Факультет инноваций и высоких технологий

Efficient Net

Сиркиза Евгений
15 января 2021 г.

1 Введение

Сверточные нейронные сети (ConvNets) обычно разрабатываются при фиксированном количестве ресурсов, а затем масштабируются для большей точности, при увеличинии ресурсов. Масшиабирование ConvNets производятся по трем направлениям: по глубине, по ширине и по разрешению. Масштабирование часто используется для увелиения точности модели, хотя сам процесс маштиброваия не был до конца изучен, поэтому обычно масштабируется только одно из трех измерений - глубина, ширина или разрешения изображения.

В этой работе продемонстрировал другой подход к масшатибованию ConvNets. вторами оригинальной стаьи Efficient Net было эмпирически показано, что важно сбалансировать все измерения (глубину, ширину и разрешения) для получения лучшей точности и эффективности. И что удивительно, такого баланса можно достичь, просто масштабируя каждый измерений с постоянным соотношением. В частности, EfficientNet-B7 обеспечивает top-1 точность $84,3\%$ в ImageNet, при этом он в $8,4$ раза меньше и в $6,1$ раза быстрее, чем лучший из существующих ConvNet. Такой результат был достигнут благодаря простому, но эффективному методу составного масштабирования. На рисунке 1.1 показана разница между методом составного масшиабирования и обычными методами.

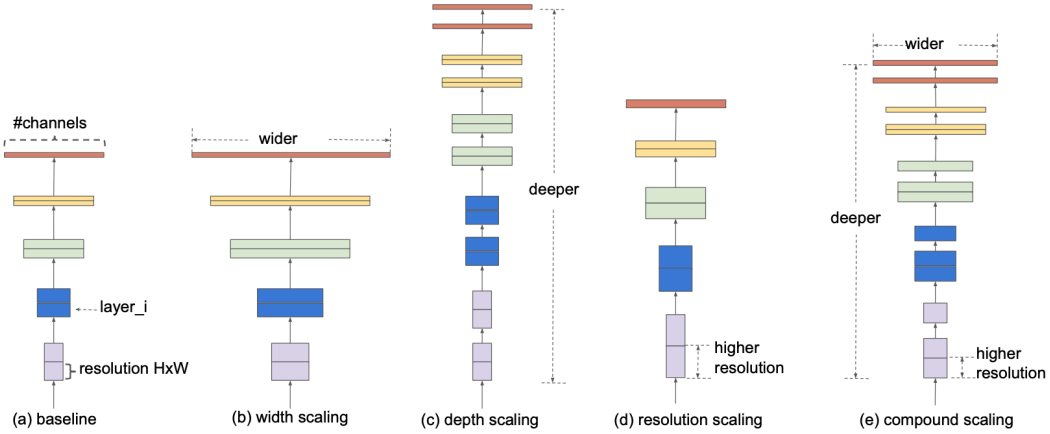

Рис. 1.1: (a) - пример базовой сети; (b) - (d) - это обычное масштабирование, которое увеличивает только одно измерение сети: ширины, глубины или разрешения. (e) - это предлагаемый метод составного масштабирования, который равномерно масштабирует все три измерения с фиксированным соотношением.

2 Составное масштабирование

Как уже было сказано выше для увеличения точности обычно масштабирование производят по одному из измерений. Рассмотрим каждое из них:

- Глубина (d). Интуиция которая лежит в основе масштабирования глубины связанна с тем, что чем глубже сверточная сеть, тем более сложные закономерности. Однако более глубокие сети также труднее обучать из-за проблемы исчезающего градиента. График зависимости качества при увеличении глубины от вычислительной мощности измеренной в FLOPS показан на рисунке 2.1 по средине.

- Ширина (w). В основе этого метода лежит лаблюдение, что более широкие сети, как правило, способны улавливать более мелкие функции и их легче обучать. Однако слишком широкие сети испытывают трудности в нахождении сложных закономерностей. График зависимости качества при увеличении ширины от вычислительной мощности измеренной в FLOPS показан на первом рисунке 2.1.

- Разрешние (r). Чем больше разрешение входного изображения тем легче ConvNet захватывать более мелкие закономерности и шаблоны. График зависимости качества при увеличении разрешения от вычислительной мощности измеренной в FLOPS показан на последнем рисунке 2.1.

