Possible Question

(그래프에는 없지만) 앙상블해서 정확도가 떨어진 이유?

어떤 방식으로 추가한 ImageNet 데이터가 DL20의 test data와 겹치지 않게 했는가?

Comparing all numpy array was too copmuationally intensive for us. So, we apply hash function to whole dataset and compare each hash with the other to check whether they contain the same data.

Randaugment 혹은 super-resolution 방법들에 대한 간략한 설명?

Q1) ImageNet에서 데이터를 추가한 이유는 무엇인가? 무조건 정확도가 좋아질거 같은데? 반칙 아닌가?
과제에 대한 설명에서 어떤 것은 하면 안 된다는 조건은 없었다.그렇다면 여러 복잡하고 화려한 테크닉을 적용하기 이전에, 수업시간에 자주 들었던 것처럼 가장 기본적인 것에서 출발하는 것이 중요하다고 생각했다. 대표적으로 데이터의 질과 양이 성능향상에 가장 중요한 요소라고 판단했다.
To be clear, there is no statement that use of external data is prohibited. Moreover, Prof. Jaesik Park answered that “anything” is allowed except cheating. And I believed TA aleardy answered in QnA board that external data is allowed. Our team thoroughly discuss should we use this external data or not and our conclusion was that (쉬고) before using some fancy and complex model and technique, why don’t we focus a basic of Deep learning : high quality of data promise better performance.

Q2) EfficientNet와 Vision Transform 모델 2개만을 선택하여 학습한 특별한 이유가 있는가?
우리가 이 프로젝트에서 확인하고 싶었던 부분은 CNN과 Transformer를 이미지데이터에 적용했을 시 서로 다른 특징을 학습하는가여부였다. 그래서 CNN과 Transformer의 대표적인 모델을 택해서 그 2개만을 비교실험했다.

What we want to see is whether CNN and Transformer capture different features, so we picked best model in CIFAR10 and ImageNet benchmark. Even ViT huge model perform better according to the paper that we reference, the model parameter is twice as much compare to ViT Large.

Q3) Super resolution을 했을때 사람눈으로 보기에는 화질이 좋아서 학습이 더 잘될거 같은데 실제 모델결과물에서는 적용하지 않았을때 정확도가 더 잘나왔다. 이유가 무엇인가?

Q4) 앙상블 모델을 만들때 기본적으로 3개 이상의 모델을 조합하는데 2개만 조합하게 되었을때
모델의 결과물에 대한 투표가 50대 50이되면.. 문제가 되는거 아닌가?

문제가 된다. 그럼에도 불구하고 2개만 택한 이유는 CNN과 Transformer의 차이를 보다 분명히 확인하기 위해서였다. 3개이상을 쓸 경우 CNN과 Transformer의 갯수가 1:1에서 달라지게 되는데, 그렇게 되면 어느 한쪽이 우세하게 되어 동일한 조건에서 비교하기가 어렵다고 판단했기 때문이다.
We acknowledge this kind of (쉬고) let me say “psudeo”-ensemble is problematic. But, here is the problem, if we choose, odd number of model, one type of architecture contribute more in prediction, thus it is hard to reflect their discrete characteristics.

Q5) Epoch이 높아질수록 validation loss가 증가하는 문제를 해결하기 위해 적용한 방법이 있는가?

We use intensive amount of data augmentation. Most of them are reported that works well with ImageNet Data. As we mentioned at the beginning, knowing that our dataset is indeed subset of ImageNet, our hypothesis is techniques that worked well in ImageNet, also show good performance in DL20.

Q6) 각각의 augmentation 기법중에서 Rand Augment를 적용했을때 가장 성능이 좋았는데 이유가 무엇이라고 생각하는가?

Again, reminding that our data is subset of ImageNet our team has two choices: AutoAugment, and RandAugment. For autoaugment, we can reuse pretrained Autoaugment policy for ImageNet. However, proposed in RandAugment paper, data augmentation highly depend on the model and dataset size. And, solution that we found in proxy task we not the optimal solution. Since, our data distribution is different that original ImageNet, since 980 classes are gone, using RandAugment will more give more generalized performance in our task.

