Содержание

1 Компоненты свёрточной нейрнной сети

Определение 1. Изображение — тензор $M \in \mathbb{R}^{m \times n \times d}$, m — ширина изображения, n — длина. Чаще всего $d = 3$ (3 канала — Red, Green, Blue).

Учитывая, что обычно рассматривают трёхканальные или четырёхканальные изображения, можно под словом "тензор"понимать трёхмерный или четырёхмерный массив.

Пусть $X \in \mathbb{R}^{m \times n \times d}$ — случайная величина "изображение".

Рассматриваются следующие задачи для изображений:

- Классификация $f : X \to \{1, \dots, K\}$, K — число классов
- Сегментация (1 объект) $f : X \to Y$, $Y \in [0,1]^{m \times n}$, $Y_{ij} = \mathsf{P}(X_{ij} \in \text{segment})$, где segment — множество точек изображения (пикселей), принадлежащих объекту. Понятно, что можно обобщить на сегментацию нескольких классов объектов.

1.1 Свёртка

Определение 2. Пусть $f, g : \mathbb{R}^n \to \mathbb{R}$, интегрируемые на $\mathbb{R}^n$. *Свёрткой* будем называть функцию $f*g : \mathbb{R}^n \to \mathbb{R}$, равную $(f*g)(x) = \int\limits_{\mathbb{R}^n} f(y)g(x-y)dy$

Далее под свёрткой будем понимать следующее определение.

Определение 3. Рассмотрим $M \in \mathbb{R}^{m \times n}$ — изображение по одному из каналов, $K \in \mathbb{R}^{k \times l}$ — ядро, *свёрткой изображения относительно отображения* $h : \mathbb{R}^{m \times n} \times \mathbb{R}^{k \times l} \to \mathbb{R}$ *называется функция* $*$, рассмотрим, $(M*K)(i,j) = \sum\limits_{a=1}^{k} \sum\limits_{b=1}^{l} M_{i+a,j+b} K_{ab}$. $(M * K) \in \mathbb{R}^{(m-k+1) \times (n-l+1)}$

Для того чтобы размер изображения после свёртки был равен исходному изображению можно как-нибудь заполнить края изображения (padding). Идея аналогична скользящему среднему с синхронизацией в центральной точке. В основном используются следующие способы padding:

- zeros — заполнение нулями.
- reflect — "отражение например, вектор $(1,2,3)$ при padding размера 2 отобразится в вектор $(3,2,1,2,3,2,1)$.
- replicate — заполнение крайними значениями.
- circular — круговое заполнение, вектор $(1,2,3)$ преобразуется в $(2,3,1,2,3,1,2)$.

