How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers

2022.01.21

Data Mining and Quality Analytics Lab

박진혁

Data Mining
Quality Analytics

목차

1. Introduction

❖ 발표자 소개

- 박진혁
- Data Mining & Quality Analytics Lab(김성범 교수님)
- 석·박사 통합과정 5학기 재학 중(2019.8 ~)

✓ 관심 분야

- Computer Vison
- Image Captioning
- Multimodal Learning

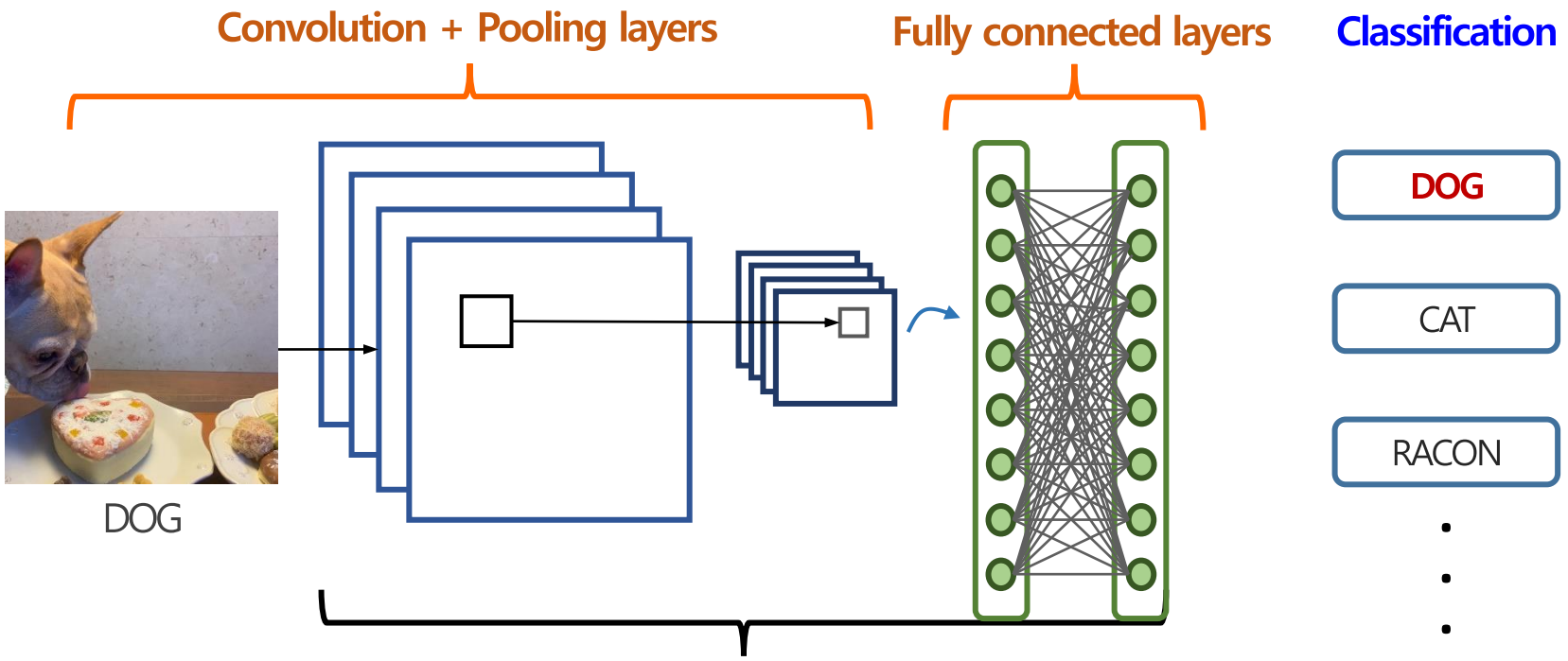
1. Introduction

❖ From CNN to ViT

➢ Convolutional Neural Network

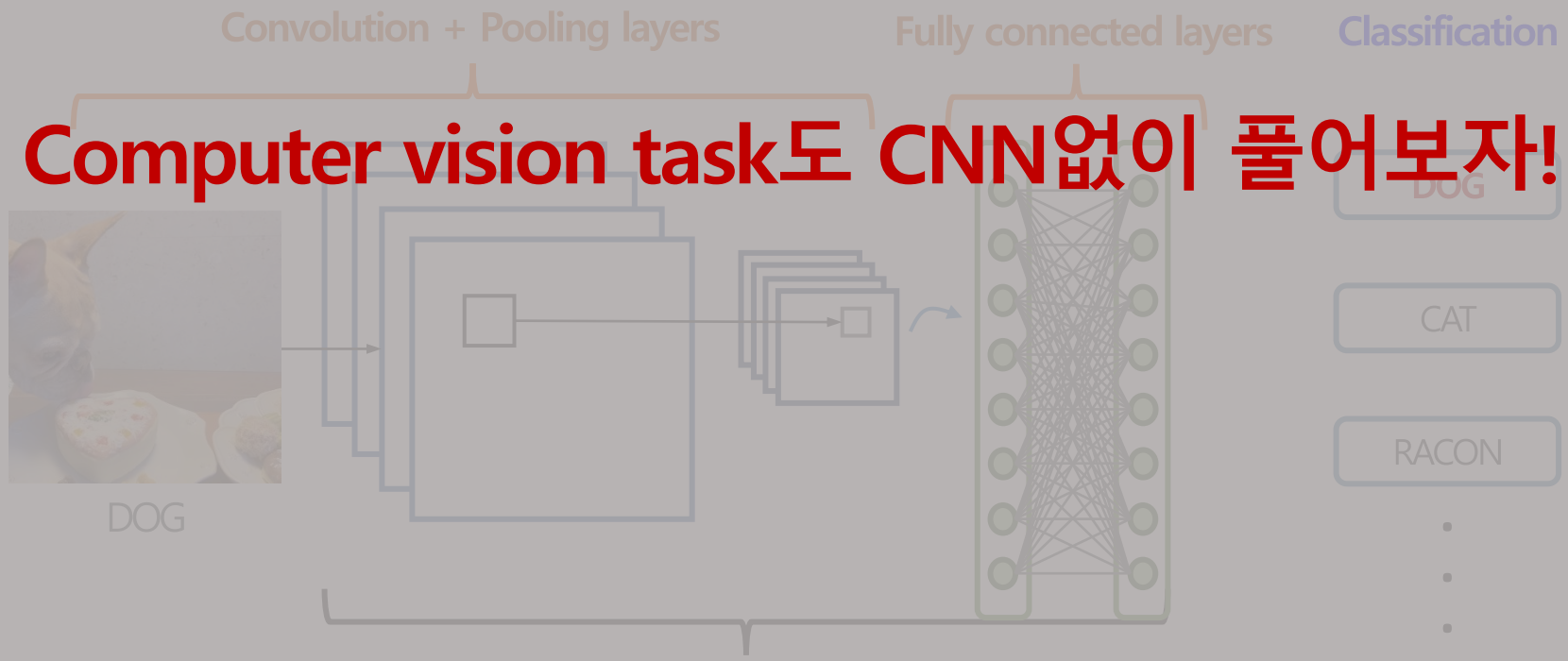
➢ Computer vision분야에서 가장 많이 사용되는 architecture

