



이현주

AI Researcher
Software Engineer

Technical Skills

<https://github.com/dlguswn3659>

이현주 (Lee Hyunju)

Seoul, Republic of Korea

Education

Undergraduate Research Internship, Image Lab (Prof. Byungwoo Hong)

ChungAng University

Jul 2020 - Sep 2020

B.S. in Software Engineering

ChungAng University

GPA 4.09/4.50

Mar 2018 - Feb 2023

M.S. in Electrical and Electronic Engineering, Computer Vision Lab (Prof. Bumsub Ham)

Yonsei University

GPA 4.21/4.30

Mar 2024 - Feb 2026



기타 논문 성과

- 국내 학회 (IPIU 2025) 논문 1편 기재 및 포스터 발표.
 - 논문 제목 : GreeZe-TAS: Efficient Greedy Transformer Architecture Search Using Zero-Cost Proxies
 - 저자 : **Hyunju Lee***, Bumsub Ham
- 국제 학회(WACV 2026, h5-index: 131, Google Scholar CV&PR분야 6위) 논문 1편 게재 승인
 - 논문 제목 : GrowTAS: Progressive Expansion from Small to Large Subnets for Efficient ViT Architecture Search
 - 저자 : **Hyunju Lee***, Y.Oh, J.Jeon, D.Baek, Bumsub Ham
- 국제 학회(AAAI 2025, h5-index: 232 , Google Scholar AI분야 4위) 논문 1편 게재
 - 논문 제목 : Efficient Few-Shot Neural Architecture Search by Counting the Number of Nonlinear Functions
 - 저자 : Y.Oh*, **Hyunju Lee**, Bumsub Ham
- 국제 학회(CVPR 2026, h5-index: 450 , Google Scholar CV&PR분야 1위) 논문 1편 제출 ([Under Review](#))
 - 논문 제목 : TAS-LoRA: Transformer Architecture Search with Mixture-of-LoRA Experts
 - 저자 : J.Jeon*, **Hyunju Lee**, Bumsub Ham

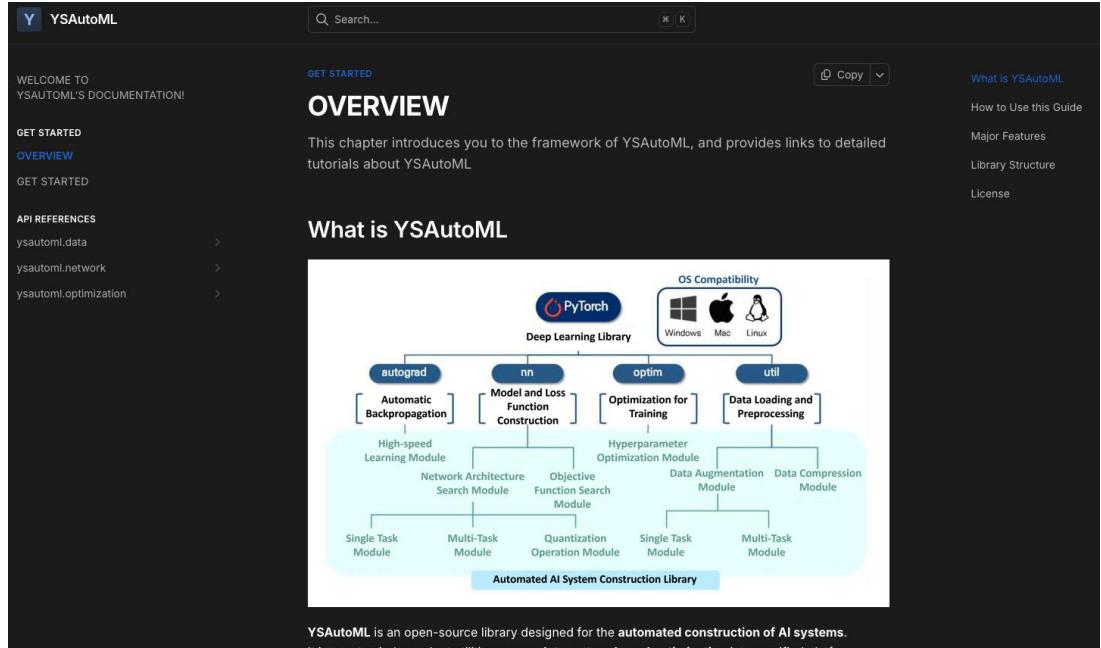
02 Projects

YSAutoML / AI Researcher & Tech Leader

2024.03~2025.12

(오픈소스 AutoML 라이브러리 개발)

- Python(Pytorch) 기반으로 NAS, 데이터 증강, Domain-Specific Batch Normalization(DSBN), Fixed-Point Quantization, Loss Search 등을 하나의 통합 AutoML 파이프라인으로 구현하여, 자동으로 최적 모델 및 학습 전략을 탐색하고 최종 모델을 생성하는 오픈소스 라이브러리를 개발함.
- **Django** 기반 데모 웹 UI를 구축하여, 사용자가 AutoML 기능을 손쉽게 테스트할 수 있는 형태로 **API를 시작화하고 접근성을 높임**.
- **프로젝트의 전체 개발 리드**를 맡아 아키텍처 설계, 기능 구현, 성능 최적화, 코드 품질 관리 등 대부분의 핵심 기능을 직접 개발함.
- **NAS 관련 연구**를 수행하고, 연구 결과를 라이브러리에 적용하여 AutoML 성능을 고도화함.
- 사용자 편의성을 위해 자세하고 체계적인 **Documentation**을 제작하여 설치, 사용법, 데모 절차 등을 명확히 안내함.



Github : <https://github.com/cvlab-yonsei/YonseiAutoML>

Document : <https://dev-julie.gitbook.io/ysautoml>

Demo Video : <https://youtu.be/UJDJXOTxk14>

02 Projects

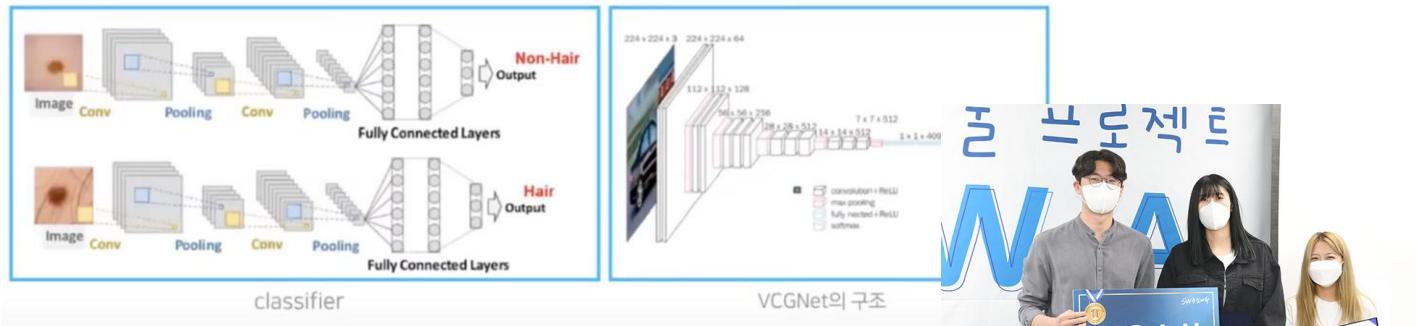
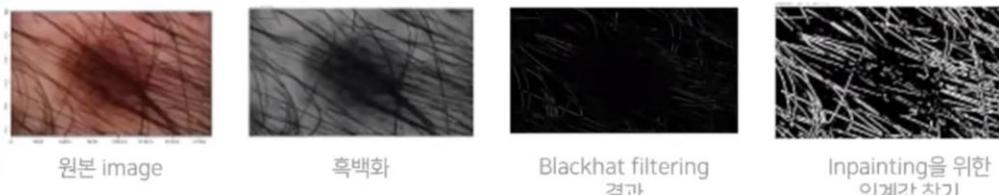
Chung-And University Image Lab / Undergraduate Student

2020.07 – 2020.09, 중앙대학교 이미지연구실

- 병변 이미지를 자동으로 진단하는 프로젝트 진행

병변 이미지의 털 유무를 판별하는 classifier 구현

병변 이미지에서 털을 제거하는 프로그램 연구 및 개발(GAN)