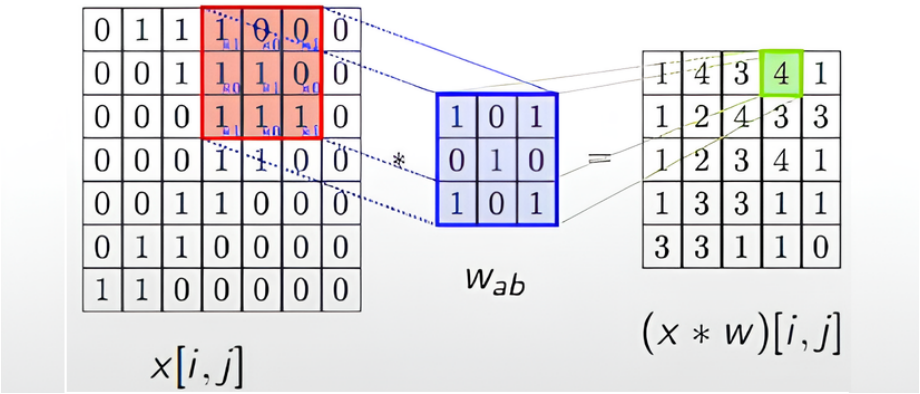

Рис. 1: Операция свёртки

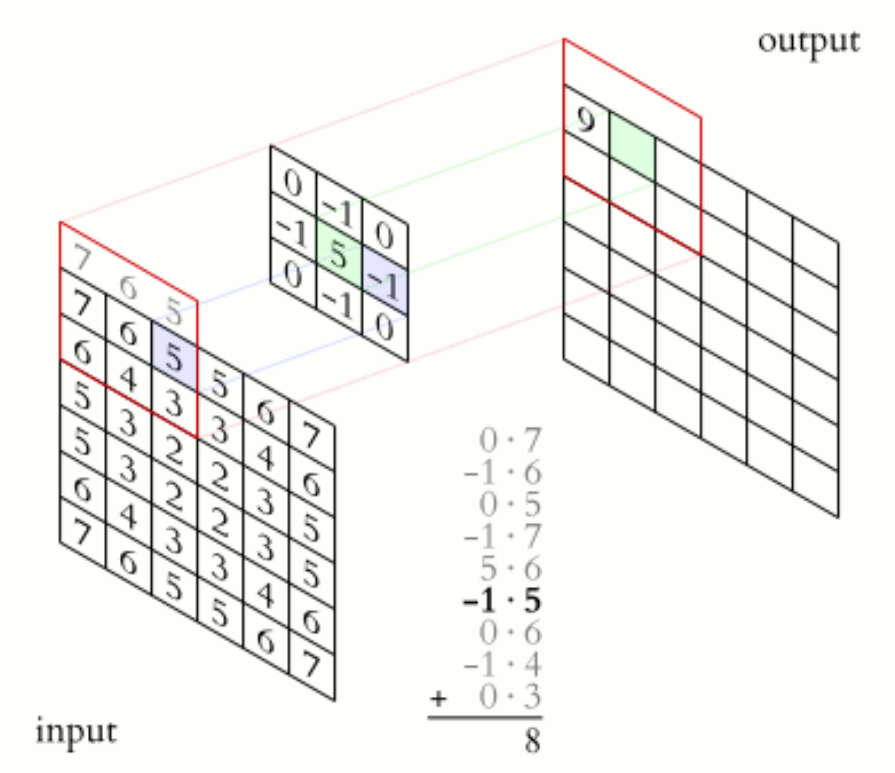

Рис. 2: Операция свёртки

1.2 Свёрточный слой

Свёрточный слой задаётся следующими параметрами:

- Размер ядер $k \times l$, $k \leqslant m$, $l \leqslant n$, $r \in \mathbb{N}$ число ядер
- Способ заполнения краёв (padding)
- Размер заполнения краёв $P \in \mathbb{N}_0$
- Величина сдвига ядра (stride) $S \in R^2$. Например, для $S = (1, 1)$ получаем операцию, похожую на операцию вложения в 2D-SSA.

1.3 Pooling слой

Свёрточная нейросеть помимо свёрточных слоёв состоит из pooling слоёв

Определение 4. Рассмотрим одноканальное изображение (матрицу) M размера $m \times n$. Выберем p и q, кратные m и n соответственно. Разобьём матрицу на дизъюнктные подматрицы размера $k \times l$ $P_{ij} = \{M_{(i-1)p+k,(j-1)q+l}\}_{k=1,l=1}^{p,q}$, $i \in 1 : \frac{m}{p}$, $j \in 1 : \frac{n}{q}$. *Операция pooling* применяет к каждой матрице P_{ij} некоторую функцию f, в результате получается матрица F, состоящая из элементов $F(i, j) = f(P_{ij})$.

Чаще всего используют max pooling, average pooling и sum pooling. Pooling слой применяется для уменьшения размерности изображения. В случае от-

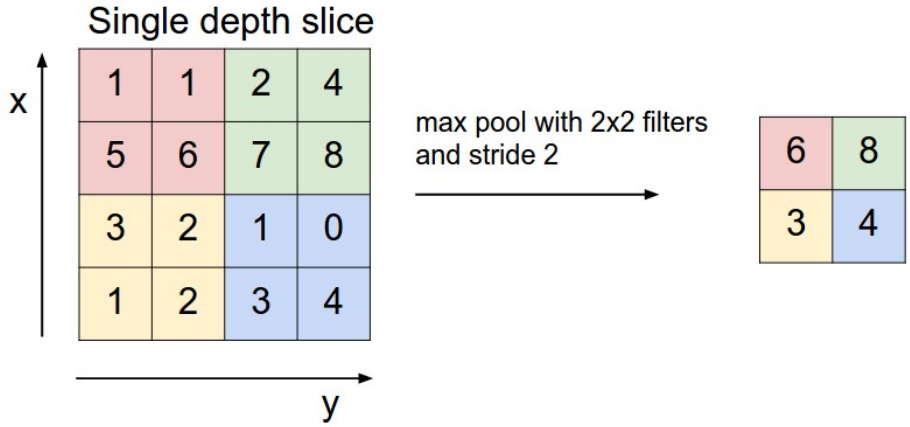

Рис. 3: Пример max-pooling

сутствия max-pooling слоёв применим алгоритм обратного распространения ошибки. Если присутствуют max-pooling слои, то градиенты пробрасываются в ту клетку, на которой достигается максимум, остальные градиенты равны нулю.