➢ 이미지를 입력 받아 이미지의 공간정보를 유지한 채 학습함



1. Introduction

- From CNN to ViT
  - Convolutional Neural Network
    - Computer vision분야에서 가장 많이 사용되는 architecture
    - 이미지를 입력 받아 이미지의 공간정보를 유지한 채 학습함

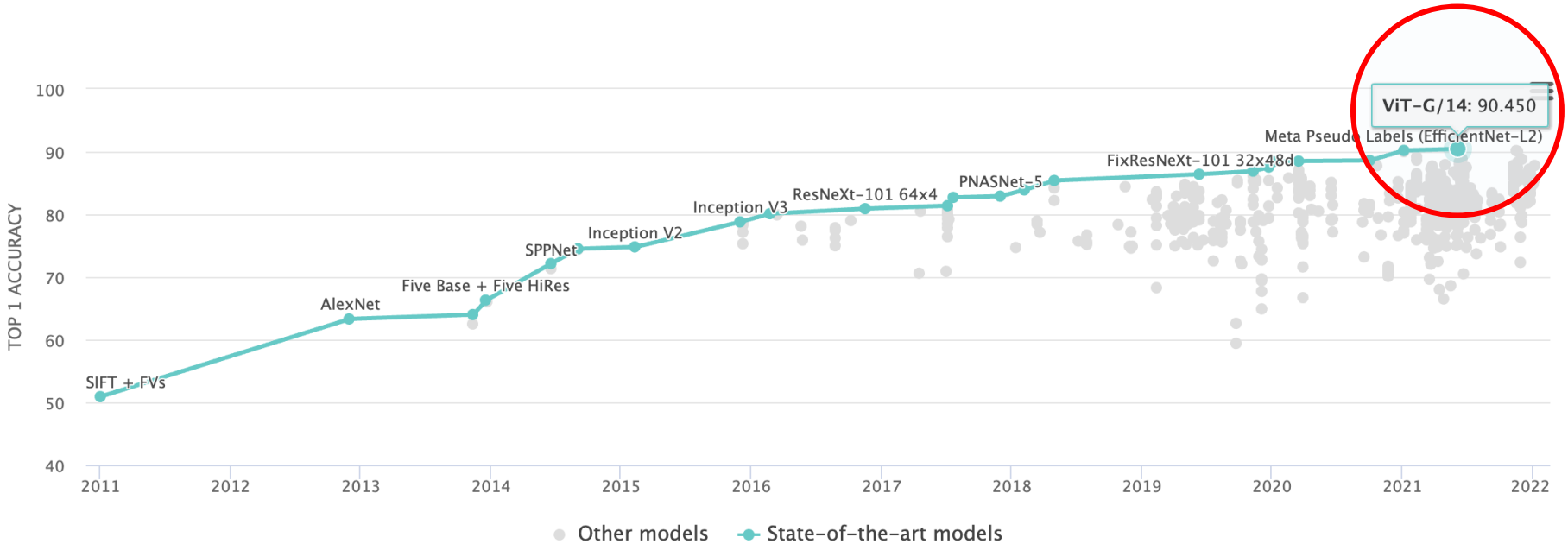


Computer vision task도 CNN없이 풀어보자!

1. Introduction

❖ From CNN to ViT

- ➢ Computer vision분야에서 합성곱 신경망이 아닌 Transformer를 적용
- ➢ Transformer architecture가 computer vision분야에서 수 많은 SOTA를 달성
- ➢ Computer vision분야에 NLP architecture의 적용 가능성을 보여준 논문



2. What is ViT?

❖ Vision Transformer(ViT)

- ➢ 2022년 1월 17일 기준 2196회 인용
- ➢ Google Research에서 발표
- ➢ Transformer architecture를 활용하여 image classification을 수행

AN IMAGE IS WORTH 16X16 WORDS: TRANSFORMERS FOR IMAGE RECOGNITION AT SCALE

Alexey Dosovitskiy*,†, Lucas Beyer*, Alexander Kolesnikov*, Dirk Weissenborn*, Xiaohua Zhai*, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, Neil Houlsby*,†

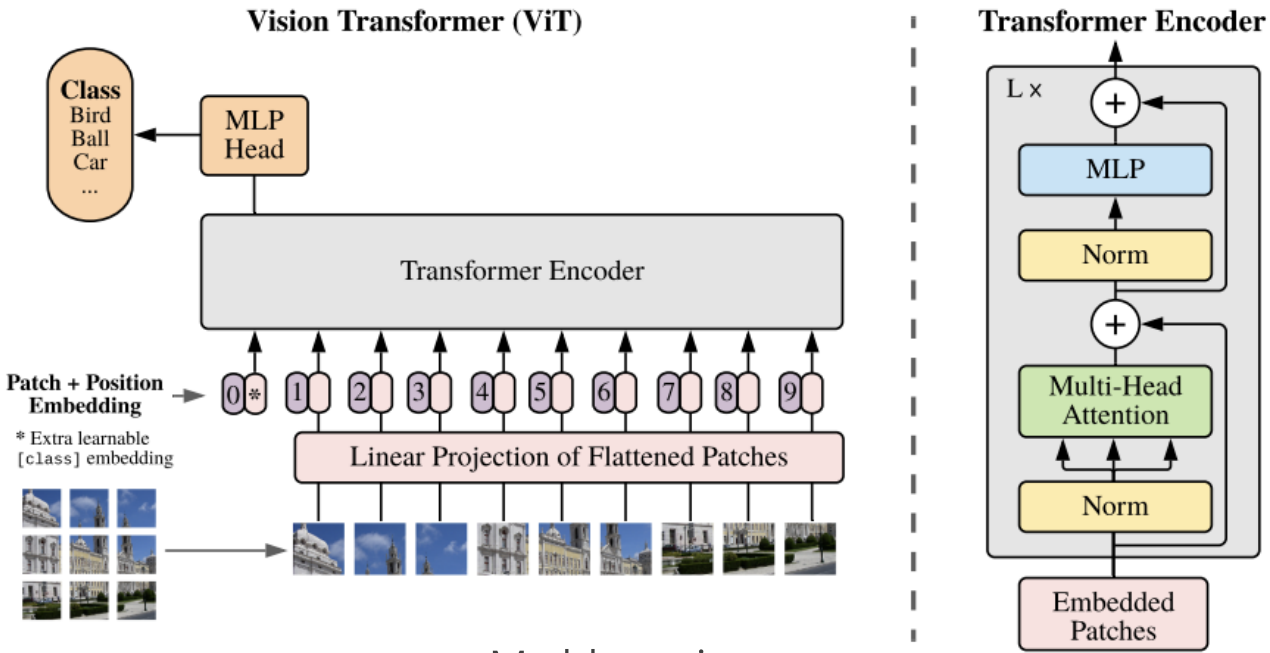
*equal technical contribution, †equal advising