클락틱스(김이윤) / Software Engineer

2019.08 – 2020.02, 서울특별시

- '마피아MC' : NUGU 스피커를 활용한 마피아 사회자 어플

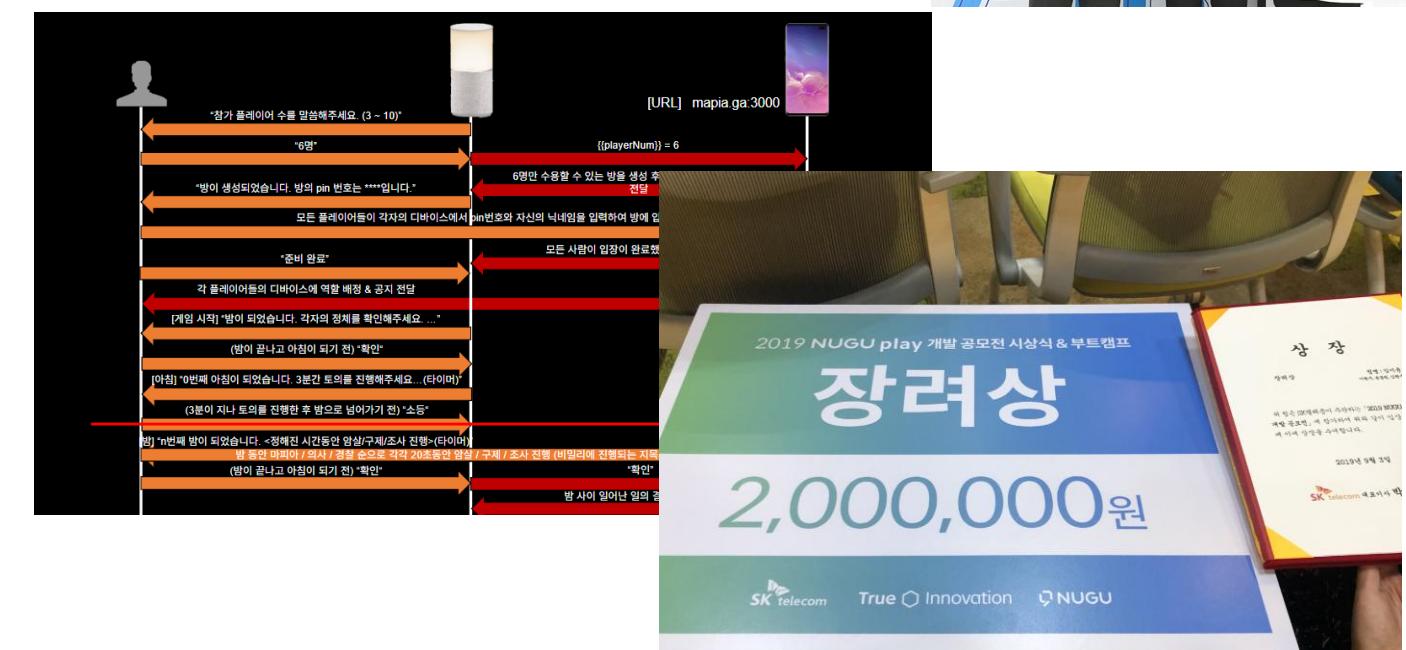
NUGU Play kit를 활용하여 어플 브랜치 생성

전반적인 게임 로직 작성 및 서비스 기획

- '숫자야구' : NUGU 스피커를 활용한 숫자야구 어플

NUGU Play kit를 활용하여 어플 브랜치 생성

전반적인 게임 로직 작성 및 서비스 기획



발표자료 : <https://bit.ly/3msfxgd>

02 Projects

티스푼 / Front-end engineer & Planner

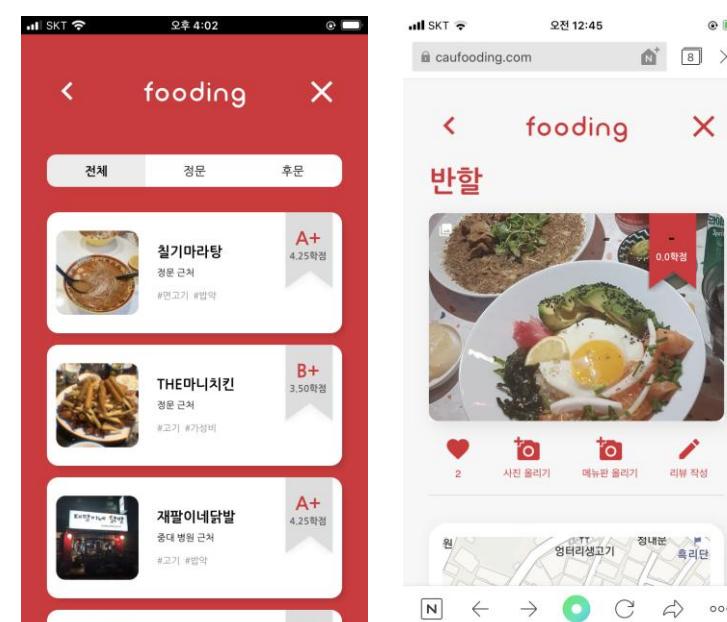
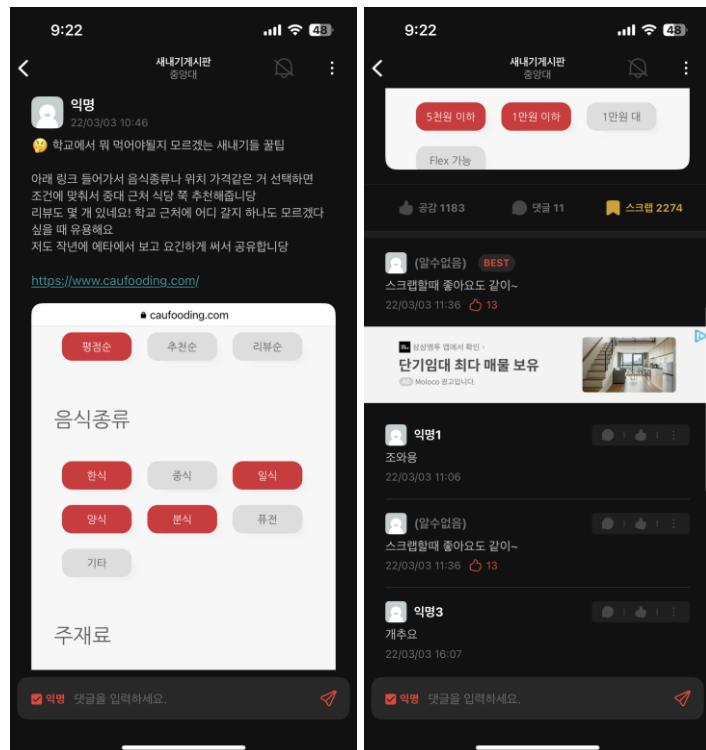
2020.07 – 2023.08

- 'Fooding' : 중앙대 재학생들을 위한 학교근처 식당 추천
플랫폼(React & Redux / Mobile Web)

필터링 검색, 문답 검색 UI 및 기능 구현

메인 홈페이지, 룰렛 서비스 UI 및 기능 구현

식당 데이터 분류 작업 & 서비스 기획



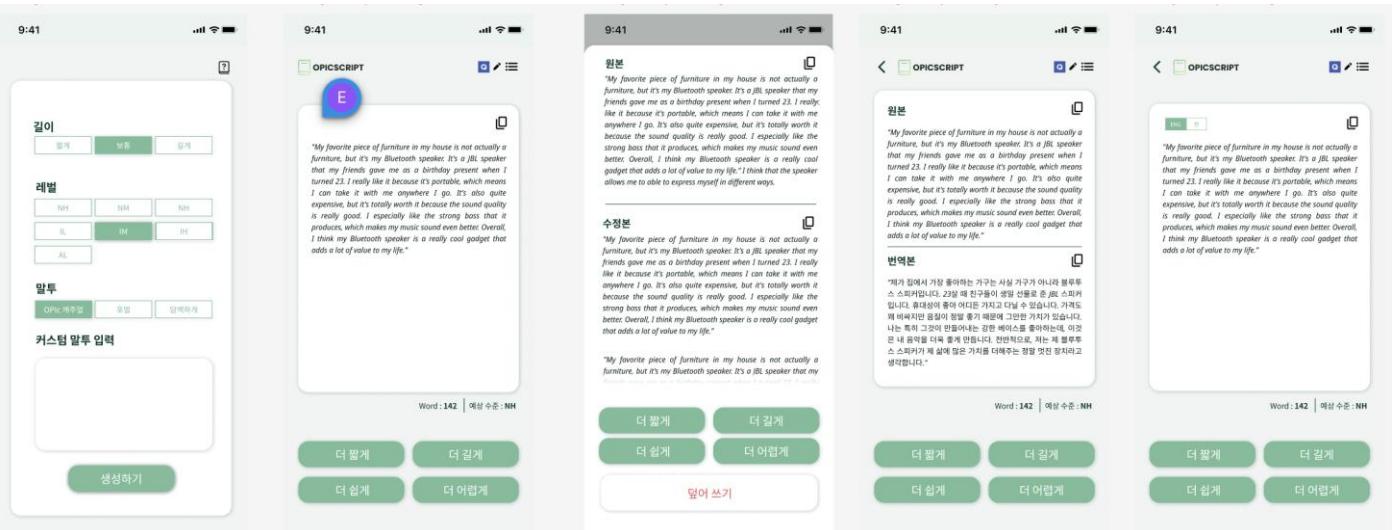
02 Projects

OPICSCRIPT / Front-end engineer & Tech Leader

2023.05~2025.05

- ‘Opicscript’ : 오픽 시험 준비생들을 위해 주제별 연습 스크립트를 사용자의 입력에 따라 난이도 별로 조정하여 자동 생성해주는 웹앱 서비스(React / Node.js / Mobile Web)

개발 총괄, 서비스 기획, front-end 개발



02 Projects

KIIYO / Full-stack engineer & Tech Leader

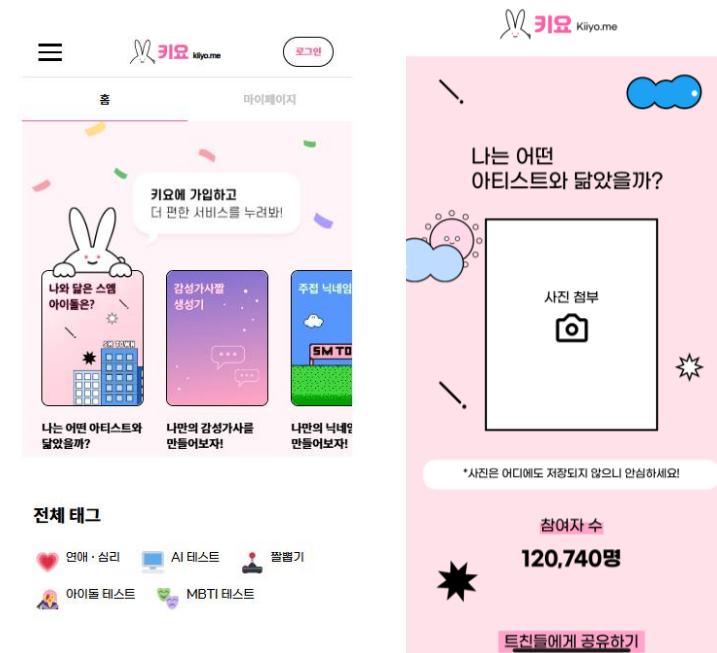
2021.07~2022.03

- 'KIIYO' : 비전 AI기술을 말랑말랑하게 적용하기 위한 목적으로 만들어진 웹앱 서비스 (React / Django / Mobile Web) & SM Ent.와 콜라보 진행

개발 총괄, front-end 개발, back-end 개발

<https://kiiyo.me/>

[https://namu.wiki/w/Kiiyo\(%ED%82%A4%EC%9A%94\)](https://namu.wiki/w/Kiiyo(%ED%82%A4%EC%9A%94))