В рис. 1.3 ненулевые градиенты будут в точках $(2, 2), (2, 4), (3, 1), (4, 4)$.

На практике не встречается использование min-pooling потому что в случае его использования может произойти затухание градиентов.
В контексте операции подвыборки (pooling), когда говорится о "наличии признака в данных областях обычно имеется в виду следующее:

MaxPooling:

Если внутри области (например, 2x2 или 3x3) есть какой-либо ярко выраженный признак (например, край объекта, текстура, угол), то максимальное значение в этой области будет высоким. MaxPooling выбирает это максимальное значение, предполагая, что оно содержит наиболее важную информацию о наличии этого признака. AveragePooling:

Если внутри области есть признаки, но они распределены более равномерно, и нет какого-то одного доминирующего признака, то усредненное значение будет относительно низким. **AveragePooling** усредняет значения, предполагая, что более равномерное распределение признаков может содержать более общую информацию. В обоих случаях мы говорим о "признаках"в контексте содержания информации о структуре изображения. Под "признаками"могут пониматься различные аспекты изображения, такие как контуры, текстуры, цвета и т. д. Максимальное или усредненное значение в подвыборке служит метрикой того, насколько выражен или равномерен этот "признак"в данной области.

2 Свёрточная нейронная сеть

В зависимости от постановки задачи после применения свёрточных и pooling слоёв и применения функций активации могут следовать полносвязные или свёрточные слои. Обычно после применения свёртки следует поэлементное применение функции ReLU, равной $\mathrm{ReLU}(x) = \max(0, x)$ для того, чтобы убрать отрицательные элементы из результатов применения свёрточных слоёв.

Каждый элемент изображения M_{ijk} показывает интенсивность выбранного пикселя соответствующей координаты и соответствующего канала, в таком случае трудно трактовать отрицательные значения в изображении.

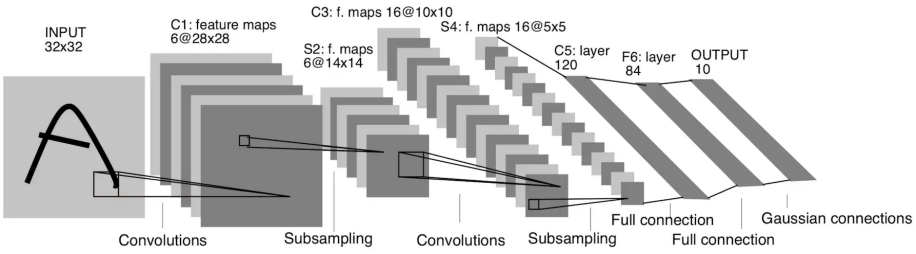

Рис. 4: Архитектура сети LeNet

Рассмотрим пример сети LeNet. На первом шаге применяется 6 свёрток размером 5×5, затем max-pooling размером 2×2, потом 16 свёрток 5×5 и max-pooling 2×2, после этого 16 полученных матриц размером 5×5 вытягиваются в один вектор, после которого следует два полносвязанных слоя, и на последнем шаге применяется функция активации softmax. В результате получаем вероятность принадлежности одному из десяти классов для каждого класса.

3 Dropout в нейронных сетях

Dropout — это техника регуляризации, применяемая в нейронных сетях для предотвращения переобучения. Она основана на идее случайного "выключения"(отключения) некоторых нейронов во время обучения.

3.1 Идея Dropout

В процессе обучения каждый нейрон в слое с вероятностью p может быть временно отключен (выходить из игры) на текущей итерации обучения. Эта вероятность p называется параметром Dropout. Таким образом, в каждой эпохе обучения структура сети меняется, что помогает предотвратить сильное переобучение.

