

# Часть 1

В первой части мне нужно было придумать архитектуру сверточной нейронной сети для классификации изображений на тензорфлоу.

Архитектура выглядит так:

[Свертка + Макс пулинг + Релу] x3 + Дропаут + Линейный слой + Софтмакс

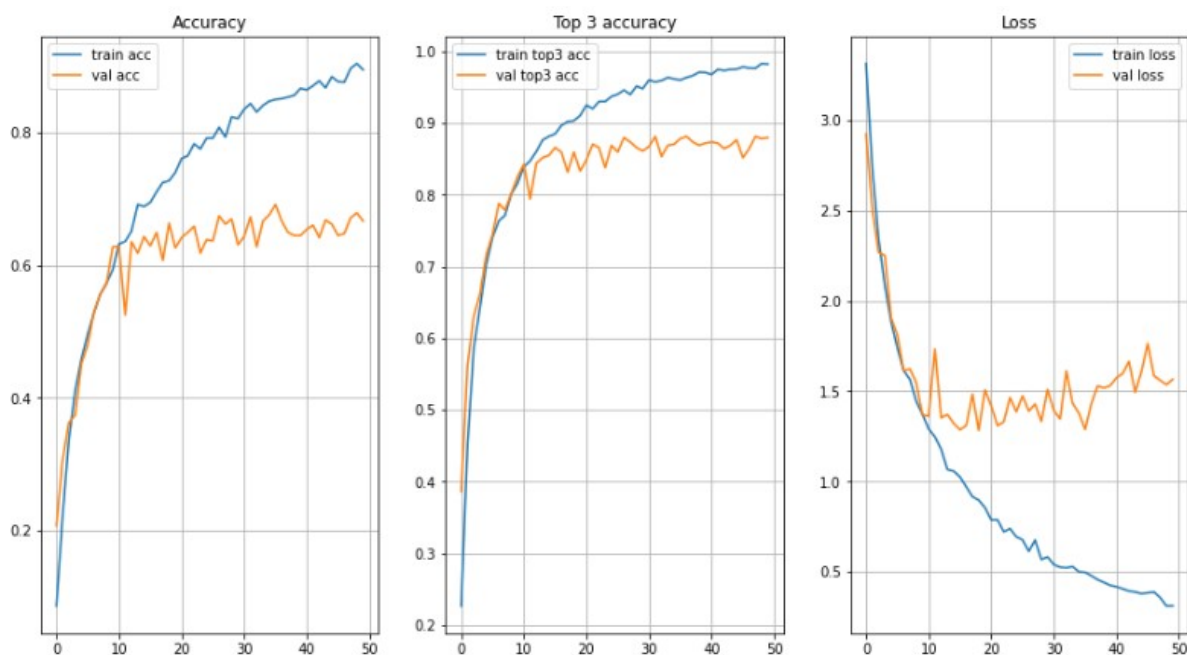
Перед обучением выполняется аугментация изображений (отражение относительно оси y, поворот или зум с вероятностью 0.1) и нормировка (приведение всех чисел к интервалу [0; 1]).

Метрики:

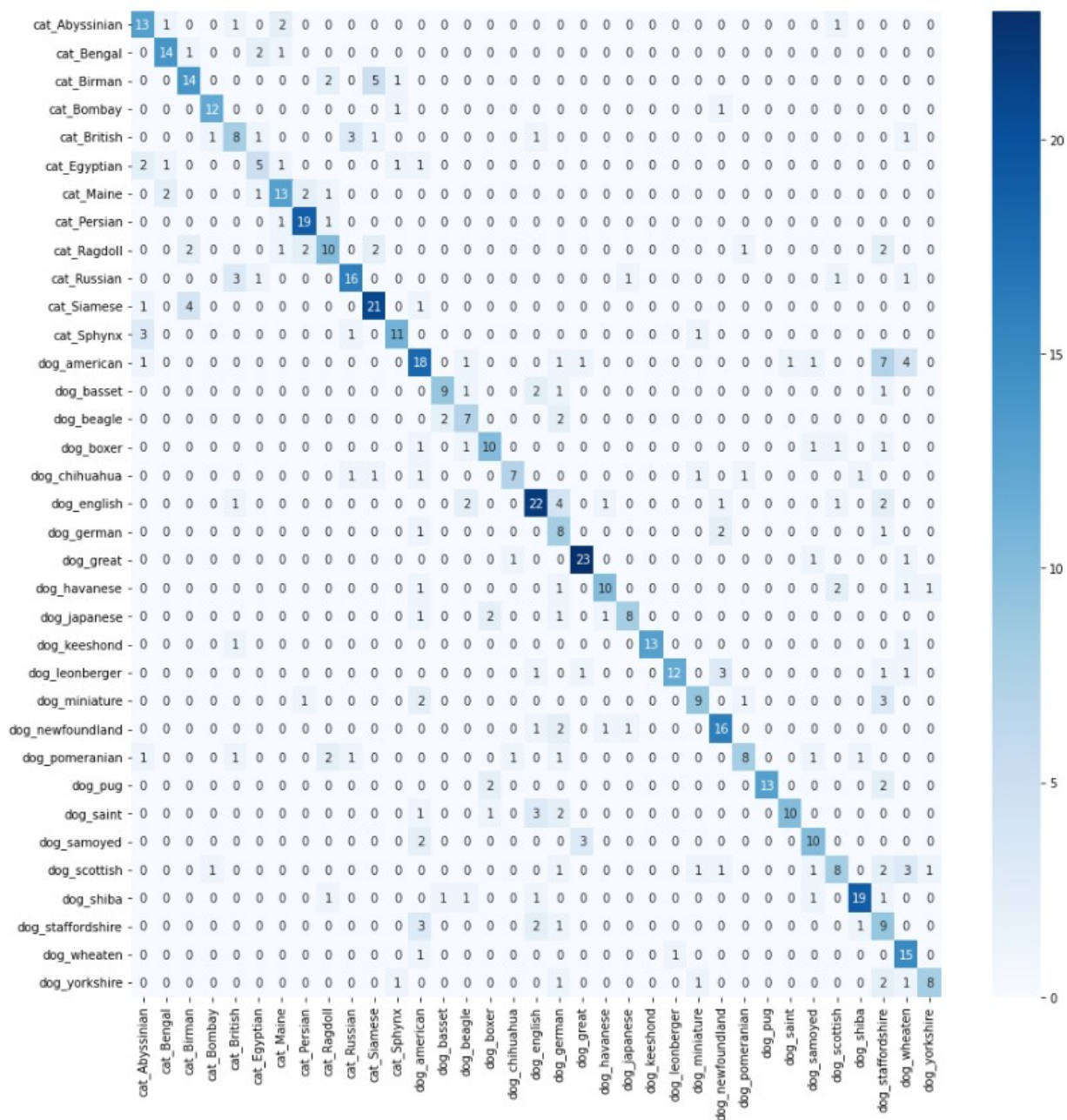
экьюраси = 0.67

точность бинарной классификации кошки vs собаки = 0.96

Графики лоссов и точностей в зависимости от эпох:



Матрица неточностей:



Модель хорошо научилась различать кошек и собак: видим точность бинарной классификации 0.96, а также видим, что левая нижняя и правая верхняя части матрицы неточностей почти нулевые (где пересекаются кошки и собаки).