Google Research, Brain Team

2. What is ViT?

❖ Vision Transformer(ViT)

- ➢ 입력 이미지
- ➢ 모델 아키텍처
- ➢ 대용량 데이터의 사전학습

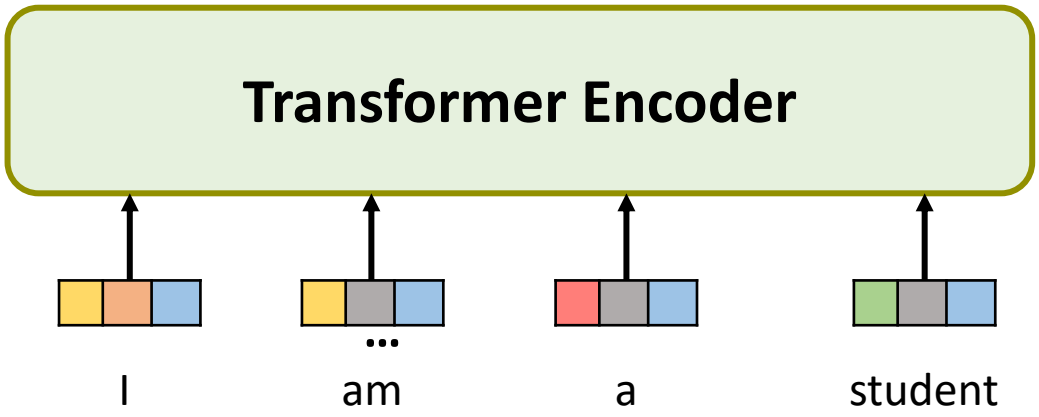


<Model overview>

2. What is ViT?

❖ Vision Transformer(ViT)

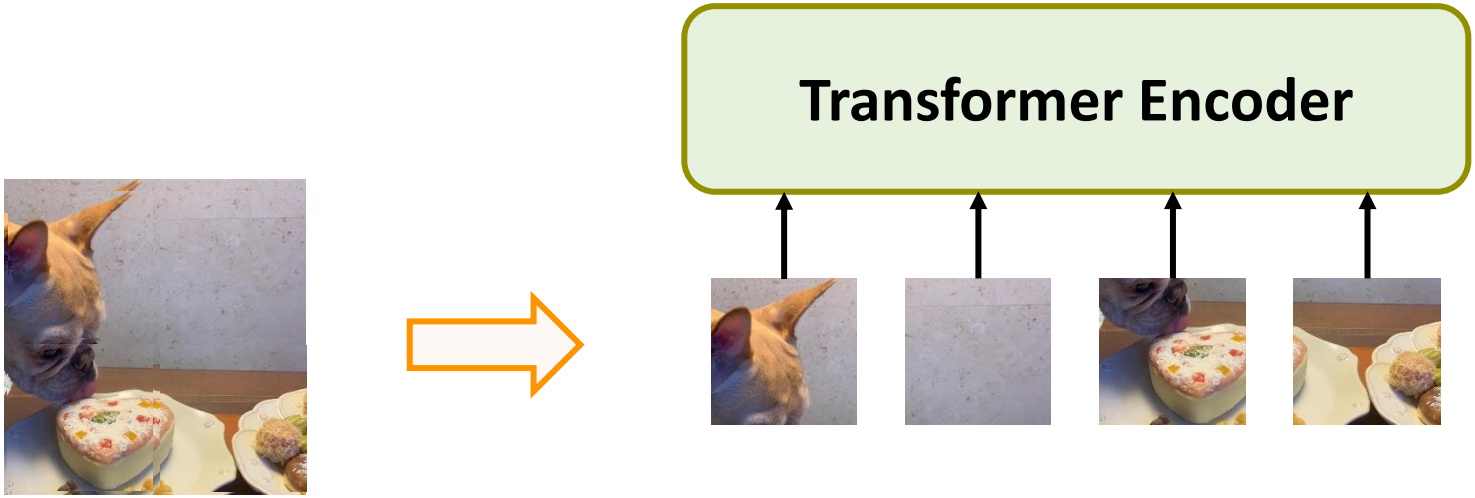
- ➢ 입력 이미지
  - ➢ 기존 Transformer에서는 token embedding을 입력
  - ➢ 이미지를 patch형태로 나눠서 입력
  - ➢ 모든 patch를 벡터와 flatten하게 생성한 뒤 입력



2. What is ViT?

❖ Vision Transformer(ViT)

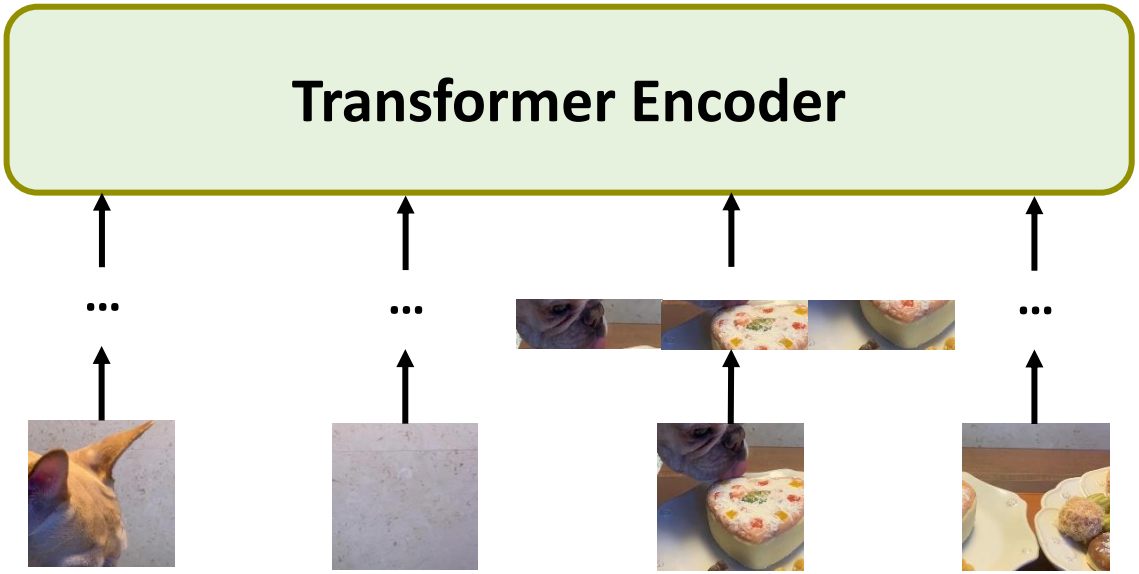
- ➢ 입력 이미지
  - ➢ 기존 Transformer에서는 token embedding을 입력
  - ➢ 이미지를 patch형태로 나눠서 입력
  - ➢ 모든 patch를 벡터와 flatten하게 생성한 뒤 입력