3.2 Математическое представление

Представим, что входной вектор для слоя l — это $\mathbf{x}^{(l)}$, а маска Dropout для этого слоя — это $\mathbf{m}^{(l)}$, где каждый элемент маски является случайной переменной, принимающей значения 0 или 1 с вероятностью p. Тогда выход слоя с применением Dropout можно записать следующим образом:

$$\mathbf{y}^{(l)} = \mathbf{m}^{(l)} \odot \mathbf{x}^{(l)}$$

где $\odot$ — поэлементное умножение (применение маски).

3.3 Обучение с Dropout

Во время обучения происходит вычисление градиентов и обновление весов, как обычно. В процессе прямого прохода Dropout применяется, а в процессе обратного прохода градиенты распространяются только через активные (не выключенные) нейроны.

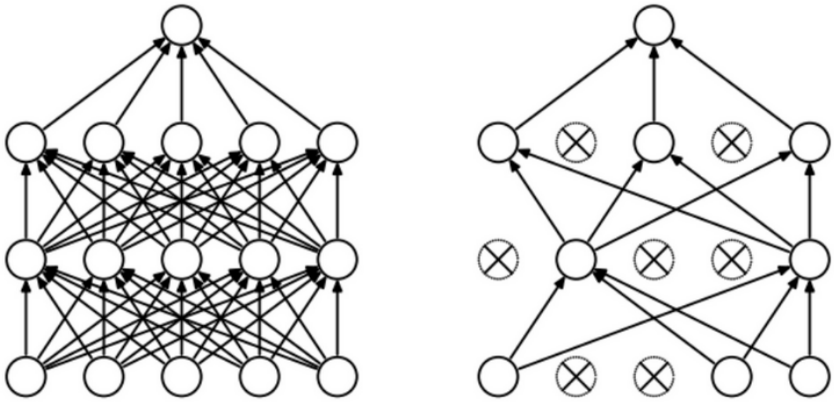

Рис. 5: dropout

4 Улчшения качества свёрточных нейросетей

4.1 Аугументация

Одной из проблем работы с изображениями является малое количество размеченных изображений. Обычно изображения размечаются вручную. Для размножения данных используются аугументации.

Определение 5. Аугументация (Augumentation) — увеличение объёма тренировочной выборки с помощью различных афинных преобразований изображений: зеркальное отражение, поворот, сдвиг, изменение масштаба.

Также аугументации позволяют улучшить качество предсказания на тестовой выборки. Например, наша задача — определение дорожных знаков по фотографии. Обучающая выборка состоит из дорожных знаков из справочника (сфотографированы "анфас"). Если обучить модель на выборке без аугументаций, то при рассмотрении реальной фотографии дорожных знаков модель может давать плохие результаты потому что как правило встречаются фотографии с некоторыми дефектами в сравнении с изображениями из обучающей выборки: повороты, размытия, сдвиги. Если рассмотреть аугументированную выборке, то в ней будет больше изображений, похожих на реальные и новая модель сможет предсказывать с большей точностью чем модель, обученная на выборке без аугументаций.

Рис. 6: Пример аугументации изображений

5 Применение CNN в обработке аудиофайлов

5.1 Спектрограммы и свертка

CNN могут эффективно использоваться для анализа аудиосигналов в виде спектрограмм. Спектрограмма представляет собой изображение, отражающее изменения частоты во времени. Сверточные слои в сети могут обнаруживать различные звуковые паттерны и особенности в разных участках времени.

5.2 Обработка последовательности звуковых данных

Сверточные нейронные сети могут быть применены для обработки временных последовательностей звуковых данных. В этом случае, сверточные слои могут извлекать локальные признаки, такие как изменения громкости или наличие определенных звуковых шаблонов.

5.3 Использование пулинга для извлечения признаков

Слой пулинга может использоваться для уменьшения размерности данных, что позволяет сети сосредотачиваться на более важных аспектах звуковой