Q7) 더 높은 성능을 낼수 있도록 모델을 좀더 보완한다면 어떠한 시도를 더 할 수 있을것이라고 생각하는가?

Q8) Vision Transformer는 데이터가 많은 상황에서 높은 성능을 내는 모델로 알려져있는데 우리 과제는 데이터가 상대적으로 많지 않은 경우였다. ViT를 baseline으로 고른 이유는?

우리는 CNN과 Transformer의 차이를 비교하기 위해 두 모델다 실험을 했었다. 또한 DL20데이터를 분석하고 test set과 겹치지 않는 것들로 imagenet에서 데이터를 추가했다. 따라서 많은 데이터를 가지고 학습시킬 수 있었고 이런 상황에서 유리한 ViT가 높은 성능을 보여 baseline 모델로 삼게 되었다.

Hi, this is team E-CAT, and we choose image classification for our final project. And we ranked 1st place in both public and private leaderboard. Our main topic was “Can transformer and CNN capture different feature from image?”

We all know CNN architecture is standard for Image Classification. But, recently transformer based model comes into image classificatixon and outperform some CNN network. During our class, we learned effective ensemble comes from diverse models. Inspired by those facts, if two different architecture capture diverse features, we hypothesis that our ensemble model can get generalized prediction for all class of image. As our baseline, we use vision transformer and effcientnet.

During data preprocessing phase, our team found that our dataset was subsampled from ImageNet dataset. According to the paper below “Do better ImageNet models transfer better?,” it is well known fact that initializing from pretrained model will allow fast convergence and better performance. So, we initialized our model from ImageNet pretrained. Compared to our given data, ImageNet data has much higher resolution. From this, we hypothesized the following: ImageNet data has higher resolution, if so, ImageNet pretrained weight will perform best in those resolutions. Then, if we can increase our data’s resolution, our model will converge faster. To validate our hypothesis, GAN-based super resolution techniques are employed. Sadly, those super-resolution preprocessing doesn’t improve our model. On the right example, you can see RealSR improved the resolution. But, in some other example, super-resolutioning hurts important feature of image, which is our reasoning for performance degradation.

Also our team exploited three techniques and compared the accuracy. Three techniques are, using pre-trained weights, RandAugmentation, external data from ImageNet. The accuracies of models were improved, so we decide to use these three techniques. As I mentioned earlier, we found that our target dataset was subset of ImageNet, and there are more images belongs to that classes. The critical things that we considered during data addition was not including test data to train or validation set. In order to manage this, our team carefully investigate our training data to have no duplicates.

We trained many types of EfficientNet and ViT. The best, among EffNet was EffNet-v2, with 97.91% top-1 accuracy. For vision transformers, ViT-L/16 was the best with 99.14%, top-1 accuracy.

To analyze the results, we investigate the number of misprediction of two models respect to classes. Although ViT generally make fewer errors, EffNet-v2 outperformed on some classes.

So it seems to be reasonable to ensemble these two models according to the pipeline in the poster. However, ensembling failed to achieve better accuracy.

Hence, our team selected the ViT-L/16 model as the final model. Also, since the lack of poster space, we don't include training result for other augmentation technique that we used. All of that are listed in pseudo code. Augmentations includes color jitter, label smoothing, random interpolation, hfilp, mixup and cutmix. The interesting augmentation that we used is mixup and cutmix for all training data, and switch between mixup and cutmix with probability of 0.5.

In overall, we exploit how CNN and transformer capture different features of the data. The key thing to remind is ViT-L/16 has roughly 300 millon paramters, where EffNet has 121 million parameters. Keeping mind of this, 97.9% of EffcientNet-v2-large was impressing. If we are able to train much larger CNN based architecture, our ensemble may improve the performance.