2. What is ViT?

❖ Vision Transformer(ViT)

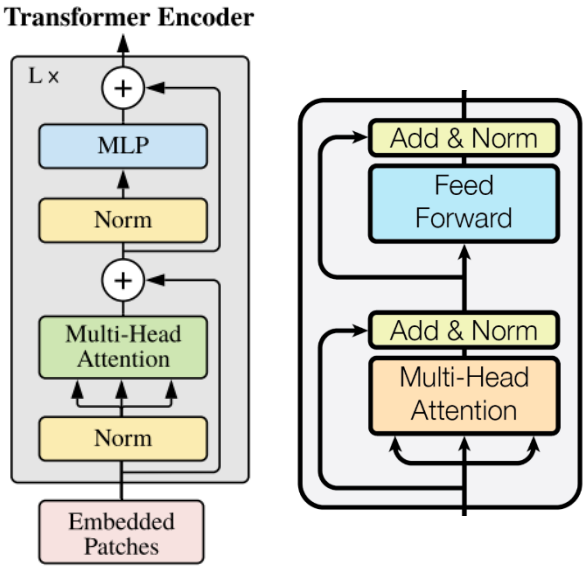
- ➢ 입력 이미지
  - ➢ 기존 Transformer에서는 token embedding을 입력
  - ➢ 이미지를 patch형태로 나눠서 입력
  - ➢ 모든 patch를 벡터와 flatten하게 생성한 뒤 입력



2. What is ViT?

❖ Vision Transformer(ViT)

- ➢ 모델 아키텍처
  - ➢ 기존 Transformer의 encoder부분과 거의 유사함
  - ➢ Transformer와 차이점: Norm의 위치, GELU사용
- ➢ 대용량 데이터의 사전학습
  - ➢ 대용량 데이터셋으로 사전학습을 한 뒤 downstream task로 fine-tuning을 진행



<ViT>

<Transformer>

私は学生です。

Je suis étudiant.

2. What is ViT?

❖ Vision Transformer(ViT)

➢ Result

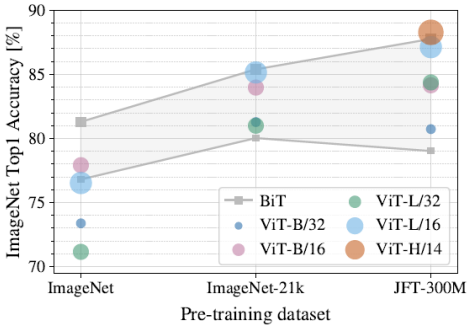
➢ JFT로 사전학습을 진행한 뒤, 각 데이터 셋에 Transfer Learning진행

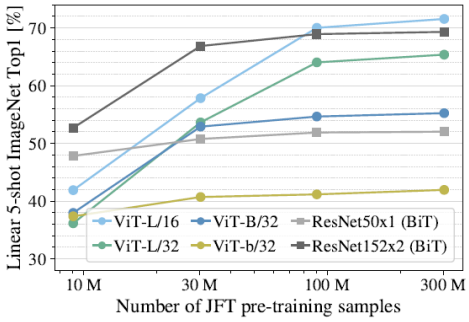
➢ CNN기반의 SoTA모델과 비교해서 ViT가 SoTA를 갱신함

➢ 사전학습을 진행할 경우 데이터 셋이 클수록 ViT성능이 향상

➢ Convolutional Inductive Bias가 존재하지 않기 때문에 많은 데이터가 필요

	Ours-JFT (ViT-H/14)	Ours-JFT (ViT-L/16)	Ours-I21k (ViT-L/16)	BiT-L (ResNet152x4)	Noisy Student (EfficientNet-L2)
ImageNet	**88.55** ± 0.04	87.76 ± 0.03	85.30 ± 0.02	87.54 ± 0.02	88.4/88.5*
ImageNet ReaL	**90.72** ± 0.05	90.54 ± 0.03	88.62 ± 0.05	90.54	90.55
CIFAR-10	**99.50** ± 0.06	99.42 ± 0.03	99.15 ± 0.03	99.37 ± 0.06	–
CIFAR-100	**94.55** ± 0.04	93.90 ± 0.05	93.25 ± 0.05	93.51 ± 0.08	–
Oxford-IIIT Pets	**97.56** ± 0.03	97.32 ± 0.11	94.67 ± 0.15	96.62 ± 0.23	–
Oxford Flowers-102	99.68 ± 0.02	**99.74** ± 0.00	99.61 ± 0.02	99.63 ± 0.03	–
VTAB (19 tasks)	**77.63** ± 0.23	76.28 ± 0.46	72.72 ± 0.21	76.29 ± 1.70	–
TPUv3-core-days	2.5k	0.68k	0.23k	9.9k	12.3k