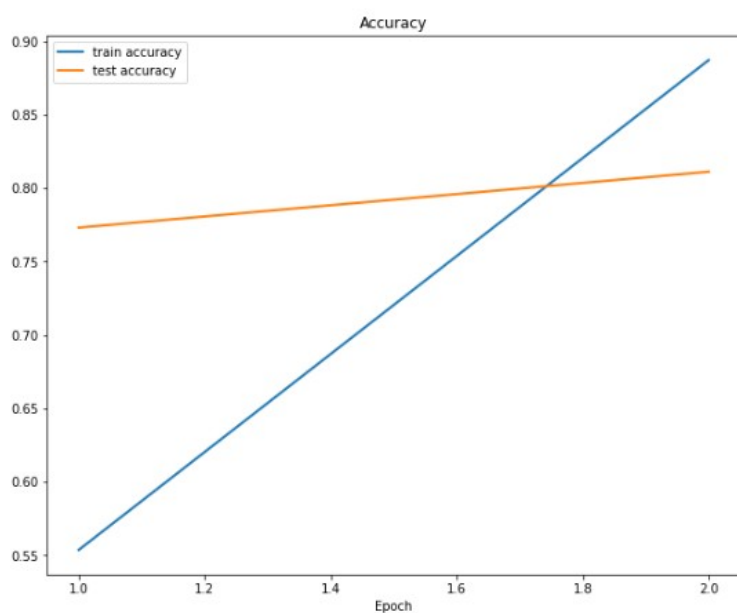
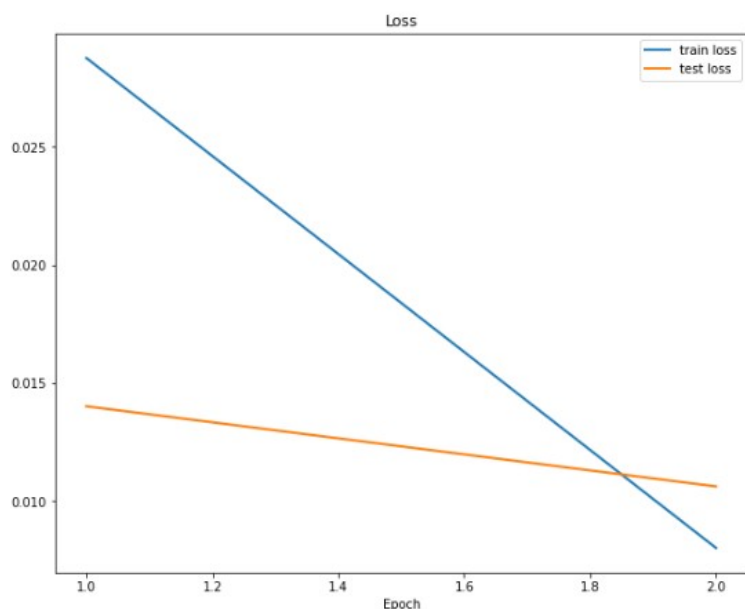
## Часть 2

Во 2 части мне нужно было использовать предобученные модели для классификации собак и кошек по породам на торче.