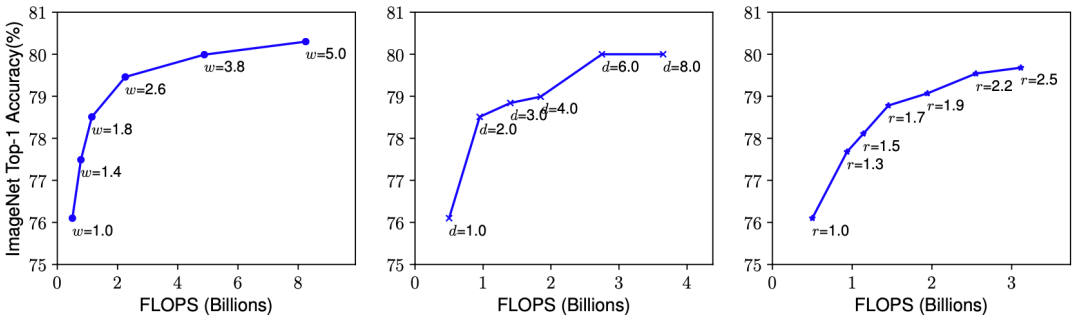

Рис. 2.1: Масштабирование базовой сети с различными коэффициентами ширину (w), глубину (d) и разрашение (r). Более крупные сети с большей шириной, глубиной или разрешением, как правило, обеспечивают более высокую точность, но прирост точности быстро насыщается после достижения 80%, демонстрируя ограниченность одномерного масштабирования.

Описанные выше наблюдения приводят к следующему наблюдению: "Увеличение любого измерения ширины, глубины или разрешения сети повышает точность, но прирост точности уменьшается с увеличением сети."

Также авторы статьи эмпирическим методом пришли к еще одному наблюдению: "Чтобы добиться большей точности и эффективности, очень важно сбалансировать все измерения (ширину, глубину и разрешение) сети."

Таким образом, эти два наблюдения позволяют сформулировать принцип составного масштабирования, в котором используется составной коэффициент ϕ для равномерного масштабирования всех измерений:

$$
\begin{aligned}
d &= \alpha^{\phi} \\
w &= \beta^{\phi} \\
r &= \gamma^{\phi} \\
\text{s.t} \quad & \alpha\beta^2\gamma^2 \approx 2 \\
& \alpha \geq 1, \beta \geq 1, \gamma \geq 1
\end{aligned}
$$

где α, β, γ - константы которые определяются небольший grid search. Коээфициент ϕ по сути параметр, который контролирует, сколько дополнительных ресурсов доступно для масштабирования модели. Поскольку FLOPS пропорционально величине dr^2w^2, то FLOPS итоговой модели для любого ϕ увеличивается примерно в 2^{ϕ} раз.

3 Архитектура EfficientNet

Авторы статьи разработали базовую сеть, используя многоцелевой поиск нейронной архитектуры, который оптимизирует как точность, так и FLOPS. В таблице 3.1 показана архитектура базовой сети EfficientNet-B0.

Стадия	Слой	Разрешение	#Канналов	#Слоев
1	Conv3x3	224 × 224	32	1
2	MBConv1, k3x3	112 × 112	16	1
3	MBConv6, k3x3	112 × 112	24	2
4	MBConv6, k5x5	56 × 56	40	2
5	MBConv6, k3x3	28 × 28	80	3
6	MBConv6, k5x5	14 × 14	112	3
7	MBConv6, k5x5	14 × 14	192	4
8	MBConv6, k3x3	7 × 7	320	1
9	Conv1x1 & Pooling & FC	7 × 7	1280	1

Таблица 3.1: Базовая сеть EfficientNet-B0 - Каждая строка описывает i-ый этап со слоями в пределах заданного разрешения и количеством выходных каналов.

Начиная с базового модели EfficientNet-B0, мы применяем наш метод составного масштабирования в два этапа:

1. Фиксируем $\phi = 1$, предполагая, что мы имеем в два раза больше ресурсов, и выполняем небольшой поиск по сетке α, β, γ. В частности, для для EfficientNet-B0 были подобраны сделующие значения: $\alpha = 1,2$, $\beta = 1.1$, $\gamma = 1.15$ при условии что $\alpha\beta^2\gamma^2 \approx 2$.