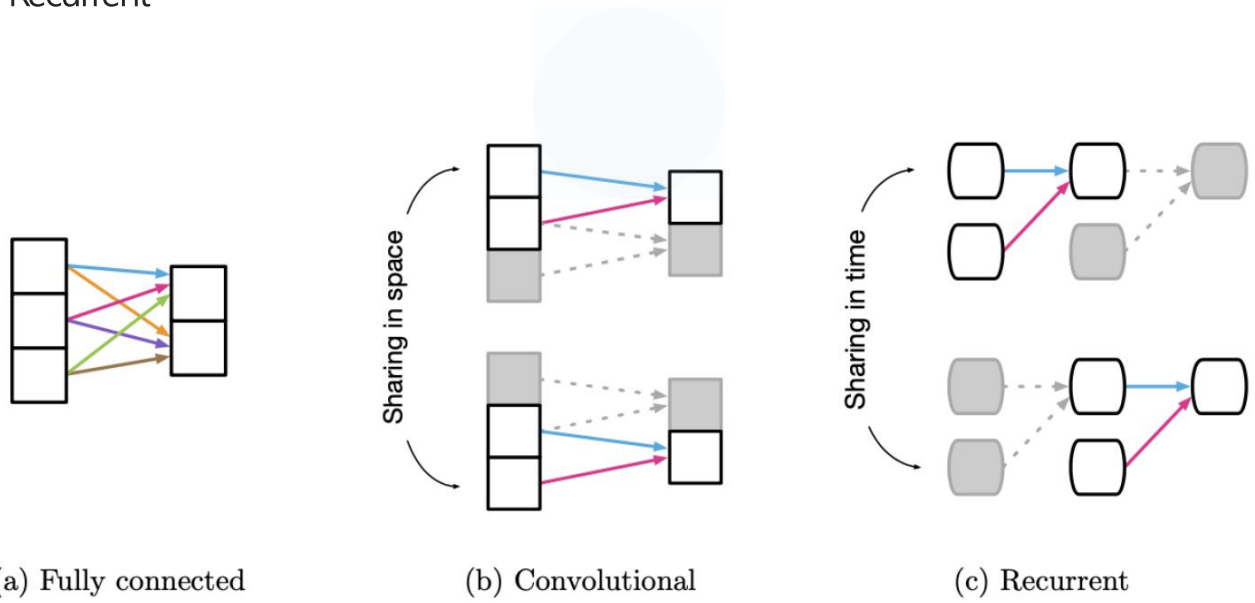




2. What is ViT?

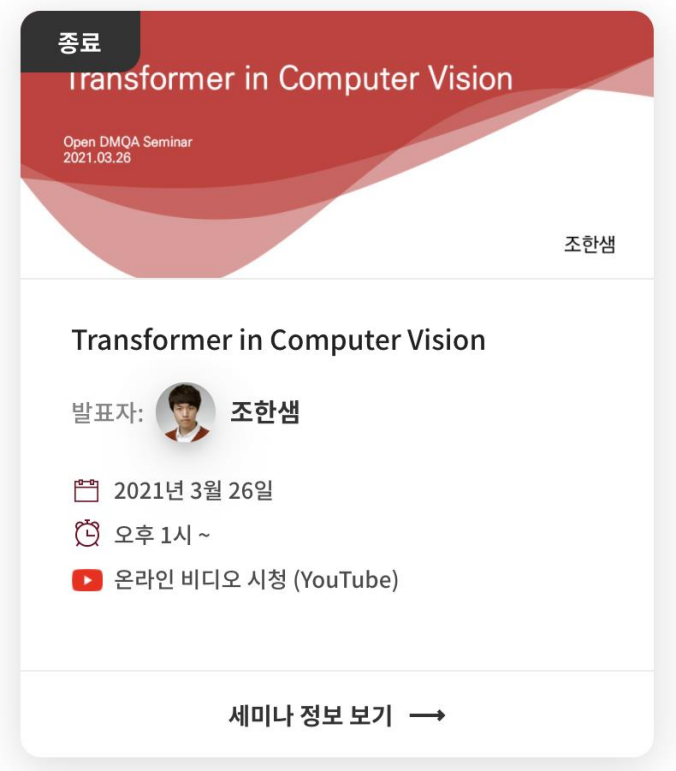
❖ Vision Transformer(ViT)

- ➢ Inductive bias
  - ➢ 학습 모델이 지금까지 만나보지 못했던 상황에서 정확한 예측을 하기 위해 사용하는 추가적인 가정
  - ➢ Fully Connected
  - ➢ Convolutional
  - ➢ Recurrent



https://enfow.github.io/paper-review/graph-neural-network/2021/01/11/relational_inductive_biases_deep_learning_and_graph_netowrks/

2. What is ViT?



http://dmqa.korea.ac.kr/activity/seminar/316

http://dmqa.korea.ac.kr/activity/seminar/295

3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- ➢ 2022년 1월 17일 기준 24회 인용
- ➢ Google Research에서 발표
- ➢ ViT model을 효율적으로 학습시키는 방법론에 대한 비교분석 실험 진행

How to train your ViT? Data, Augmentation, and Regularization in Vision Transformers

Andreas Steiner*, Alexander Kolesnikov*, Xiaohua Zhai*
Ross Wightman†, Jakob Uszkoreit, Lucas Beyer*

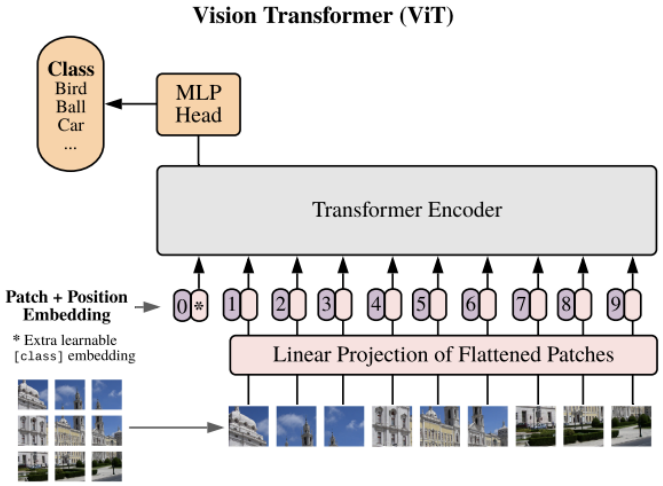
Google Research, Brain Team; †independent researcher

{andstein,akolesnikov,xzhai,usz,lbeyer}@google.com, rwightman@gmail.com

3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- ➢ Models: ViT-Ti, ViT-S, ViT-B , ViT-L
- ➢ Patch-size: 16, 32(ViT-S, ViT-B)
- ➢ 기존 ViT에서 head의 hidden layer를 제거
- ➢ Backbone으로 ResNet을 사용하여 나온 특징벡터를 ViT에 입력(Hybrid models)



<ViT architecture>

Table 1: Configurations of ViT models.

Model	Layers	Width	MLP	Heads	Params
ViT-Ti [34]	12	192	768	3	5.8M
ViT-S [34]	12	384	1536	6	22.2M
ViT-B [10]	12	768	3072	12	86M
ViT-L [10]	24	1024	4096	16	307M

Table 2: ResNet+ViT hybrid models.

Model	Resblocks	Patch-size	Params
R+Ti/16	[]	8	6.4M
R26+S/32	[2, 2, 2, 2]	1	36.6M
R50+L/32	[3, 4, 6, 3]	1	330.0M

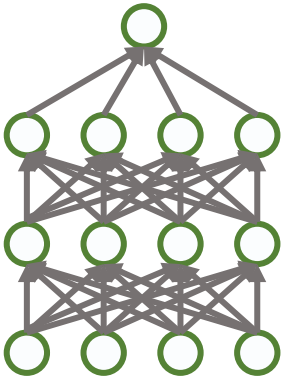
3. How to train your ViT?