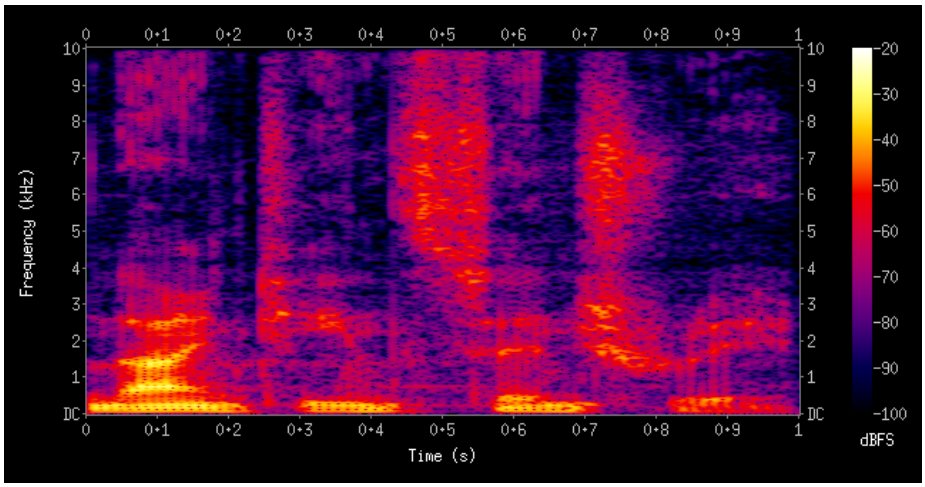

Рис. 7: Спектрограмма

информации. Например, максимальный пулинг может выделять наиболее значимые частотные компоненты в каждом временном интервале.

5.4 Задачи обработки аудиосигналов с использованием CNN

CNN могут быть применены для решения различных задач обработки аудиосигналов, таких как:

- Распознавание речи.
- Классификация звуковых событий.
- Идентификация музыкальных инструментов.
- Разделение звуковых дорожек.

6 Использование CNN в NLP

Последовательные слова в тексте представляются векторами с помощью векторных представлений(fastText, word2vec, One-hot encoding и тд.)

- **Классификация текста**

  CNN могут быть успешно применены для задачи бинарной или многоклассовой классификации текста. Сверточные слои выступают в роли фильтров, выделяя локальные паттерны в предложениях или текстах.

- **Анализ тональности**

  В решении задачи определения тональности текста, CNN извлекают контекстуальные признаки из последовательности слов, позволяя выявлять позитивные, негативные или нейтральные настроения.

- **Извлечение признаков из текста**

  Сверточные слои могут быть использованы для извлечения важных признаков из текстовых данных. Это полезно для создания эмбеддингов, представляющих слова или фразы в векторной форме.

- **Детекция именованных сущностей (NER)**

  Для задачи NER, CNN применяются для обнаружения и классификации именованных сущностей в тексте.

- **Анализ последовательности**

  В сочетании с рекуррентными слоями, такими как LSTM или GRU, CNN могут быть использованы для анализа последовательности слов, учитывая контекст в тексте.

- **Создание архитектур для задачи NLP** CNN могут составлять часть более сложных архитектур, например, TextCNN, которые включают несколько слоев свертки, слоев пулинга и полносвязанных слоев для решения конкретных задач NLP.

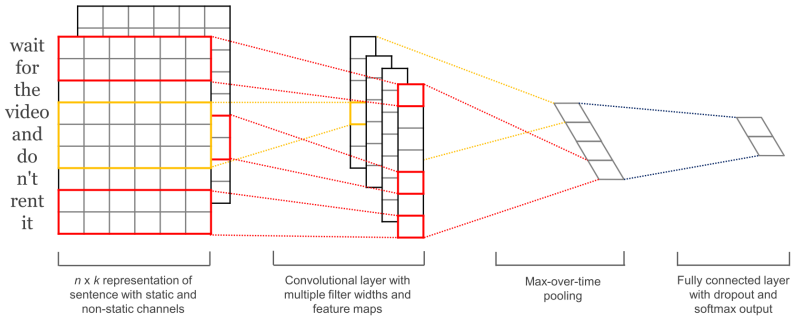

Рис. 8: Разложение слов на векторы и их дальнейшее преобразования

7 ImageNet

ImageNet — это крупнейшая база данных изображений, используемая в задачах компьютерного зрения и обучения глубоких нейронных сетей. Она играет ключевую роль в развитии и оценке различных архитектур нейронных сетей.