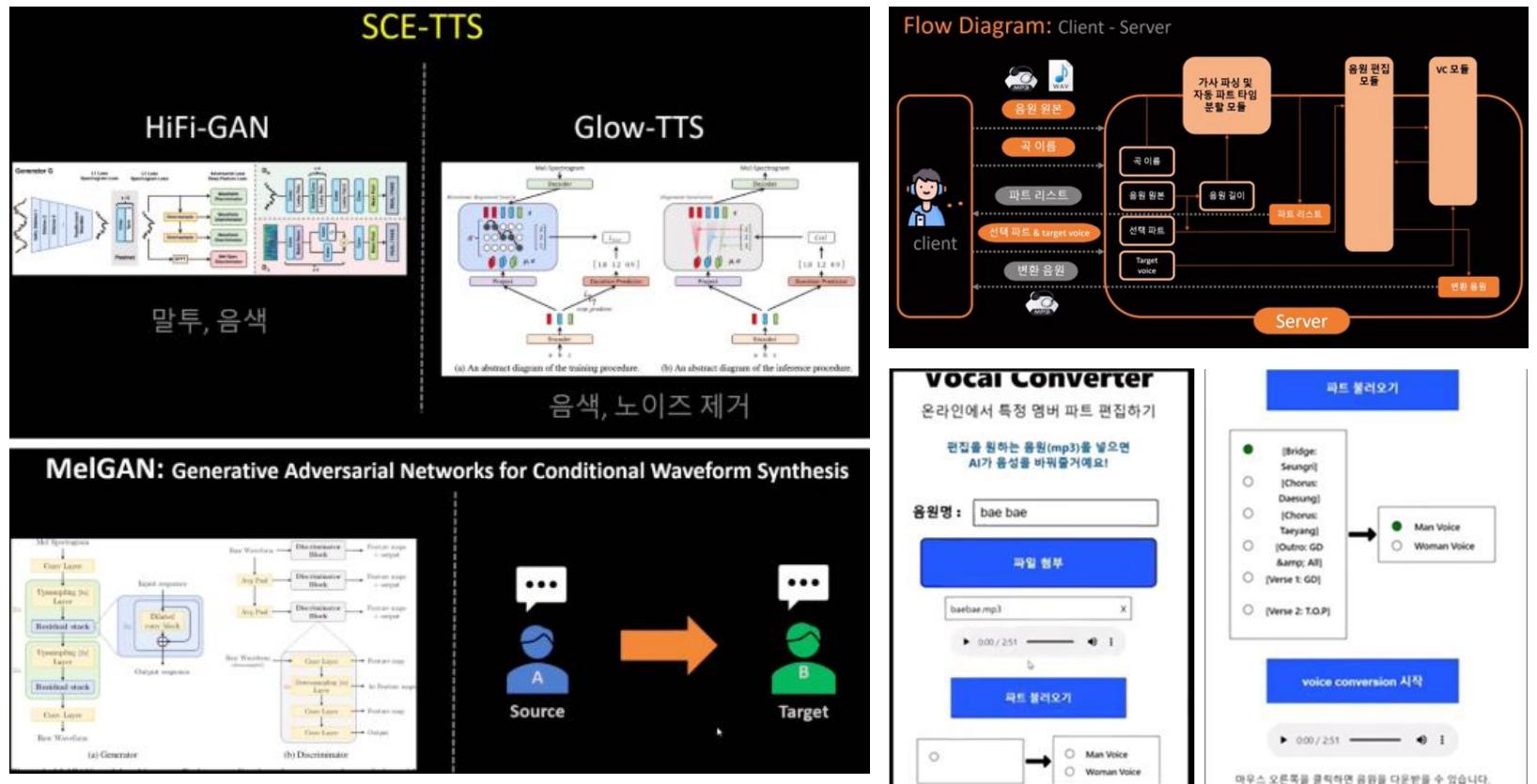
- **프로젝트 목표:**
사용자가 손쉽게 AI 기반 콘텐츠를 체험할 수 있는 웹 플랫폼 구축
- **핵심 아이디어:**
 - 예: "나는 어떤 SM 아티스트와 닮았을까?"
 - 소량의 이미지·라벨 데이터로 분류(Classification) 모델 학습
- **기술 구성:**
 - 데이터 일반화 향상: 이미지 증강(Image Augmentation) + 전이학습(Transfer Learning)
 - 모델 배포: Django 기반 API 서버
 - 프론트엔드: React로 구현 → 이미지 업로드 후 즉시 결과 시각화



02 Projects

보컬 변환 어쿠스틱 머신러닝 모듈 - 씨 없는 멜론

CAU Capstone Design, Sep 2021 - Dec 2021



프로젝트 목표:

음원 내 특정 멤버의 목소리를 다른 사람의 목소리로 변환하는 웹 애플리케이션 제작

핵심 기술:

- 음성 변환(Voice Conversion): **MelGAN vocoder** 활용
- 음성 합성(TTS): **SCE-TTS (HiFi-GAN + Glow-TTS) 기반 data augmentation**
- 음원 분리: **Spleeter**로 MR/보컬 분리
- 웹 배포: **React (Frontend) + Flask (Backend)**

데이터 처리:

동일 문장 발화 데이터 부족 문제를 TTS 기반 음성 합성으로 해결

시스템 구조:

사용자가 음원 업로드 → 변환 모델 수행 → 변환 결과 다운로드

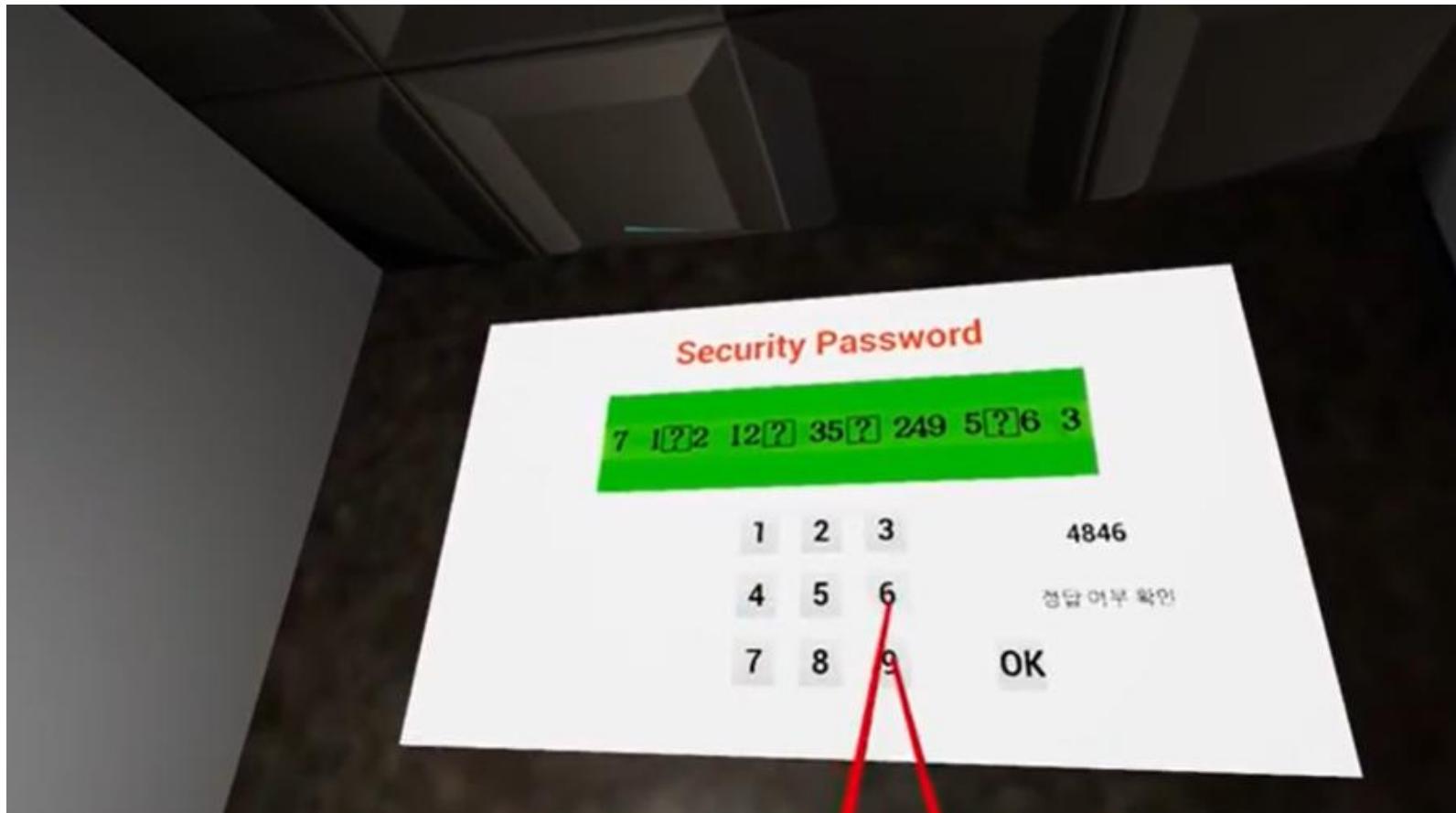
프로젝트 발표영상 : https://www.youtube.com/watch?v=E_UenhcZ1x4

02 Projects

VR Projects

Grad Escape (Unreal Engine VR 공포 방탈출 게임)

