

Transfer Learning

Grigory Sapunov  CTO / Intento

Transfer Learning

Что делать, когда большой базы картинок нет?

Или когда нет возможности обучать модель неделями?

Можно взять готовую модель и дообучить.

Модели распространяются так же, как и код. Существуют репозитории готовых моделей:

- Keras <https://keras.io/applications/>
- TF Hub <https://tfhub.dev/>
- Model Zoo/Caffe http://caffe.berkeleyvision.org/model_zoo.html
- Community models/Caffe <https://github.com/BVLC/caffe/wiki/Model-Zoo>
- <https://modelzoo.co/>

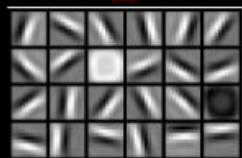
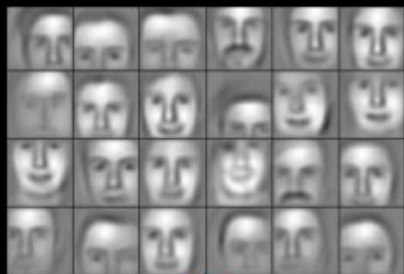
Transfer Learning

Почему это должно работать?

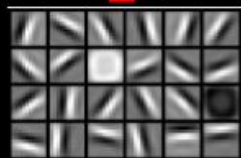
Потому что модель, обученная на большом датасете и на много классов, скорее всего выучила общие закономерности в изображениях, которые будут полезны и для других задач классификации, где объекты примерно похожи.

Кроме того, модели имеют тенденцию выучивать более общие признаки на нижних уровнях, и более задаче-специфичные на верхних.

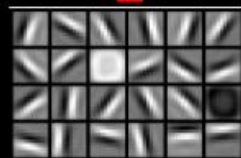
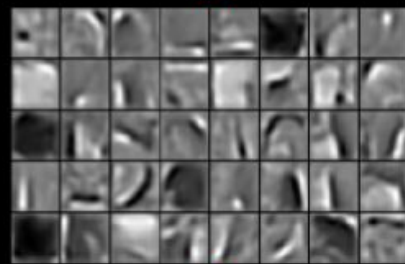
Faces



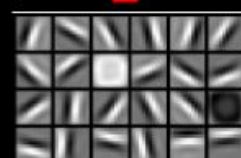
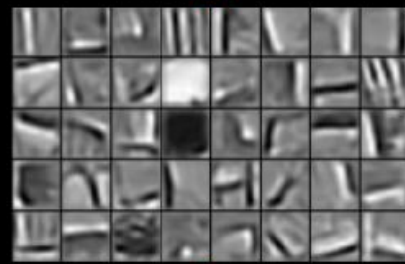
Cars



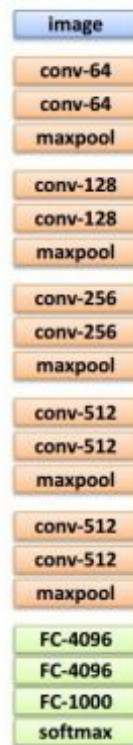
Elephants



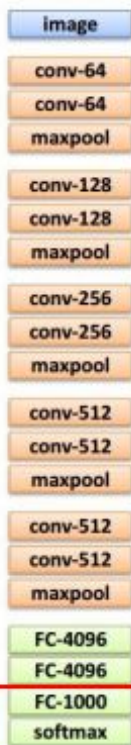
Chairs



Transfer Learning with CNNs

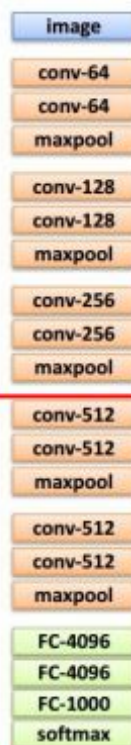


1. Train on ImageNet



2. If small dataset: fix all weights (treat CNN as fixed feature extractor), retrain only the classifier

i.e. swap the Softmax layer at the end



3. If you have medium sized dataset, “**finetune**” instead: use the old weights as initialization, train the full network or only some of the higher layers

retrain bigger portion of the network, or even all of it.

(Colab)

Transfer learning в TF/Keras

Практика Transfer Learning

- Мало данных:
 - Новый датасет похож на оригинальный: обучаем только классификатор (как в примере).
 - Новый датасет не похож: тогда верхние полносвязные слои скорее всего заточены на оригинальный датасет, лучше взять какой-то из более ранних слоёв и SVM.
- Много данных:
 - Новый датасет похож: обучаем больше слоёв, в пределах все (Fine-tuning).
 - Новый датасет не похож: можно обучать новую сеть с нуля, но можно и fine-tuning по всей сети.
- Обычно уменьшаем глобальный learning rate. Для отдельных слоёв learning rate варьируем: новым больше, старым меньше.
Идея в том, чтобы старая часть модели менялась слабо, а новая обучалась быстро.
- Также обычно меняем процедуру обучения, чтобы учесть размеры данных.

Другие использования обученной модели

- Готовую модель можно использовать для генерации признаков для других алгоритмов машинного обучения
 - Для этого можно снять значения с выхода верхних слоёв обученной модели (с оторванным слоем классификации) и использовать их дальше как готовый вектор признаков (например, для SVM).
 - Этот вектор можно также использовать для поиска похожих изображений.
- Модель можно использовать для построения хитрой функции потерь (перенос стиля, будет дальше)

Ресурсы

- <http://cs231n.github.io/transfer-learning/>
- https://www.tensorflow.org/tutorials/images/transfer_learning
- https://www.tensorflow.org/tutorials/images/transfer_learning_with_hub
- <https://www.packtpub.com/books/content/transfer-learning>
- http://caffe.berkeleyvision.org/gathered/examples/finetune_flickr_style.html
- <https://habrahabr.ru/company/mailru/blog/252965/>
- <https://medium.com/towards-data-science/transfer-learning-using-keras-d804b2e04ef8>
- <https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html>
- https://www.tensorflow.org/js/tutorials/transfer/what_is_transfer_learning