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Datasets for pre-training
    - ImageNet-1k
    - ImageNet21k
      - De-duplicate images in ImageNet-21k
  - Datasets to evaluate transfer learning
    - VTAB benchmark
    - CIFAR-100
    - Oxford IIIT Pets
    - Resisc45 and Kittidistance

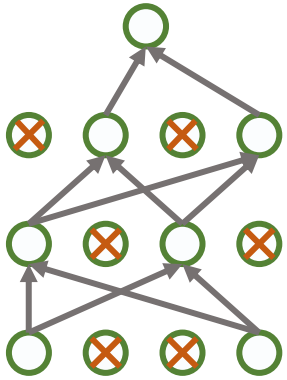
3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- **Regularization**
  - **Dropout to intermediate activations of ViT**
  - Stochastic depth regularization technique
- Data augmentations
  - Mixup
  - RandAugment



<Standard Neural Net>

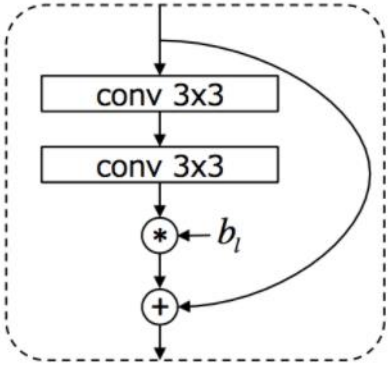


<After applying dropout>

3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- **Regularization**
  - Dropout to intermediate activations of ViT
  - **Stochastic depth regularization technique**
- Data augmentations
  - Mixup
  - RandAugment



<I번째 residual block>

$$pl = 1 - \frac{l}{2L}$$

https://engineer-mole.tistory.com/43

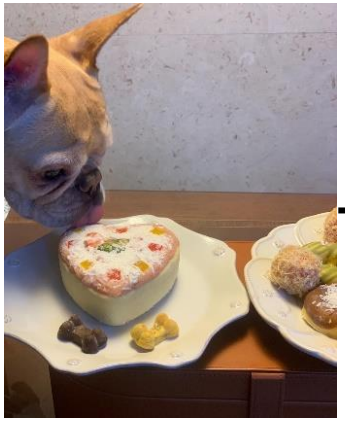
3. How to train your ViT?

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Regularization
    - Dropout to intermediate activations of ViT
    - Stochastic depth regularization technique
  - **Data augmentations**
    - **Mixup**
    - RandAugment

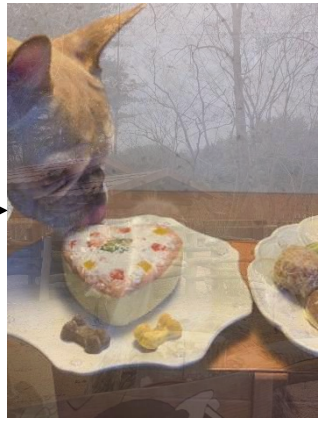
[1, 0]

+

lambda=0.5



[0, 1]



[0.5, 0.5]

3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- ➢ Regularization data augmentations
  - ➢ Dropout to intermediate activations of ViT
  - ➢ Stochastic depth regularization technique
- ➢ **Data augmentations**
  - ➢ Mixup
  - ➢ **RandAugment**

<RandAugment example>

RandAugment: Practical automated data augmentation with a reduced search space

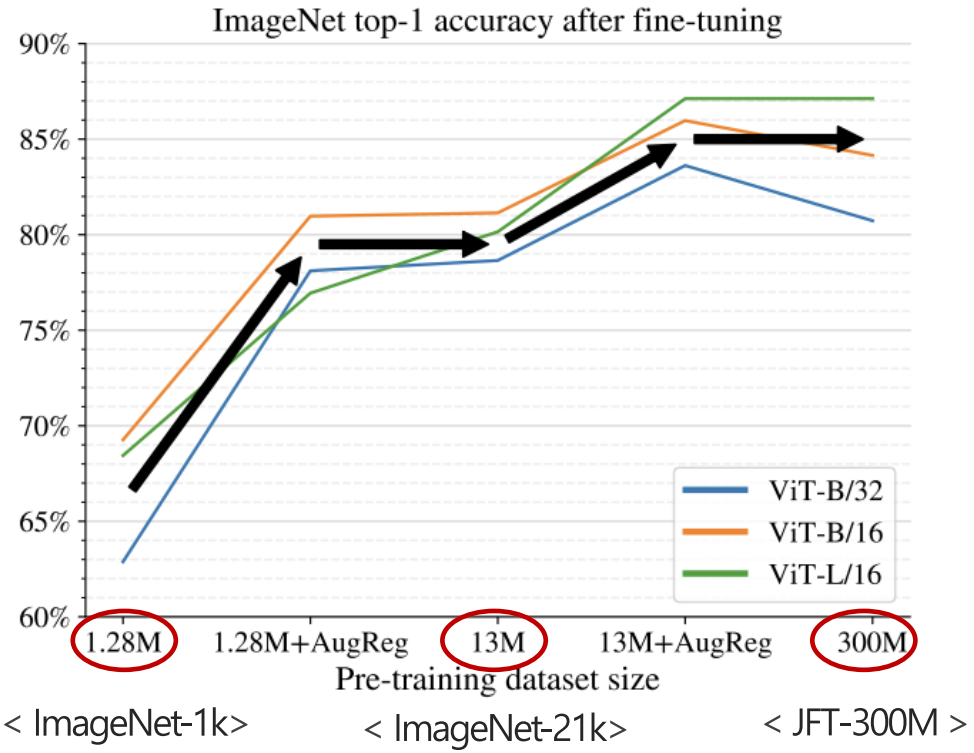
3. How to train your ViT?

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Pre-training
    - Optimizer: Adam
    - Batch size: 4096
    - Cosine learning rate schedule
    - Image pre-process: Inception-style cropping, random horizontal flipping
    - Epoch: ImageNet-1k(300), ImageNet-21k(30 & 300)
  - Fine-tuning
    - Optimizer: SGD
    - Batch size: 512
    - Cosine decay learning rate schedule