CAU 가상 증강 혼합현실, Nov 2021 - Dec 2021

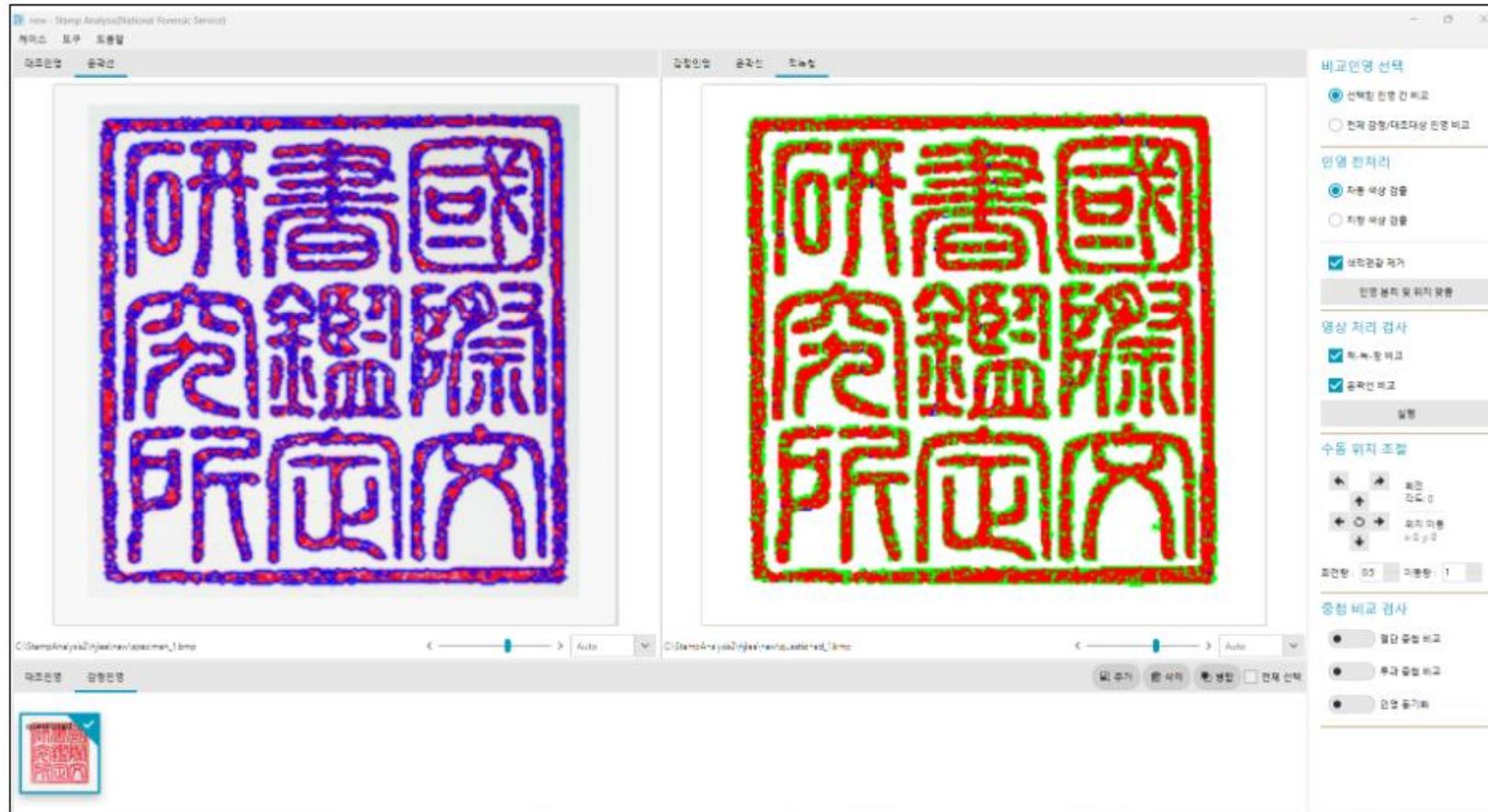


프로젝트 데모영상 : <https://www.youtube.com/watch?v=Qx5GMoysw9E>

02 Projects

NFS 인영 감정 프로그램

국립과학수사연구원, Feb 2023 - Jul 2023



프로그램 메뉴얼 : https://dlguswn3659.github.io/assets/files/README_mosaic.pdf

Article

- **July 2022:** Multi Wallet Dashboard, "MEPE"로 HackATOM 2022에서 HackATOM Seoul Critics' Awards - Interchain Award를 수상하고 블록체인 컨퍼런스 Buidl Asia 2022에서 발표
- **September 2021:** 한국프롭테크포럼X서울대학교 환경대학원에서 주최한 프롭테크 인 캠퍼스 SNU 2021에서 상업용 부동산 챗봇 및 전대차 플랫폼 스페이스톡으로 3위 수상
- **September 2020:** 중앙대학교 다빈치 SW/AI TECH FAIR 2020 PE 연구활동 발표대회에서 병변 이미지 처리 학습을 주제로 최우수상(1위) 수상
- **November 2019:** 2019 서울장학재단 공익인재장학 대회에서 고독사 프로젝트로 우수상(함께공감상) 수상
- **September 2019:** 2019 SKT NUGU Play 공모전에서 음성 마피아 게임 사회자 서비스, 마피아 MC로 장려상 수상

8 Honors and Awards

Evmos-Covalent DeFi (2nd Place) Dec 2022

HackATOM Seoul 2022 Cosmos SDK (1st Place) Jul 2022

HackATOM Seoul 2022 Critics' Awards - Interchain Award Jul 2022

Academic Merit Scholarship, ChungAng University 2021-2

프롬테크 인 캠퍼스 SNU 2021 (3rd Place) Sep 2021

중앙대학교 다빈치 SW/AI TECH FAIR 2020 PE 연구활동 발표대회 최우수상(1st Place) Sep 2020

중앙대학교X로봇협회 창의융합설계대회 로봇협회장상(1st Place) Dec 2019

중앙대학교 다빈치sw교육원 2019 Best Software Awards 우수상(2nd Place) Nov 2019

2019 서울장학재단 공익인재장학 우수상(함께공감상)(2nd Place) Nov 2019

2019 SKT NUGU Play 공모전 장려상(3rd Place) Sep 2019

Always-on 프로젝트 경진대회 우수상(2nd Place) Nov 2018

05 Leadership & Volunteer

Leadership

소프트웨어대학 홍보대사 1기 미디어부장

Software Collage, ChungAng University, 2019

CLUG 안드로이드 셀 부셀장

Software Engineering Academic Club, ChungAng University, 2019

전교부회장

명지고등학교, Jul 2016 - Jul 2017

Volunteer

2018 평창동계올림픽 SID Passion Crew

강릉아이스하키센터, GHC Sports Information Desk, Feb 2018

연구소개서

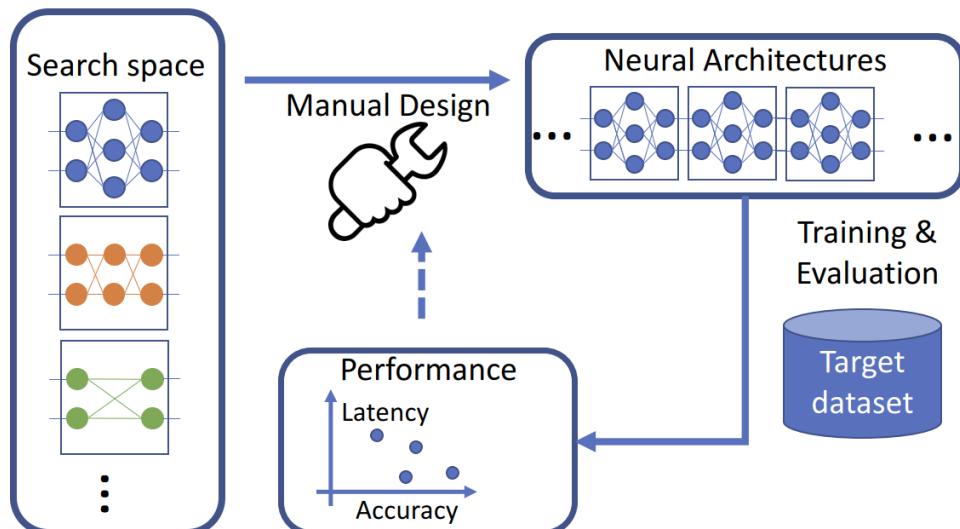
Hyunju Lee

School of Electrical and Electronic Engineering

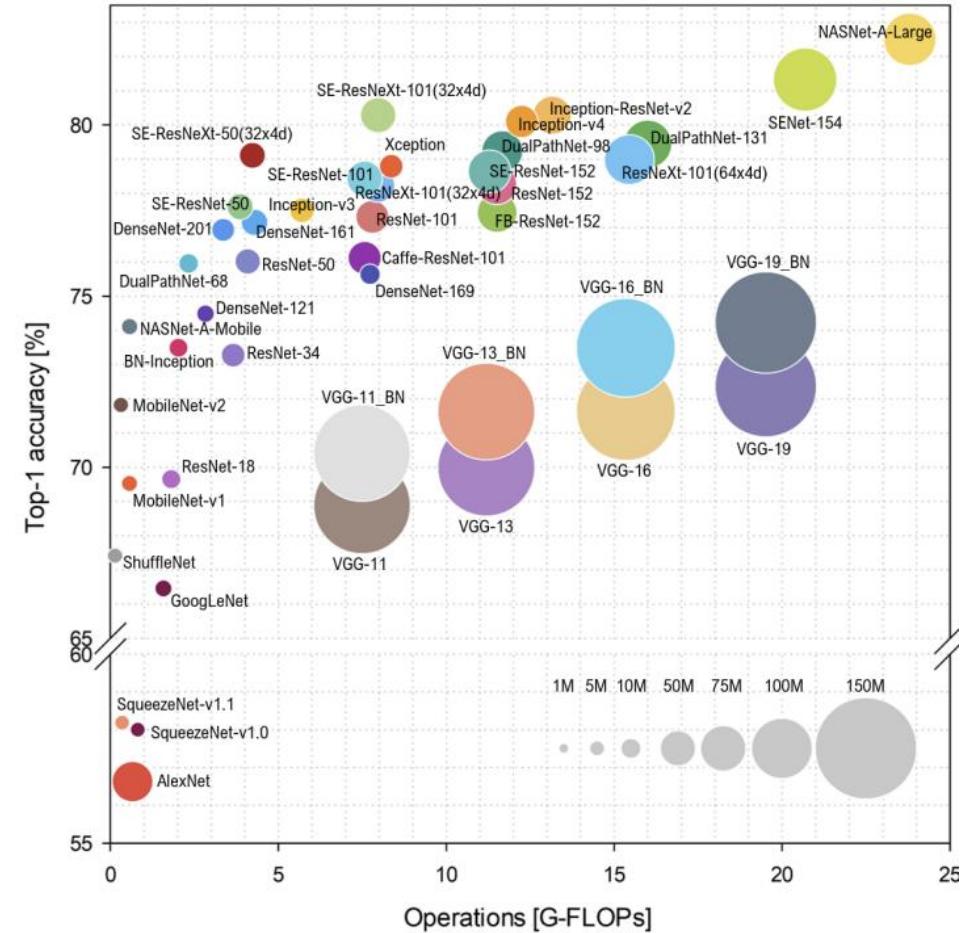


딥러닝 모델 네트워크 구조 설계

- 수동적인 네트워크 아키텍처 설계의 한계
 - 전문성 및 경험 요구: 머신러닝 전문가의 지식과 경험이 필수적입니다.
 - 반복적인 수작업: 많은 시간과 시행착오를 거쳐야 합니다.
 - 높은 훈련 비용: 막대한 시간, GPU, 전력 비용이 소모됩니다.



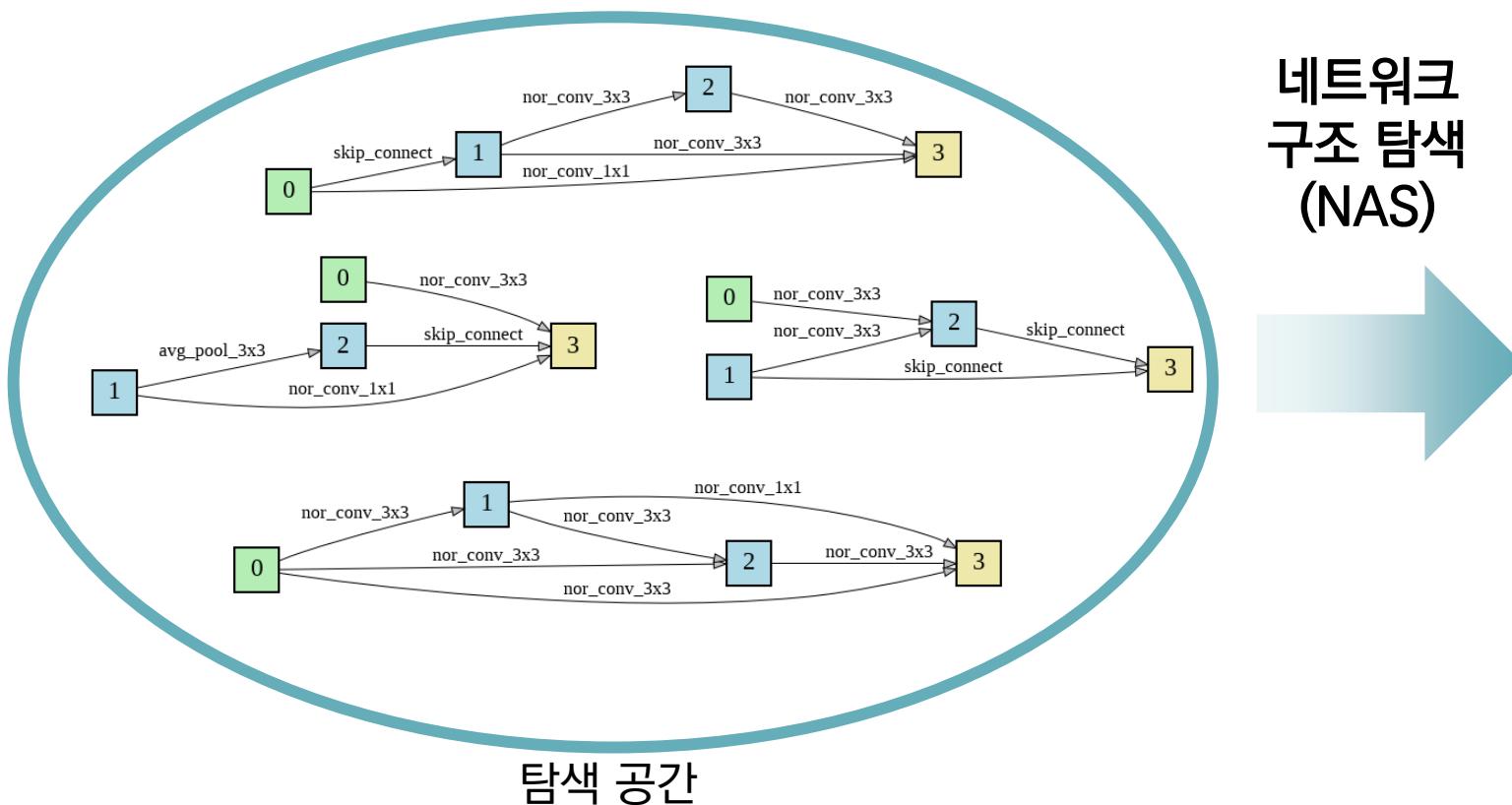
[Source] Wu et al., CVPR, 2019.



