 탐색하기 위한 방안

✓ SuperNet 구성에서 Dynamic Pooling Layer를 적용

기존 연구) NAS SuperNet 구성에서는 고정된 Pooling(Reduction)층과 Pooling size

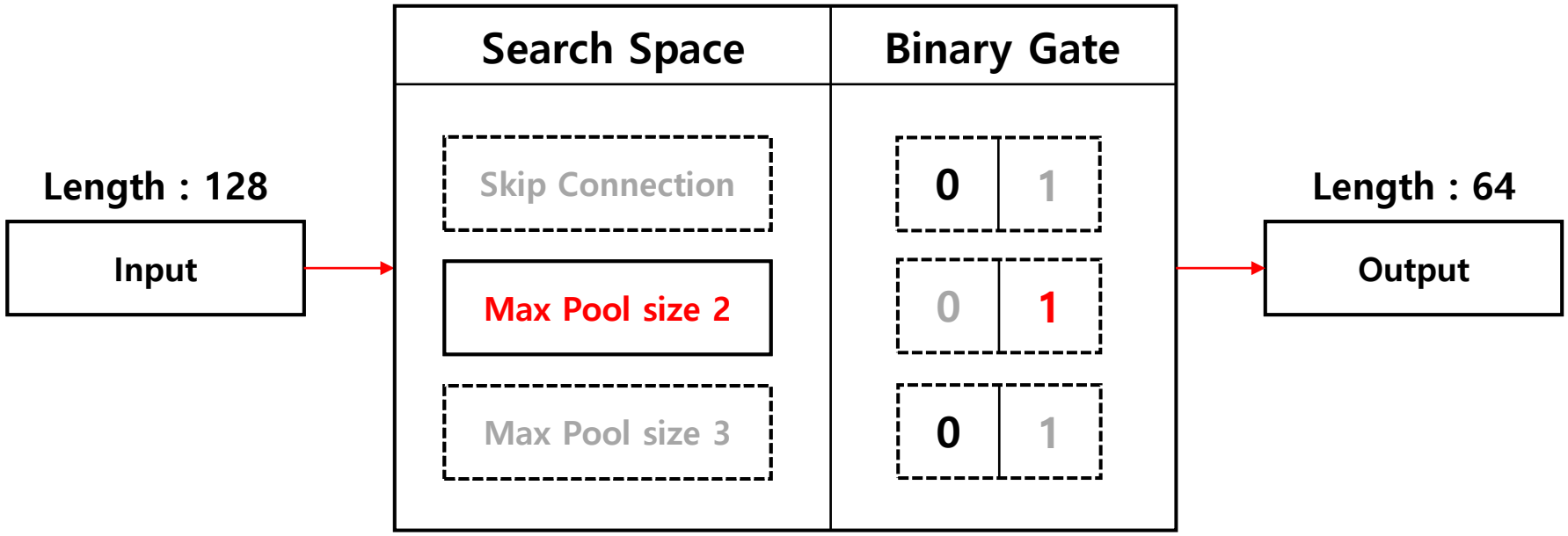
1. Pooling Layer를 탐색 가능하게 함으로써 탐색 과정에서 계산량과 연관되는 Feature map을 줄일 수 있는 가능성을 부여
2. HAR sensor 데이터의 1차원 특성상 넓은 인식 영역이 성능에도 좋은 영향을 보임
   Ex) 실험에서 Cross-Entropy만을 Loss 함수로 설정한 후 탐색하였을 때에도 Standard Conv보다 Dilated Conv이 좋은 효과를 보임
3. Pooling size가 클수록 뒤쪽 layer에서 작은 kernel size를 사용 가능

- 계산이 필요하지 않은 효율적인 연산자 Skip Connection을 탐색 공간에 같이 구성하여 효율성 유지하면서 정확도 하락 보정

1. 효율적으로 동작하는 모델을 탐색하기 위한 방안

✓ Dynamic Pooling Layer Overview

1. 같은 Layer에 서로 다른 Pooling size를 적용하면 각기 다른 Feature map 사이즈가 나오므로 Binary Gate를 통해 한 step 학습 시 한 개의 Pooling 연산자만 동작하도록 구성



Thank you

참고 문헌

[1] Zela, Arber et al. “Understanding and Robustifying Differentiable Architecture Search.” ICLR, 2020.