3. How to train your ViT?

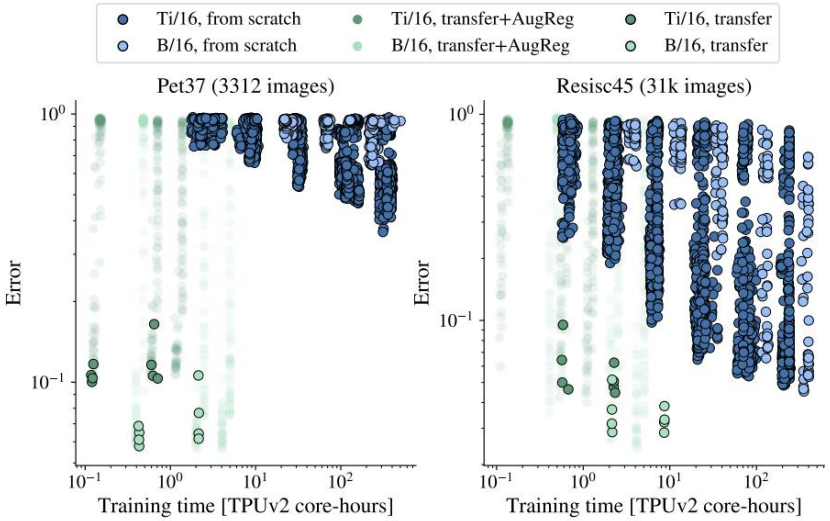
❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

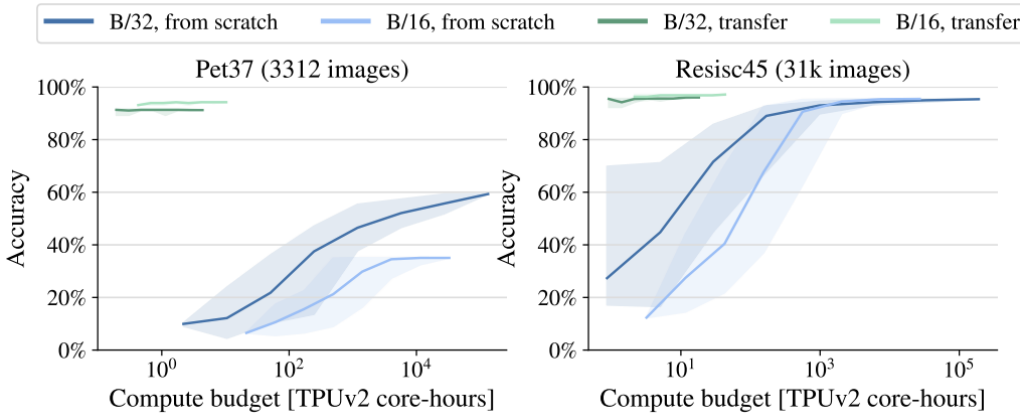
- ➢ Scaling datasets with AugReg(augmentation, Reugularization) and compute
  - ➢ AugReg ImageNet-1k perform about equal to the same models pre-trained ImageNet-21k dataset
  - ➢ AugReg ImageNet-21k perform about equal to the same models pre-trained plain JFT-300M dataset



3. How to train your ViT?

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Transfer is the better option
    - Transferring a pre-trained model is both more cost-efficient and leads to better results
    - It does not seem possible to train ViT models from scratch to reach accuracy anywhere near that of transferred models
    - The ViT models from scratch are impossible to train with the accuracy of transferred models

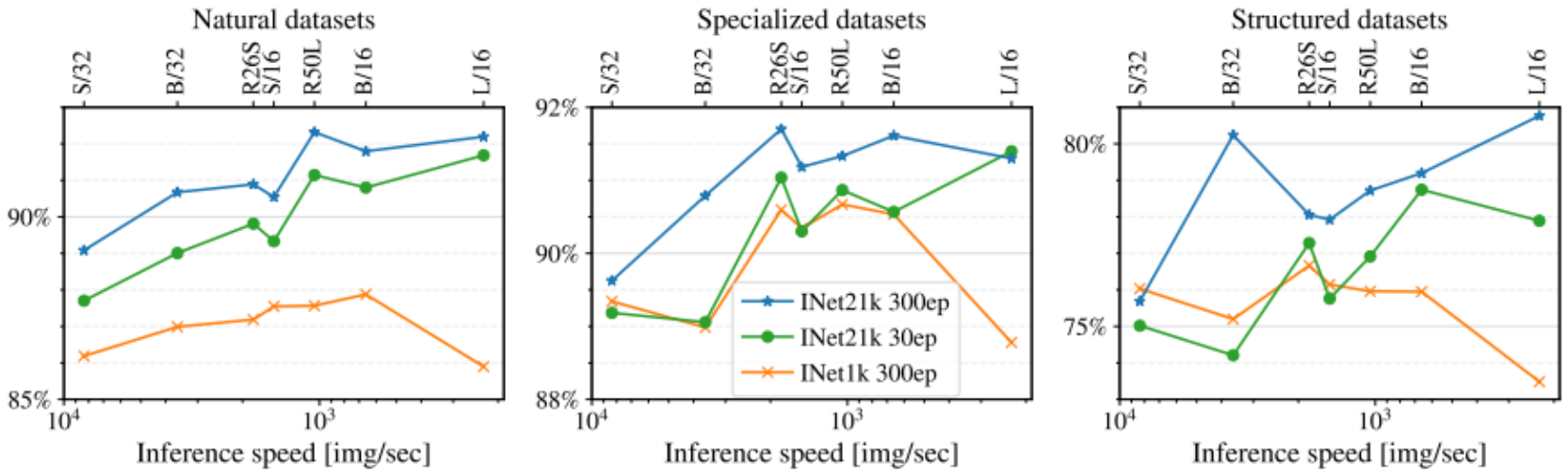




3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

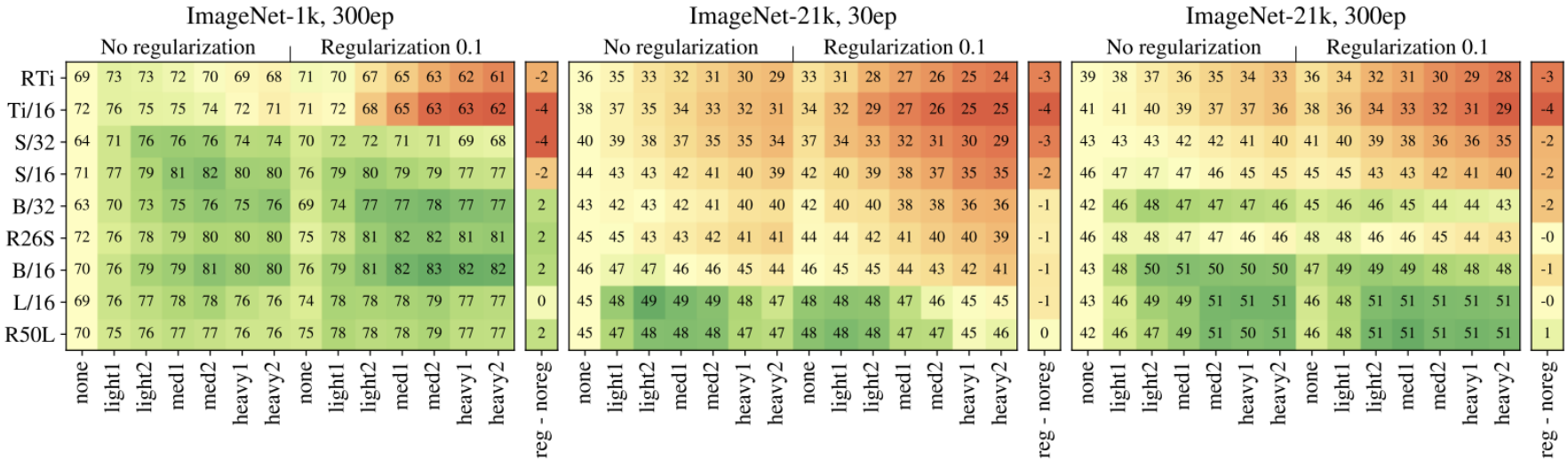
- ➢ More data yields more generic models
  - ➢ Model pre-trained on ImageNet-21k(30 epochs) is better than the ImageNet-1k(300 epochs)
  - ➢ **More data yields more generic models**