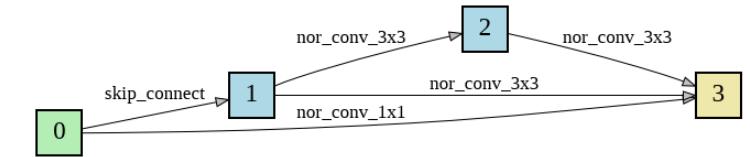
[Source] Simone et al., IEEE Access, 2018.

네트워크 구조 탐색 (NAS)

- 커널 수, 레이어 개수, 연산 종류 등이 서로 다른 수많은 네트워크 구조 검색 공간 내에서 최고의 성능을 가지는 아키텍처를 자동으로 찾아내는 연구



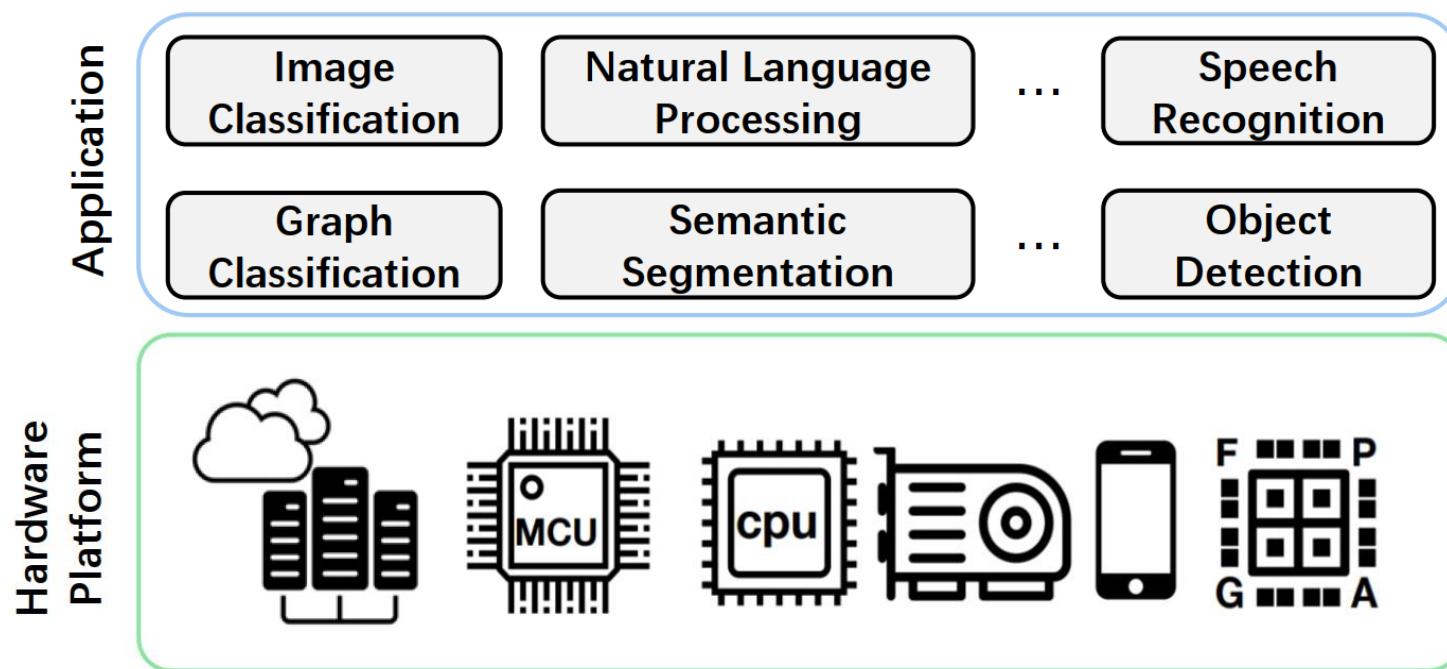
네트워크
구조 탐색
(NAS)



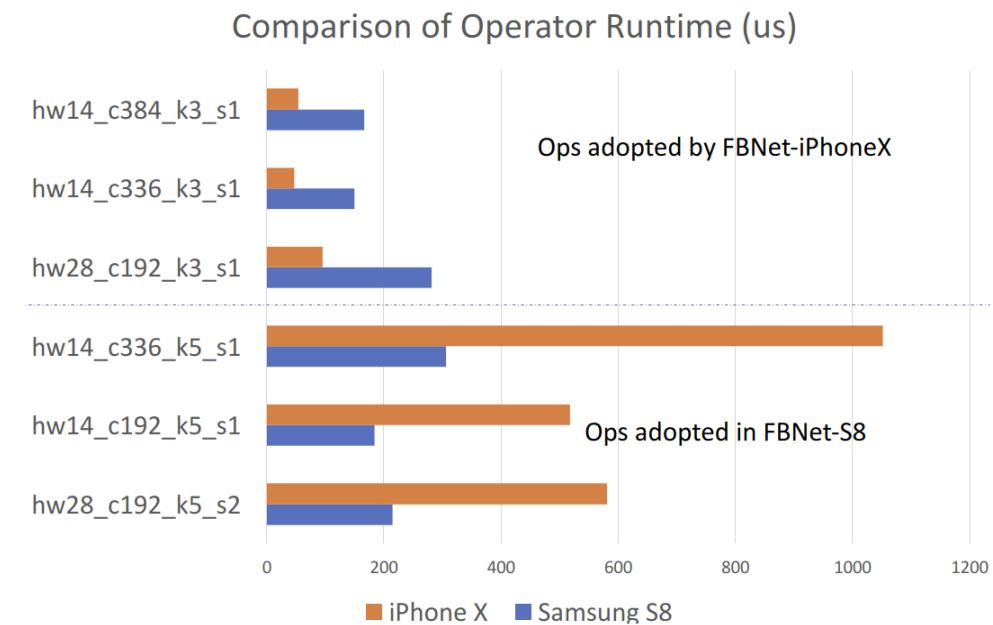
선택된 최적의 네트워크

왜 NAS가 중요한가?

- 제약 조건 내에서 새로운 아키텍처 설계: 특정 제약 조건에 맞는 새롭고 맞춤화된 네트워크 아키텍처를 설계합니다.
- 다양한 환경에 적응: 여러 애플리케이션이나 하드웨어 리소스(예: 매개변수, FLOPS, 지연 시간, 전력 등)에 맞춰 효율적으로 대처할 수 있습니다.



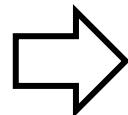
[Source] Guihong et al., IEEE TPAMI, 2024.



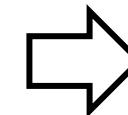
[Source] Wu et al., CVPR, 2019.

네트워크 구조 탐색(NAS) 과정

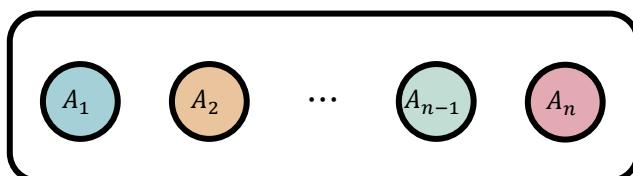
탐색 공간 정의



전략적으로 탐색
공간 내의 후보
구조들을 평가



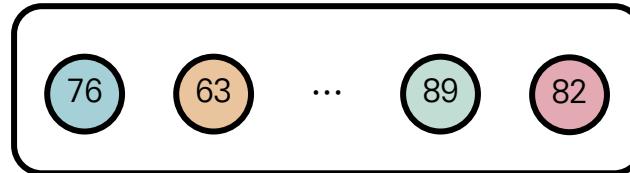
가장 최적의 후보 구조를 선택한
후 해당 네트워크만을 평가



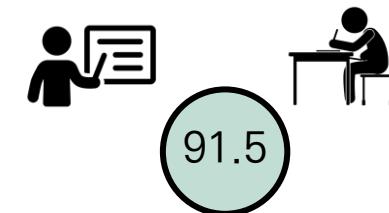
연산 후보
모델 깊이와 너비
커널 크기
채널 확장 비율
...