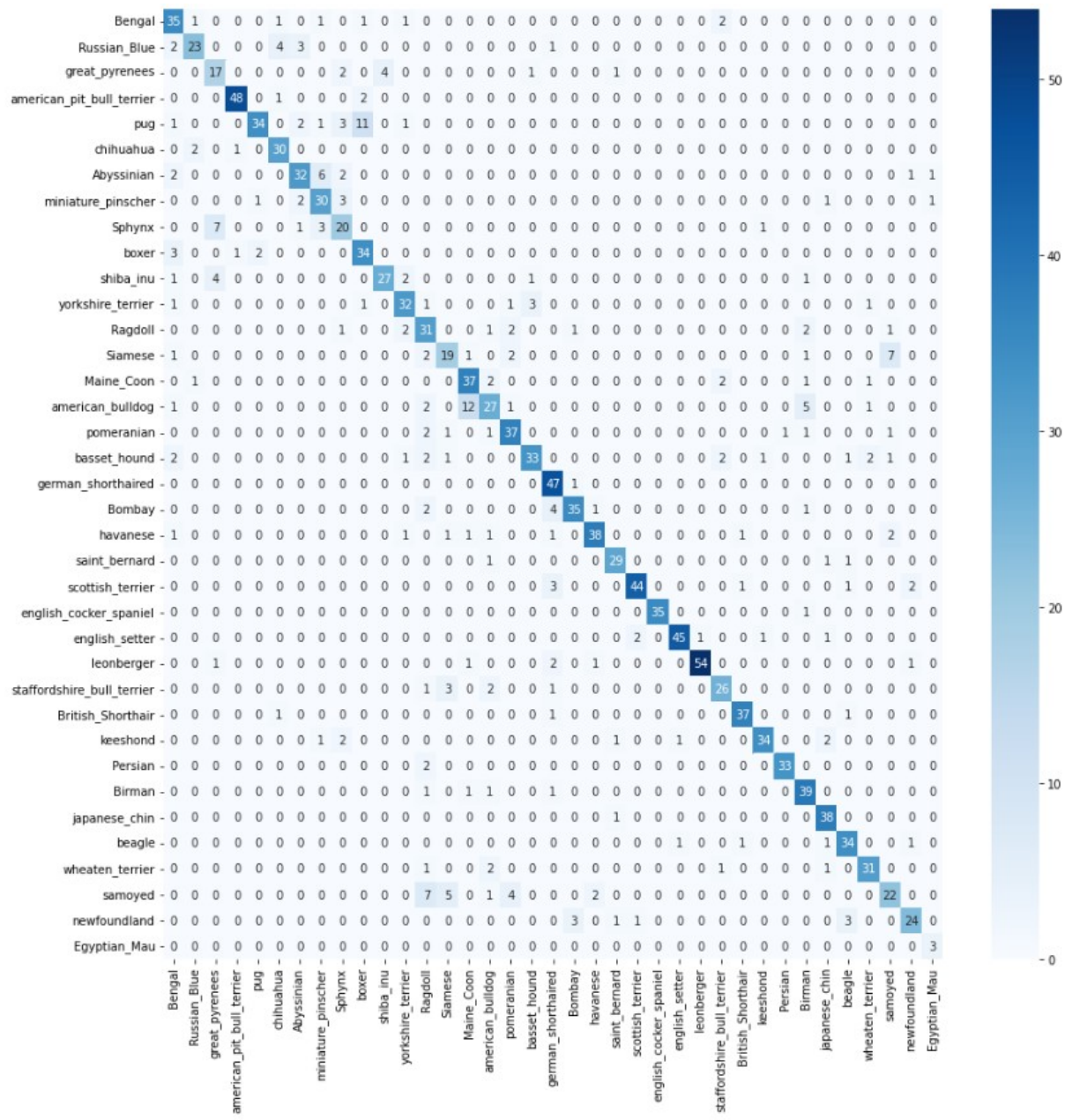
Я попробовал две сети: VGG-16 и ResNet-18. Обе сети были предобучены на ImageNet. В каждой сети я заменил последний полносвязный слой, чтобы на выходе у меня было не 1000 классов, а столько, сколько в моем датасете.

Так как модели предобучены, то они уже неплохо справляются с выделением фичей из картинок. Это значит, что не нужно много эпох на дообучение. Я решил сделать 2 эпохи обучения. Обе сети показали себя примерно одинаково - получили эккюраси 0.81-0.82.

Графики лосса и эккюраси от эпох:



Матрица неточностей:



## Вывод

В этой работе я попрактиковался в обучении сверточных нейросетей для классификации изображений. Я попробовал два подхода: обучение сети с нуля и трансфер лернинг. Второй подход, конечно же, оказался более точным из-за инициализации модели весами, полученных после обучения на ImageNet.

Также я познакомился с фреймворком Тензорфлоу, на котором мне до этого не приходилось работать. Оказалось, что это очень удобный инструмент, где большая часть работы уже сделана за нас. Но этот фреймворк не очень гибкий по сравнению с торчом. Поэтому я бы использовал тензорфлоу для решения простых задач.