3. How to train your ViT?

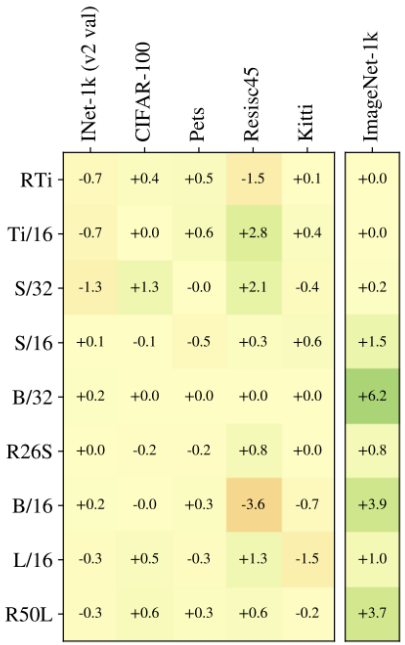
❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- ➢ Prefer augmentation to regularization
  - ➢ Mid-sized ImageNet-1k dataset, any kind of AugReg helps
  - ➢ ImageNet-21k dataset and keeping compute fixed, any kind of AugReg hurts performance for all but the largest models



3. How to train your ViT?

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Choosing which pre-trained model to transfer
    - **How to select a model for further adaption for an end application?**
    - Run downstream adaptation for all available pre-trained models
    - One can select a single pre-trained model based on the upstream validation accuracy, which is much cheaper
    - Cheaper strategy works equally well as the more expensive strategy in the majority of scenarios

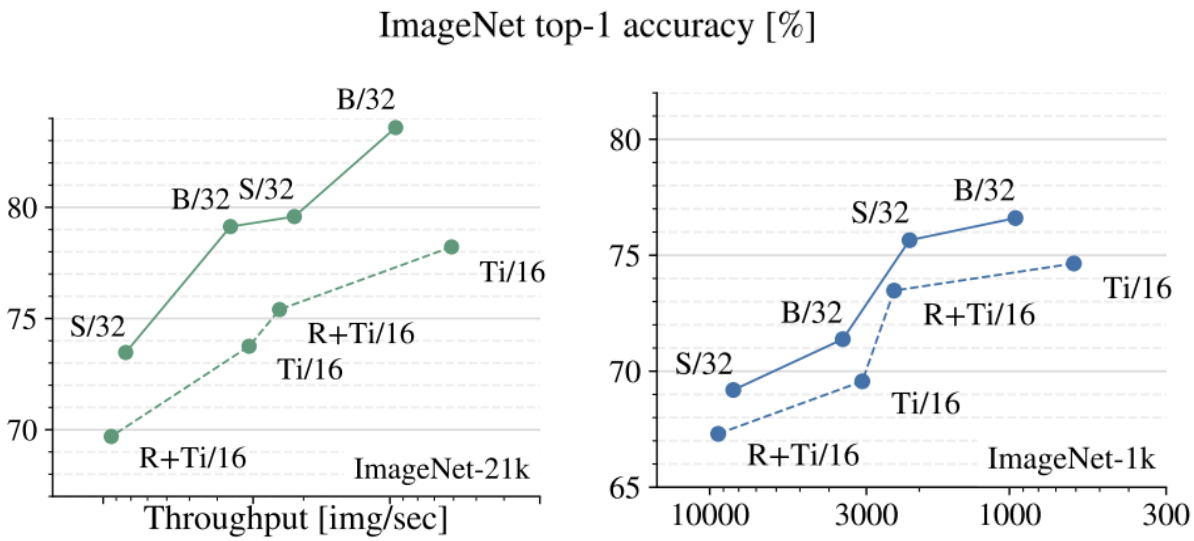


	INet-1k (v2 val)	CIFAR-100	Pets	Resisc45	Kitti	ImageNet-1k
RTi	-0.7	+0.4	+0.5	-1.5	+0.1	+0.0
Ti/16	-0.7	+0.0	+0.6	+2.8	+0.4	+0.0
S/32	-1.3	+1.3	-0.0	+2.1	-0.4	+0.2
S/16	+0.1	-0.1	-0.5	+0.3	+0.6	+1.5
B/32	+0.2	+0.0	+0.0	+0.0	+0.0	+6.2
R26S	+0.0	-0.2	-0.2	+0.8	+0.0	+0.8
B/16	+0.2	-0.0	+0.3	-3.6	-0.7	+3.9
L/16	-0.3	+0.5	-0.3	+1.3	-1.5	+1.0
R50L	-0.3	+0.6	+0.3	+0.6	-0.2	+3.7

3. How to train your ViT?

❖ How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers

- ➢ Prefer increasing patch-size to shrinking model-size
  - ➢ "Tiny" variants perform significantly worse than the similarly fast larger models with "/32" patch-size
  - ➢ **Parameter count is reflective neither of speed, nor of capacity**



4. Conclusion

❖ 결론

- ViT
  - Computer vision분야에서 합성곱 신경망이 아닌 Transformer를 적용
  - Transformer architecture가 computer vision분야에서 수 많은 SOTA를 달성

- How To Train Your ViT? Data, Augmentation, and Regularization in Vision Transformers
  - Regularization과 augmentation만 잘 써도 적은 데이터로 비슷한 성능을 도출 할 수 있음
  - Downstream task시 대용량 데이터로 사전 학습한 모델로 transfer learning하는 것이 좋음
  - Transfer learning시 대용량 데이터로 사전학습한 모델의 성능이 좋음
  - 데이터가 많이 존재할 경우 regularization과 augmentation은 효과가 거의 없음
  - 사전학습 모델이 많은 경우 upstream에서 가장 성능 좋은 모델을 사용하면 좋음
  - Inference의 속도가 중요한 경우 patch size를 키우면 성능이 유지됨

감사합니다.