고성능 아키텍처를 출력하는
제어기를 훈련시킴



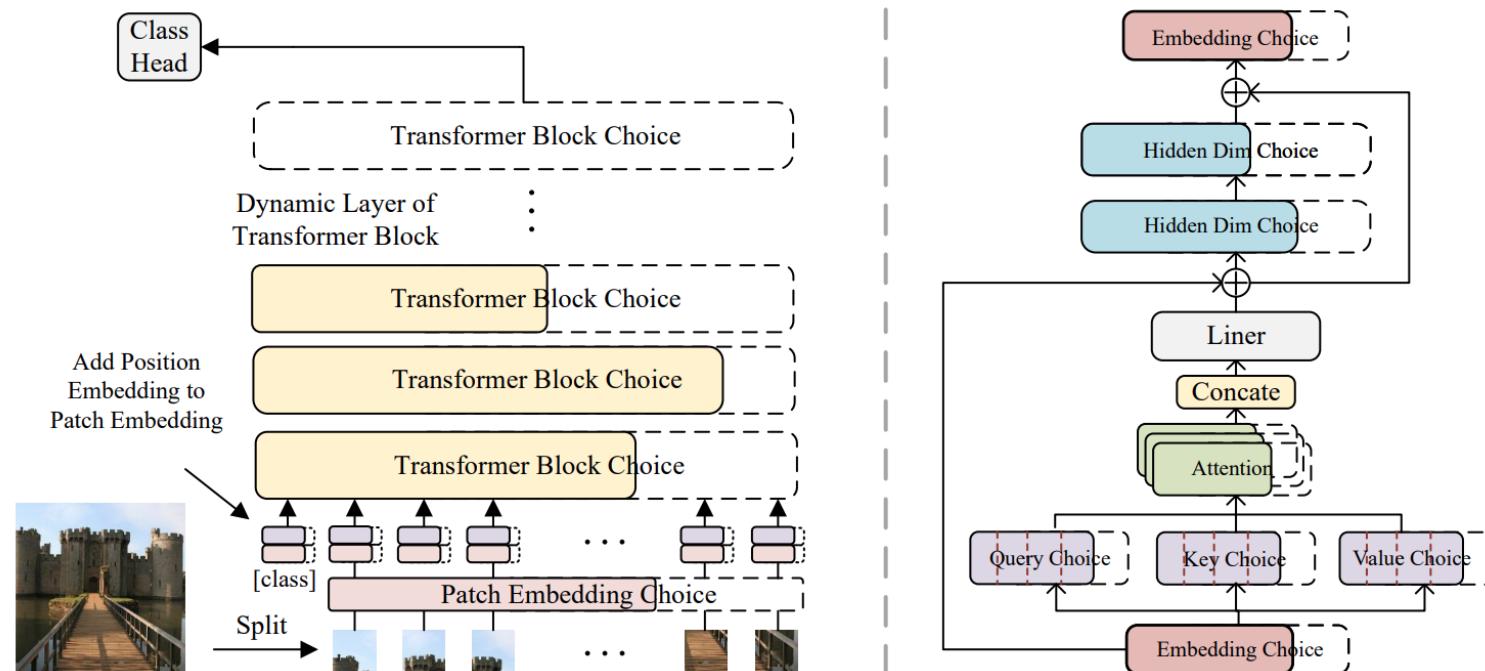
후보 아키텍처의 점수를 할당함 (실제/근사 성능,
대리 점수 등)



최종적으로 선택된 네트워크를
목표 데이터셋으로 훈련하고
테스트함

대표적인 ViT 모델 탐색 공간 : AutoFormer

- 기존에 CNN에만 적용되던 NAS를 비전 모델이 발전함에 따라서 ViT 모델에도 적용하고자 정의된 대표적인 ViT 모델 탐색 공간
 - 모든 블록이 공유하는 모델 설계 인자: 레이어 개수(Depth), 임베딩 차원수(embedding dimension)
 - 블록 별로 독립적으로 가지는 모델 설계 인자: 멀티 레이어 퍼셉트론 레이어의 채널 확장 비율(Expanding ratios in MLPs), 어텐션 헤드의 개수(#attention heads)

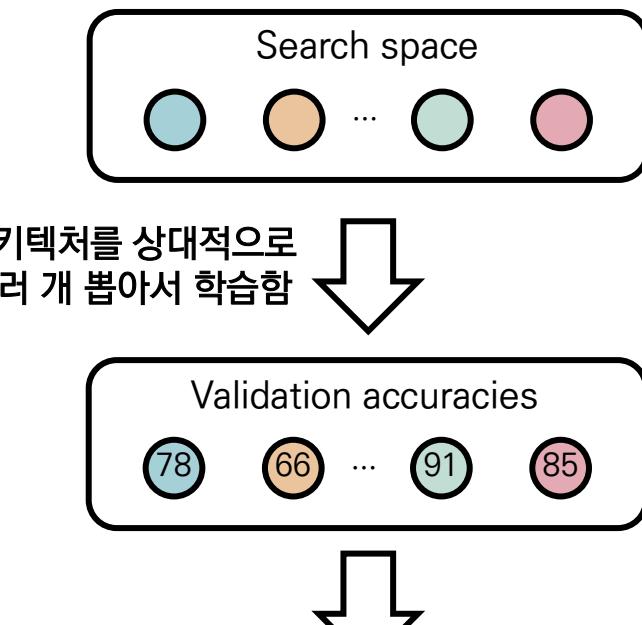


[Source] Chen et al., ICCV, 2021.

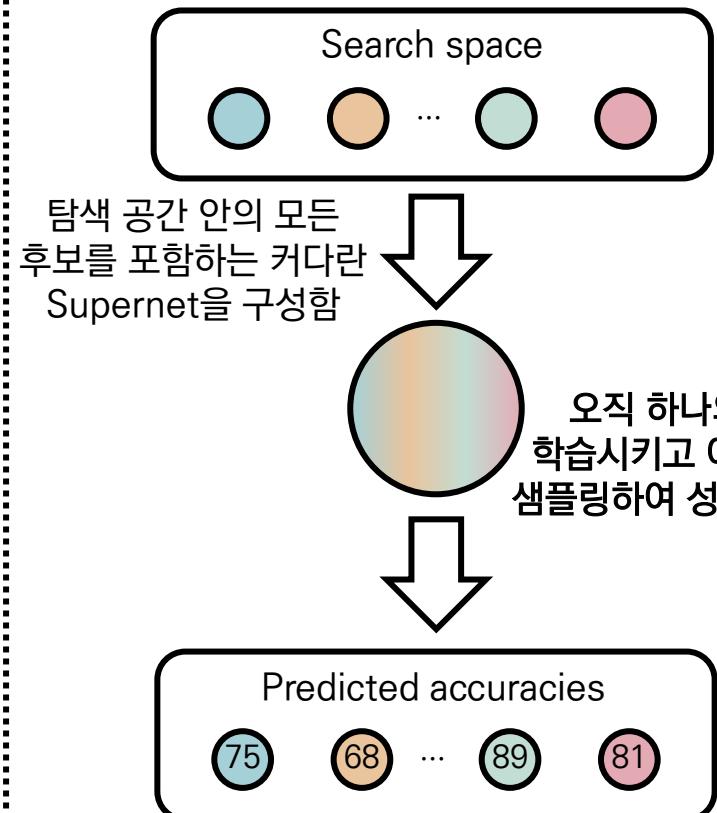
탐색 전략 종류: Multi-/One-(Few-)/Zero-shot

- 탐색 공간 내에서 가장 좋은 성능을 가지는 네트워크 구조를 찾아냄

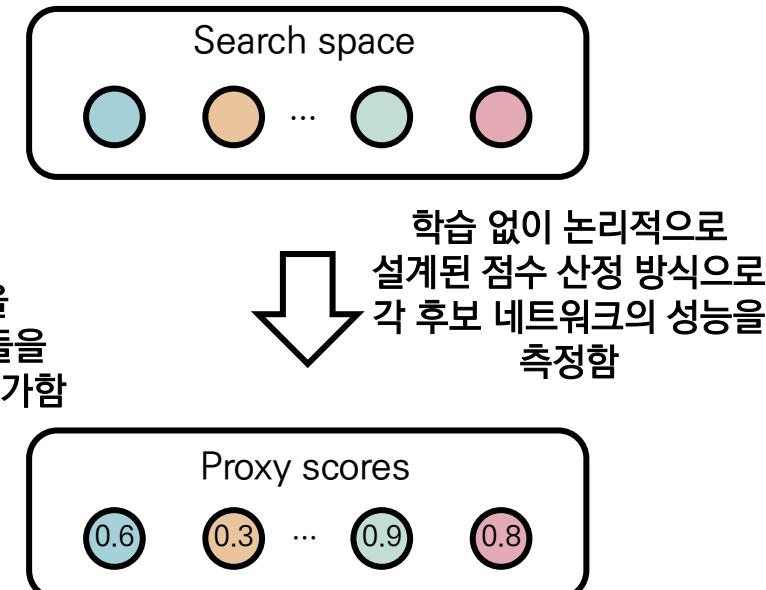
Multi-shot



One(few)-shot

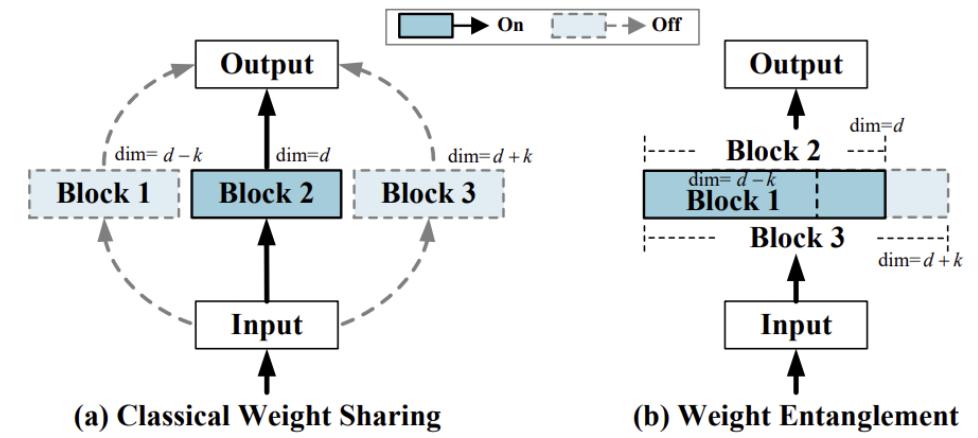
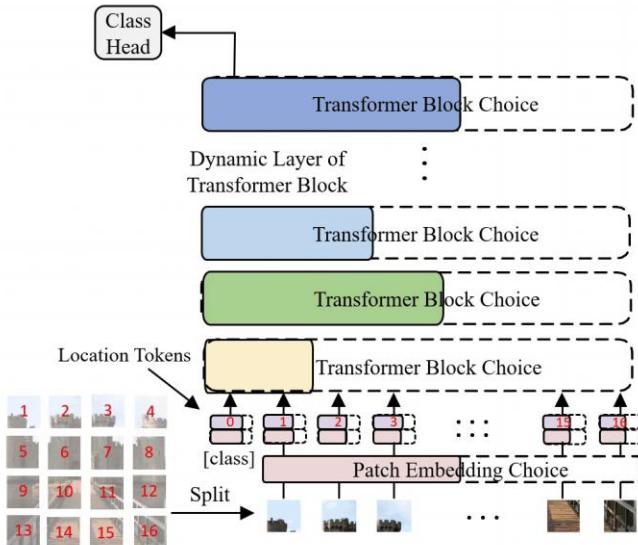


Zero-shot



One-shot: AutoFormer

- AutoFormer는 One-shot 전략에 적용 가능한 search space
- 모든 후보 아키텍처를 포함하는 SuperNet이 서로의 weigh를 공유하며 동시에 학습됨
 - 따라서 모든 포함된 모든 후보 아키텍처(subnets)들은 각각 독립적으로 학습이 될 필요가 없음(그러나 전문적인 훈련 기법이나 이론적인 보장은 없음)