7.1 Описание

- **Размер базы данных:** ImageNet содержит более 14 миллионов изображений, представляющих более 20 000 категорий объектов.
- **Категории:** Каждое изображение в ImageNet привязано к одной из тысяч категорий, включающих широкий спектр объектов, таких как животные, растения, транспортные средства, еда и многое другое.
- **Задачи:** ImageNet используется для нескольких задач, включая классификацию изображений (присвоение каждому изображению одной из категорий), детекцию объектов (обнаружение и классификация объектов на изображении) и локализацию объектов (определение местоположения объектов на изображении).
- **Использование в соревнованиях:** ImageNet служит основой для проведения ежегодного соревнования ILSVRC (ImageNet Large Scale Visual Recognition Challenge), в котором исследователи и инженеры создают и сравнивают различные модели глубокого обучения на задачах классификации изображений.

7.2 Важность для глубокого обучения

ImageNet стал своего рода эталоном для оценки производительности и обобщающей способности моделей глубокого обучения. Разработка и сравнение моделей на базе данных ImageNet позволяет измерять и сравнивать способность моделей распознавать и классифицировать разнообразные объекты в реальном мире.

8 Выводы

Проблемы:

- Необходимость разметки данных для обучения
- Большое количество параметром, следовательно долгое обучение, даже на GPU

Решения:

- Использование размеченных библиотек изображений: ImageNet (14M изображений, 1000 категорий), OpenImages (9M изображений, 60K меток, 20K категория)
- Использование модели, обученной на размеченной библиотеке изображений известной архитектуры, например Alexnet, vgg net, Resnet.

Обычно новые архитектуры публикуются в результате достижения рекордной точности на каком-нибудь соревновании. Например, архитектура vgg 16 показала точность 0.927 на датасете ImageNet на соревновании ImageNet Large Scale Visual Recognition Challenge 2014 и после была опубликована.

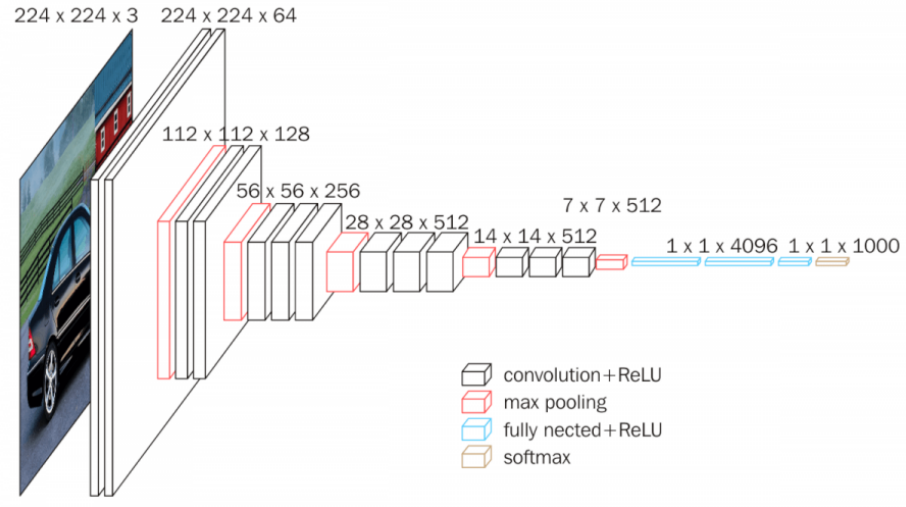

Рис. 9: Архитектура сети vgg 16

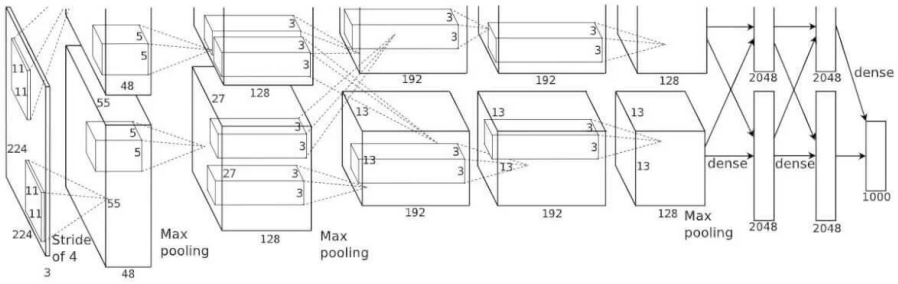

Рис. 10: Архитектура сети AlexNet

Все эти сети имеют разные(или почти разные конфигурации), но могут быть загружены и использованы для своих нужд. В таком случае, обычно вся архитектура сети загружается со своими полученными весами. Необходимо добавить последний полносвязнный слой, и обучить на ваших данных