Appendix

문제 정의) HAR이 주로 동작하는 모바일 환경에 최적화된 효율적인 모델 구조 탐색

✓ Differentiable-NAS(DARTS)기반의 Accuracy와 Latency를 최적화하는 방식에서의 1 Phase 최적화의 문제점

원인 파악)

1. Loss = Cross-Entropy + λE[latency] 에서 각 term의 scalability가 다름
   -> Latency term이 dominant하면서 구조가 결정되는 경향
   -> 오히려 효율적인 모델 구성에 장점 가능
   -> 또한, 단순히 각 term의 scale을 조절하면 해결 가능한 문제

Appendix

✓ Differentiable-NAS(DARTS)기반의 Accuracy와 Latency를 최적화하는 방식에서의 1 Phase 최적화의 문제점

원인 파악)

2. DARTS기반 최적화 방식의 구조적 문제 : α가 성능 지표를 보장하지 않음

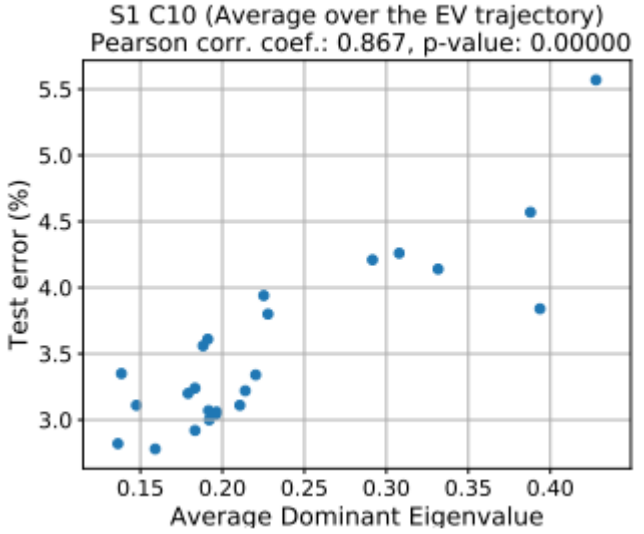
$$\min_{\alpha} \quad \mathcal{L}_{val}(w^*(\alpha), \alpha) \qquad \text{--- (1)}$$
$$\text{s.t.} \quad w^*(\alpha) = \text{argmin}_w \ \mathcal{L}_{train}(w, \alpha) \qquad \text{--- (2)}$$

$$\min_{\alpha} \quad \mathcal{L}_{val}(w^*(\alpha), \alpha) \qquad \text{--- (1) outer optimi}$$
$$\text{s.t.} \quad w^*(\alpha) = \text{argmin}_w \ \mathcal{L}_{train}(w, \alpha) \qquad \text{--- (2) inner optimiz}$$

식(1)을 최적화 하기 전에 식(2)가 최적화 되어야 하는데,
계산비용이 크게 들기 때문에 inner optimization을
식(3)Approximation을 통해 계산 비용을 감소

$$\nabla_\alpha \mathcal{L}_{val}(w^*(\alpha), \alpha)$$
$$\approx \nabla_\alpha \mathcal{L}_{val}(w - \xi \nabla_w \mathcal{L}_{train}(w, \alpha), \alpha) \qquad \text{--- (3)}$$

여기서 구조 파라미터 α가 업데이트할 때 사용하는 $\nabla^2_\alpha \mathcal{L}_{valid}$
에 대한 Hessian Matrix의 eigenvalues의 크기에 따라
생성된 구조의 Test error가 증가한다는 상관 관계를
증명한 연구 결과 보고[1]



Appendix

✓ Differentiable-NAS(DARTS)기반의 Accuracy와 Latency를 최적화하는 방식에서의 1 Phase 최적화의 문제점

원인 파악)