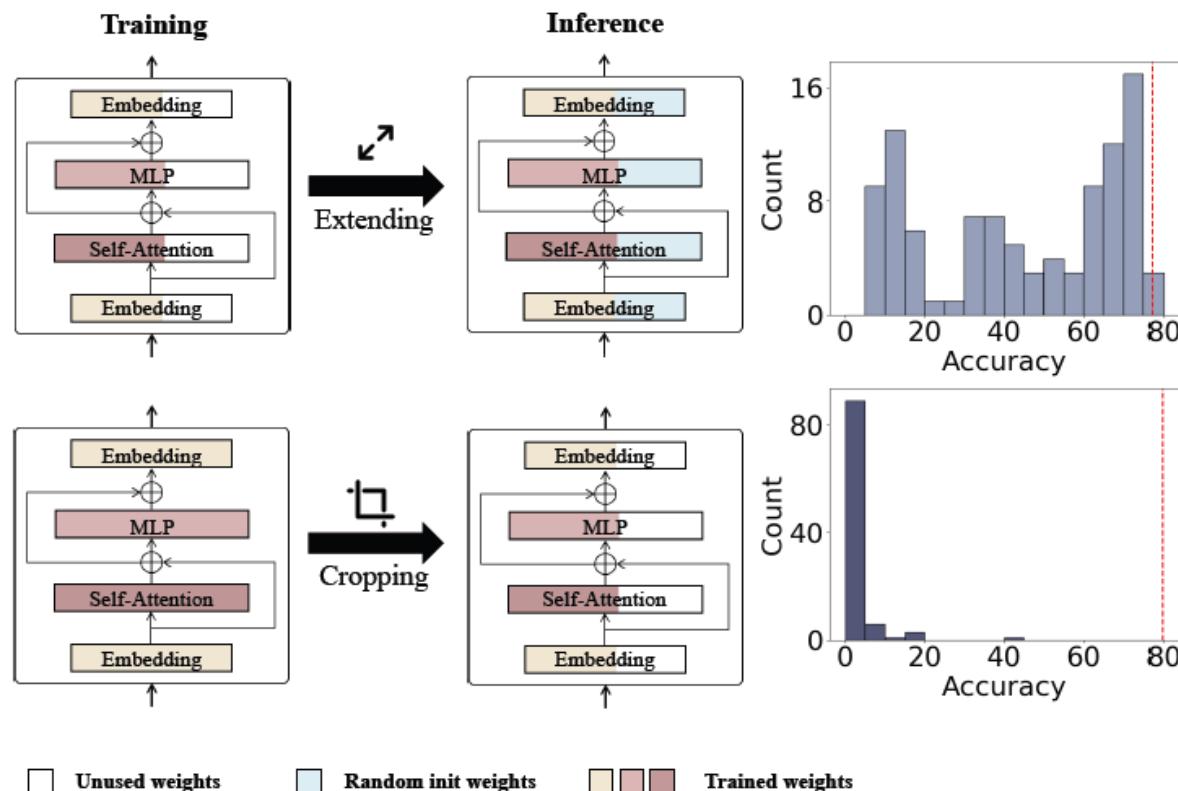
[Source] Chen et al., ICCV, 2021.

GrowTAS (Gradual Optimization for Weight-Sharing TAS)

WACV 2026 Accepted

- 기존 One-shot TAS(Transformer Architecture Search) 프레임워크들은 subnet 사이즈에 따라 supernet 학습에서 받는 영향도가 다름을 고려하지 않기 때문에 이를 고려한 새로운 One-shot TAS 프레임워크인 **GrowTAS**를 제안함. 이 프레임워크는 점진적인 방식으로 서브넷을 훈련시켜 검색 성능을 향상 시킴.
- 공유 가중치를 고정한 상태에서 큰 서브넷을 미세 조정하여 작은 서브넷의 성능 저하 없이 효과적인 전략을 제시함
- 표준 벤치마크에 대한 실험을 돌려본 결과 제안된 방법이 가장 좋은 성능을 보이며 해당 방법의 효율성을 입증함

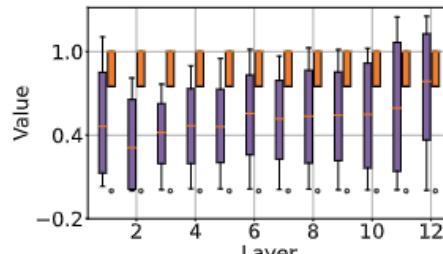
Motivation : 분석 실험



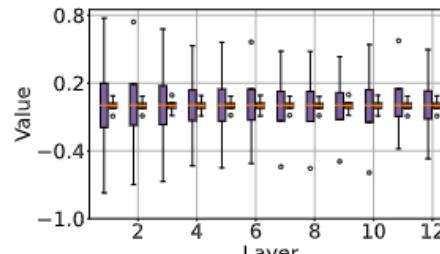
훈련 비대칭성

- 슈퍼넷을 가장 작은 서브넷으로만 훈련: 무작위로 샘플링된 큰 서브넷들도 여전히 좋은 성능을 보입니다.
- 슈퍼넷을 가장 큰 서브넷으로만 훈련: 무작위로 샘플링된 작은 서브넷들은 성능이 급격히 저하됩니다.
- 훈련 비대칭성: '작은 net → 큰 net'으로 훈련하는 것이 '큰 net → 작은 net'보다 더 효과적입니다.
- 결론: 작은 서브넷부터 훈련을 시작할 때 더 나은 전이성(transferability)을 보입니다.