2. Затем мы фиксируем α, β, γ как константы и масштабируем базовую сеть с другим ϕ.

Результаты второго этапа показаны на рисунке 3.1.

Model	Top-1 Acc.	Top-5 Acc.	#Params	Ratio-to-EfficientNet	#FLOPs	Ratio-to-EfficientNet
EfficientNet-B0	**77.1%**	**93.3%**	**5.3M**	**1x**	**0.39B**	**1x**
ResNet-50 (He et al., 2016)	76.0%	93.0%	26M	4.9x	4.1B	11x
DenseNet-169 (Huang et al., 2017)	76.2%	93.2%	14M	2.6x	3.5B	8.9x
EfficientNet-B1	**79.1%**	**94.4%**	**7.8M**	**1x**	**0.70B**	**1x**
ResNet-152 (He et al., 2016)	77.8%	93.8%	60M	7.6x	11B	16x
DenseNet-264 (Huang et al., 2017)	77.9%	93.9%	34M	4.3x	6.0B	8.6x
Inception-v3 (Szegedy et al., 2016)	78.8%	94.4%	24M	3.0x	5.7B	8.1x
Xception (Chollet, 2017)	79.0%	94.5%	23M	3.0x	8.4B	12x
EfficientNet-B2	**80.1%**	**94.9%**	**9.2M**	**1x**	**1.0B**	**1x**
Inception-v4 (Szegedy et al., 2017)	80.0%	95.0%	48M	5.2x	13B	13x
Inception-resnet-v2 (Szegedy et al., 2017)	80.1%	95.1%	56M	6.1x	13B	13x
EfficientNet-B3	**81.6%**	**95.7%**	**12M**	**1x**	**1.8B**	**1x**
ResNeXt-101 (Xie et al., 2017)	80.9%	95.6%	84M	7.0x	32B	18x
PolyNet (Zhang et al., 2017)	81.3%	95.8%	92M	7.7x	35B	19x
EfficientNet-B4	**82.9%**	**96.4%**	**19M**	**1x**	**4.2B**	**1x**
SENet (Hu et al., 2018)	82.7%	96.2%	146M	7.7x	42B	10x
NASNet-A (Zoph et al., 2018)	82.7%	96.2%	89M	4.7x	24B	5.7x
AmoebaNet-A (Real et al., 2019)	82.8%	96.1%	87M	4.6x	23B	5.5x
PNASNet (Liu et al., 2018)	82.9%	96.2%	86M	4.5x	23B	6.0x
EfficientNet-B5	**83.6%**	**96.7%**	**30M**	**1x**	**9.9B**	**1x**
AmoebaNet-C (Cubuk et al., 2019)	83.5%	96.5%	155M	5.2x	41B	4.1x
EfficientNet-B6	**84.0%**	**96.8%**	**43M**	**1x**	**19B**	**1x**
EfficientNet-B7	**84.3%**	**97.0%**	**66M**	**1x**	**37B**	**1x**
GPipe (Huang et al., 2018)	84.3%	97.0%	557M	8.4x	-	-

Рис. 3.1: Результаты производительности EfficientNet в ImageNet. Все модели EfficientNet масштабируются из базовой EfficientNet-B0 с использованием разных составных коэффициентов ϕ. ConvNets с похожей точностью топ-1/топ-5 сгруппированы вместе для сравнения эффективности. Масштабируемые модели EfficientNet сокращают и количесвто параметров и количество операций FLOPS на порядок (до 8,4 раза меньше параметров и до 16 раз меньше FLOPS), чем существующие ConvNets.

4 Выводы

В этой работе мы рассмотрели различные способы масштабирование ConvNet и выяснили, что лишь баланс ширины, глубины и разрешения моделей позволяет нам повысить результирующую точность и эффективность. Для достижения этого баланс блы предложен метод составного масштабирования, который позволяет нам легко масштабировать базовую сеть в соответствии с любыми ограничениями целевых ресурсов, сохраняя при этом эффективность модели. Используя предложенный метод, было продемонстрированно, что модель EfficientNet мобильного размера может быть масштабирована очень эффективно, превосходя современную точность с помощью на порядок меньшего количества параметров и количества операций FLOPS в ImageNet.