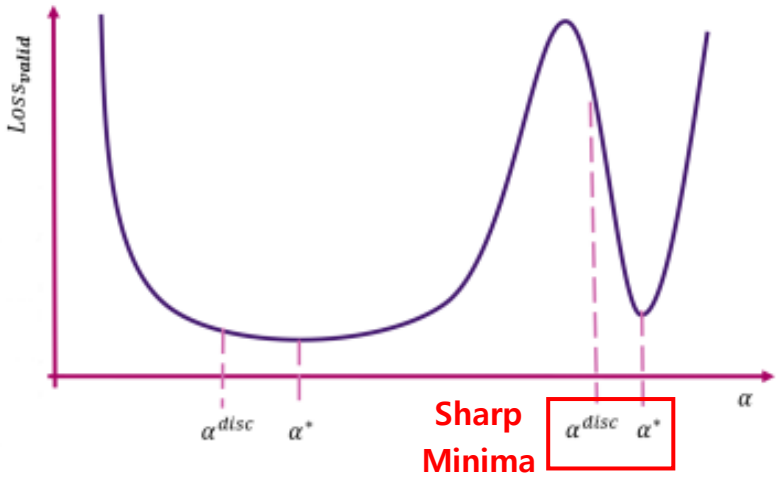
3. 최종 네트워크 구조 선택 과정에서 continuous space의 optimal 구조 파라미터 α^*를 discretize하였을 때, α^{disc} 구조의 loss는 α^* 성능의 주위에 존재

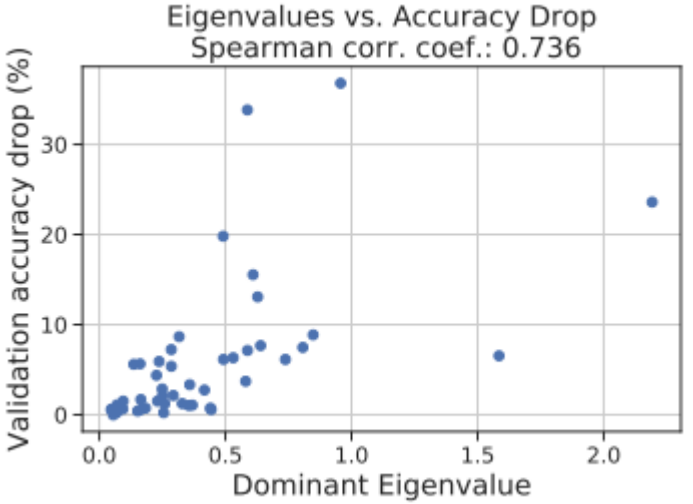
-> α^*가 Sharp minima에 있을 경우 α^{disc} 와의 loss 차이가 크게 나타남

-> Dominant eigenvalue를 α의 discretize에 따른 Loss 변화량 을 확인한 결과에서도 클수록 loss 차이가 크게 나타남

Loss 변화량 : $\mathcal{L}_{valid}(\alpha^{disc}, w^*) - \mathcal{L}_{valid}(\alpha^*, w^*)$

따라서, α가 정확도를 직접적으로 반영하는 지표가 되기 어려운 경우 존재





Appendix

✓ 기존 주장에 대한 고찰

- 결론적으로 현재 목표로 하는 효율적으로 동작하는 모델 탐색에 대한 원인으로는 작용하지 않고 모델의 generalization 문제
- Model Selection 과정에서 고려는 가능
- Latency와 같은 다른 Metric이 포함되었을 때의 영향은 더욱 알기 어려우므로 해당 문제는 다루지 않음

✓ 효율적으로 동작하는 모델을 탐색하기 위한 방안 제시

실험 계획 추진 보고

✓ Search Space의 operator를 Backbone network와 통일하여 실험

✓ Flops의 제한만 두고 layer의 개수에는 제한을 두지 않음.

1주차 실험(10/12)

- 기존 실험에서 Layer의 개수 30까지 실험
- Layer 3 ~ 30
- 전체 데이터셋에 대하여 3회 반복 실험

2주차 실험(10/19)

- 재균이형 Operator 추가하여 Layer 30 실험
- 오분류 패턴 개선 가능성 확인 실험: 1)CNN + LSTM 혼합 모델[1] 실험, 2)통계치 피처 추가 방식[2]
- Real-time을 위한 bottleneck(LSTM) 개선 operator: Dilated Causal Convolution

3, 4주차 실험(10/26, 11/02)

- 새로 발표된 HAR NAS논문[6] 비교 실험.
- 교수님 피드백에 따른 추가 실험 예정.

➢ 다양한 실험에도 불구하고, 개선사항이 선명하지 않을 경우 NAS 시도 자체의 의도를 살려서 연구 보고하는 것도 고려하기 바람.

➢ 즉, S9, S10, A31의 디바이스별 latency와 최적 도출 모델들의 상관관계 등 예전 실험을 보다 심도 있게 추진하여 돌파구를 찾는 실험도 병행.