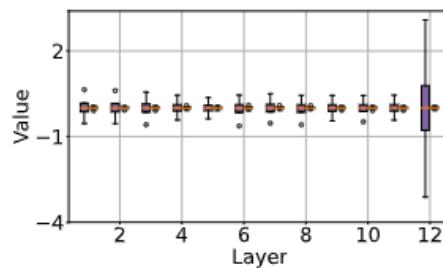
Motivation : 분석 실험



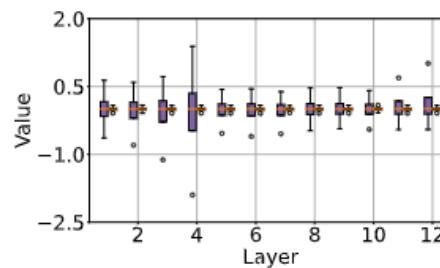
(a) LayerNorm1



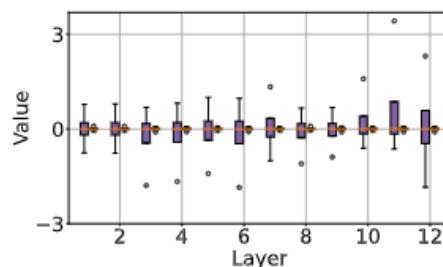
(b) QKV



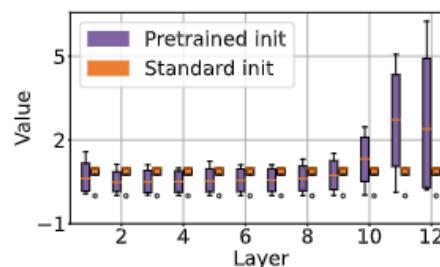
(c) Linear1



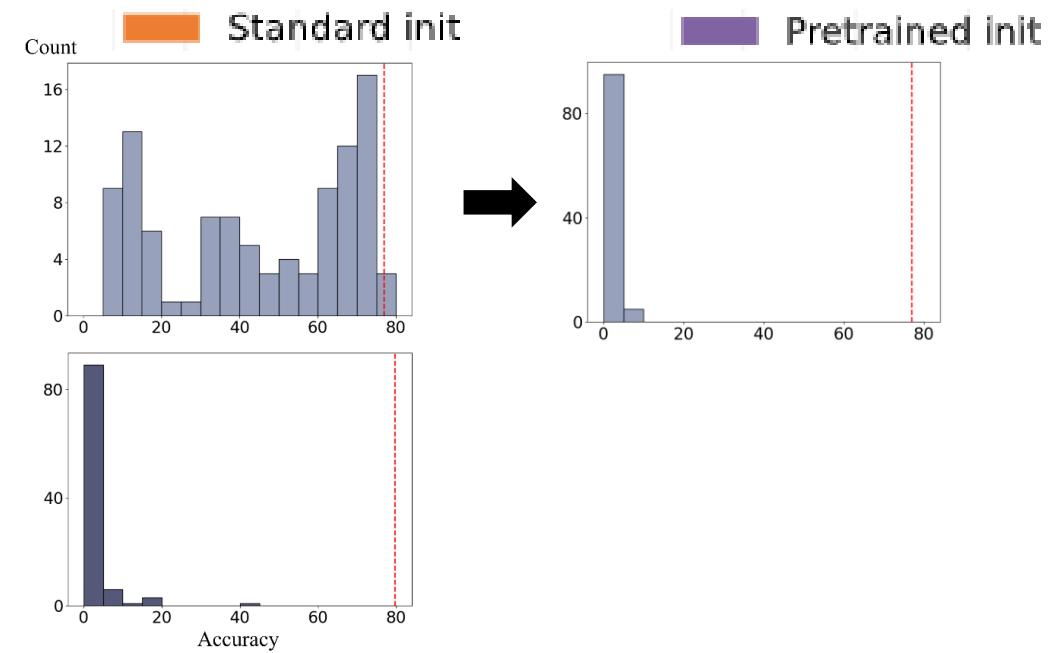
(d) Linear2



(e) Linear3



(f) LayerNorm2

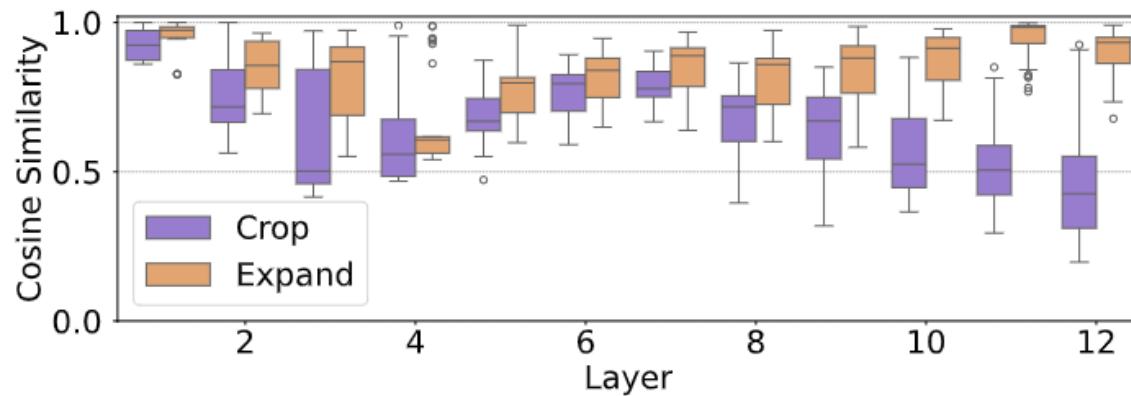


레이어 단위로 더욱 심층적으로 분석해 본 결과,

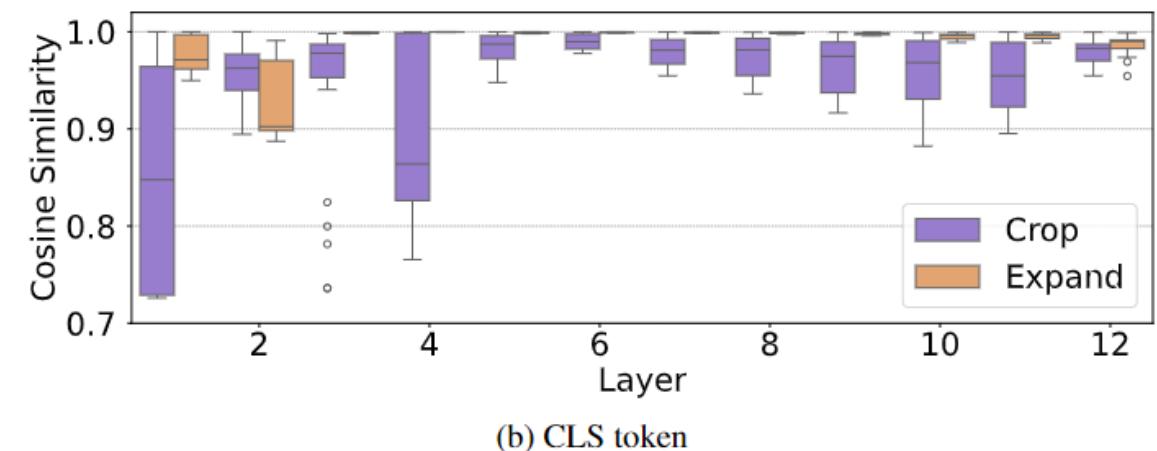
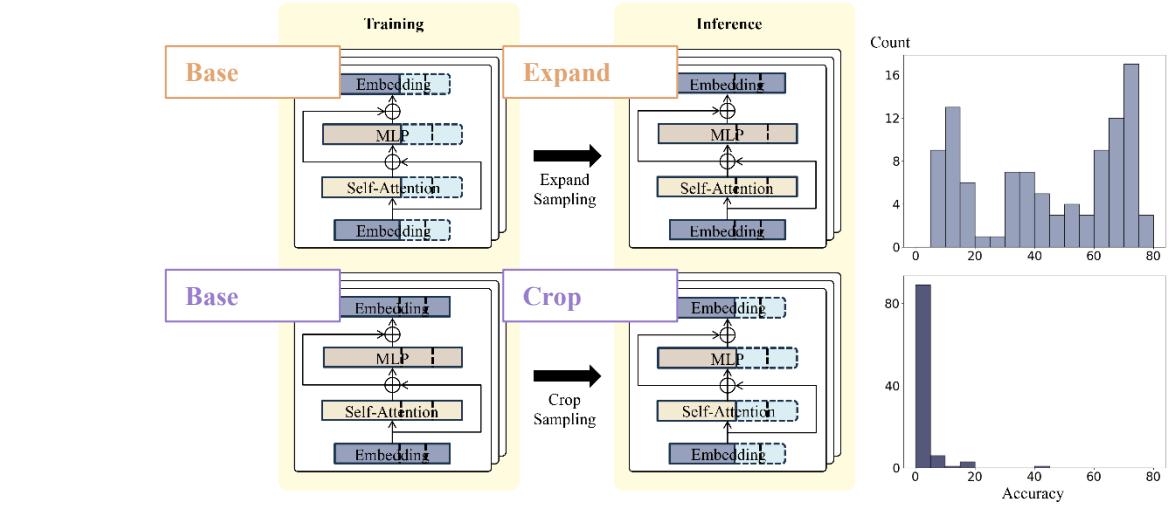
큰 서브넷으로만 슈퍼넷을 학습시킨 후 작은 서브넷들을 샘플링하여
레이어별 output을 측정해보면 학습이 제대로 된 서브넷의 output과
더 큰 오차를 발생시키는 것을 확인할 수 있었다.

Motivation : 분석 실험

레이어 단위로 더욱 심층적으로 분석해 본 결과,
큰 서브넷으로만 슈퍼넷을 학습시킨 후 작은 서브넷들을 샘플링하여
레이어별 output을 측정해보면 학습이 제대로 된 서브넷의 output과
더 큰 오차를 발생시키는 것을 확인할 수 있었다.

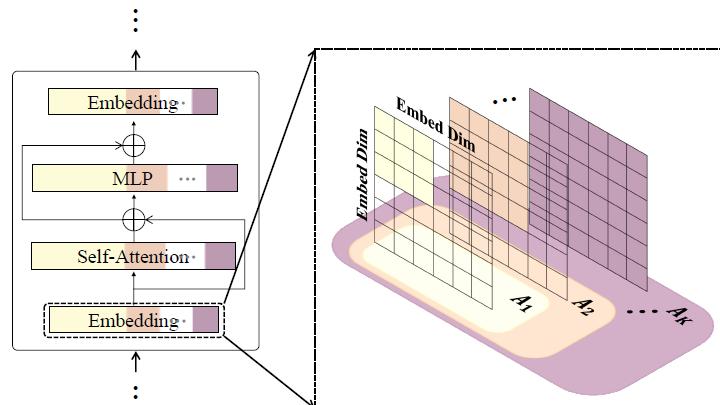
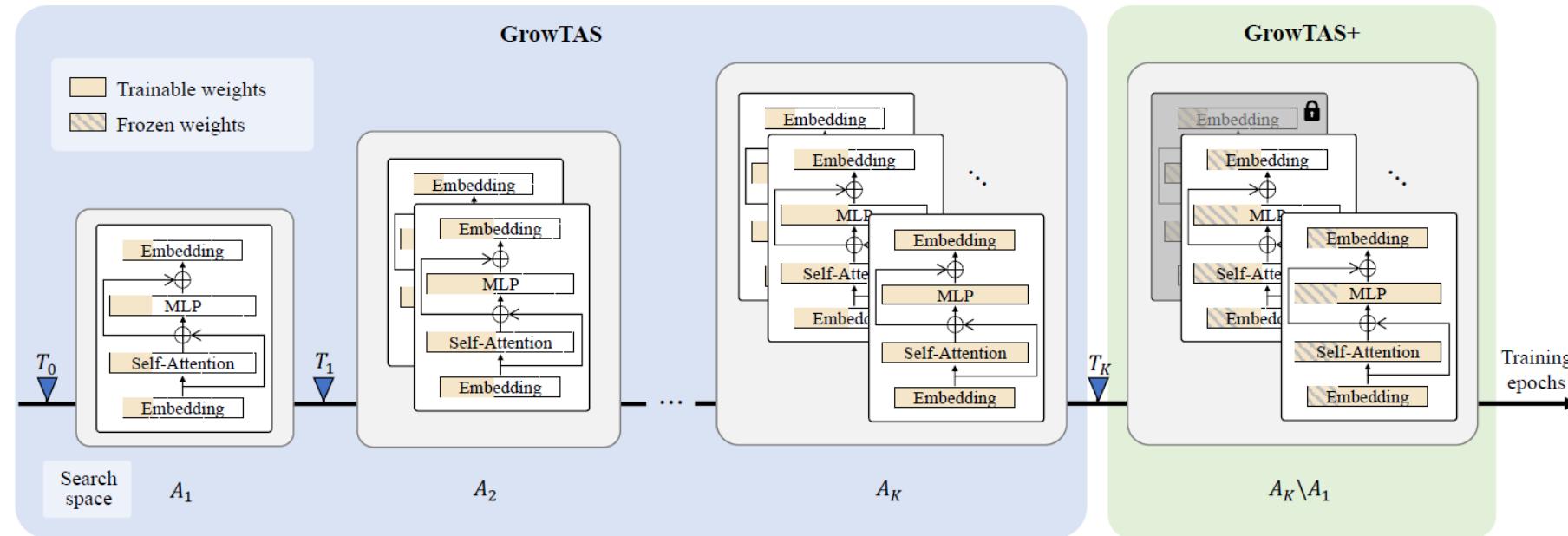


(a) ALL tokens



(b) CLS token

Method : GrowTAS



따라서 작은 서브넷이 큰 서브넷과의 학습으로 성능 저하가 발생할 위험이 크므로 작은 서브넷 \rightarrow 큰 서브넷으로 점진적으로 탐색 공간을 늘려나가면서 학습을 시키는 새로운 프레임워크인 GrowTAS를 제안함

왼쪽과 같이 사이즈를 기준으로 search space를 사전에 분류하여 슈퍼넷 학습 스케줄링에 사용함

Model	Top-1 (%)	#Params (M)	FLOPs (G)
ViT-T [9]	74.5	5.7	-
DeiT-T [32]	72.2	5.7	1.2
ConViT-T [10]	73.1	5.7	1.2
TNT-T [14]	73.9	6.1	1.4
ViTAS-C [29]	74.7	5.8	1.2
FocusFormer-T [22]	75.1	6.2	1.4
PreNAS-T† [35]	77.1	5.9	1.3
AutoFormer-T [5]	74.7	5.7	1.3
AutoFormer-T* [5]	74.9	5.9	1.3
GrowTAS-T	75.2	5.9	1.3
GrowTAS-T+	75.2	5.9	1.3
GrowTAS-T†	77.1	5.9	1.3
ViT-S [9]	78.8	22.1	5.1
DeiT-S [32]	79.8	22.1	4.7
ConViT-S [10]	81.3	27.0	5.4
TNT-S [14]	81.5	23.8	5.2
ViTAS-F [29]	80.5	27.6	5.5
FocusFormer-S [22]	81.6	23.7	5.1
PreNAS-S† [35]	81.8	22.9	5.1
AutoFormer-S [5]	81.7	22.9	5.1
AutoFormer-S* [5]	81.6	22.9	5.1
GrowTAS-S	81.6	22.8	5.1
GrowTAS-S+	81.8	22.4	5.1
PVT-Large [36]	81.7	61.0	9.8
DeiT-B [32]	81.8	86.0	18.0
ViT-B [9]	79.7	86.0	18.0
ConViT-B [10]	82.0	86.0	17.0
FocusFormer-B [22]	81.9	52.8	11.0
PreNAS-B† [35]	82.6	54.0	11.0
AutoFormer-B [5]	82.4	54.0	11.0
AutoFormer-B* [5]	82.4	54.0	11.0
GrowTAS-B	82.6	53.2	11.0
GrowTAS-B+	82.7	53.3	11.0

Experiments

Table 2. Transfer learning performance comparison across multiple datasets: CIFAR-10/100, Oxford Flowers, Stanford Cars, and iNaturalist-19. We report top-1 accuracy (%) and the number of model parameters (in millions).

Model	#Params (M)	CIFAR-10	CIFAR-100	Flowers	Cars	INAT-19
ViT-B/16 [?]	86M	98.1	87.1	89.5	-	-
ViT-L/16 [?]	307M	97.9	86.4	89.7	-	-
DeiT-B [?]	86M	99.1	90.8	98.4	92.1	77.7
ViTAE-S [?]	24M	98.8	90.8	97.8	91.4	76.0
DearKD-S [?]	22M	98.4	89.3	97.4	91.3	-
PreNAS-S [22]	23M	99.1	91.2	97.6	92.2	76.4
AutoFormer-S [3]	22.9M	98.9	89.6	97.9	92.1	77.4
ProgTAS-S (Ours)	22.4M	99.0	90.9	98.2	-	77.9

- 기존 베이스라인들과 비교했을 때 가장 좋은 성능을 가지는 것을 확인할 수 있다.