

Введение в искусственный интеллект. Современное компьютерное зрение

Тема семинара: Несверточные слои

Бабин Д.Н., Иванов И.Е.

кафедра Математической Теории Интеллектуальных Систем

28 февраля 2023 г.



- 1 Сведение к свертке
- 2 О сигмоиде
- 3 Сверточные механизмы внимания

- Предположим, что мы используем пакет размера $T = 1$ (здесь и далее опустим этот индекс)

- Предположим, что мы используем пакет размера $T = 1$ (здесь и далее опустим этот индекс)
- Y_{ij}^k — трехмерный тензор значений для некоторого слоя, где

- Предположим, что мы используем пакет размера $T = 1$ (здесь и далее опустим этот индекс)
- Y_{ij}^k — трехмерный тензор значений для некоторого слоя, где
 - $1 \leq i \leq H, 1 \leq j \leq W$ — пространственные координаты (ширина и высота),
 - $k = 1 \dots K$ — номер карты признаков.

- Предположим, что мы используем пакет размера $T = 1$ (здесь и далее опустим этот индекс)
- Y_{ij}^k — трехмерный тензор значений для некоторого слоя, где
 - $1 \leq i \leq H, 1 \leq j \leq W$ — пространственные координаты (ширина и высота),
 - $k = 1 \dots K$ — номер карты признаков.
- Выход нормализованного слоя: $Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$



Пакетная нормализация как линейная операция от входа

- $$Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$

Пакетная нормализация как линейная операция от входа

- $$Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$
- Перепишем формулу в другом виде:

Пакетная нормализация как линейная операция от входа

- $Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$
- Перепишем формулу в другом виде:

$$Z_{ij}^k = Y_{ij}^k \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$



Пакетная нормализация как линейная операция от входа

- $Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$

- Перепишем формулу в другом виде:

$$Z_{ij}^k = Y_{ij}^k \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$

- Т.о., получаем $Z_{ij}^k = G^k Y_{ij}^k + g^k$, где



Пакетная нормализация как линейная операция от входа

- $Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$

- Перепишем формулу в другом виде:

$$Z_{ij}^k = Y_{ij}^k \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$

- Т.о., получаем $Z_{ij}^k = G^k Y_{ij}^k + g^k$, где

- Мультипликативный член $G^k = \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$,



Пакетная нормализация как линейная операция от входа

- $Z_{ij}^k = \gamma^k \frac{Y_{ij}^k - \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$

- Перепишем формулу в другом виде:

$$Z_{ij}^k = Y_{ij}^k \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}} + \beta^k$$

- Т.о., получаем $Z_{ij}^k = G^k Y_{ij}^k + g^k$, где

- Мультипликативный член $G^k = \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$,
- Аддитивный член $g^k = \beta^k - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$.



- $Z_{ij}^k = G^k Y_{ij}^k + g^k$

Пакетная нормализация как свертка

- $Z_{ij}^k = G^k Y_{ij}^k + g^k$
- Значит, пакетная нормализация — это поканальная (depthwise, см. предыдущую лекцию) свертка с ядром размера 1×1 !



Пакетная нормализация как свертка

- $Z_{ij}^k = G^k Y_{ij}^k + g^k$
- Значит, пакетная нормализация — это поканальная (depthwise, см. предыдущую лекцию) свертка с ядром размера 1×1 !
- А композиция сверток — тоже свертка (*Упражнение: доказать*)



Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация

Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация
- По слоям:

Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация
- По слоям:

$$X_{ij}^m \xrightarrow{F_{uv}^{mk}, b^k} Y_{ij}^k \xrightarrow{G^k, g^k} Z_{ij}^k$$

Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация
- По слоям:

$$X_{ij}^m \xrightarrow{F_{uv}^{mk}, b^k} Y_{ij}^k \xrightarrow{G^k, g^k} Z_{ij}^k$$

- Выписываем еще раз формулы для свертки:

Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация
- По слоям:

$$X_{ij}^m \xrightarrow{F_{uv}^{mk}, b^k} Y_{ij}^k \xrightarrow{G^k, g^k} Z_{ij}^k$$

- Выписываем еще раз формулы для свертки:

$$Y_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1, j+v-1}^m \cdot F_{uv}^{mk} + b^k, \quad \forall k = 1 \dots K$$

и для пакетной нормализации:

Объединение свертки и пакетной нормализации (1)

- Обычно: сначала свертка, потом пакетная нормализация
- По слоям:

$$X_{ij}^m \xrightarrow{F_{uv}^{mk}, b^k} Y_{ij}^k \xrightarrow{G^k, g^k} Z_{ij}^k$$

- Выписываем еще раз формулы для свертки:

$$Y_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1, j+v-1}^m \cdot F_{uv}^{mk} + b^k, \quad \forall k = 1 \dots K$$

и для пакетной нормализации:

$$Z_{ij}^k = G^k Y_{ij}^k + g^k$$



Объединение свертки и пакетной нормализации (2)

- Объединяя, получим:

Объединение свертки и пакетной нормализации (2)

- Объединяя, получим:

$$Z_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1,j+v-1}^m \cdot F_{uv}^{mk} \cdot G^k + b^k + g^k, \quad \forall k = 1 \dots K$$

- Т.о., мы получили свертку с параметрами O_{uv}^{mk}, o^k , где (подтягиваем параметры пакетной нормализации):

Объединение свертки и пакетной нормализации (2)

- Объединяя, получим:

$$Z_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1,j+v-1}^m \cdot F_{uv}^{mk} \cdot G^k + b^k + g^k, \quad \forall k = 1 \dots K$$

- Т.о., мы получили свертку с параметрами O_{uv}^{mk}, o^k , где (подтягиваем параметры пакетной нормализации):
 - Ядро $O_{uv}^{mk} = F_{uv}^{mk} \cdot \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$,



Объединение свертки и пакетной нормализации (2)

- Объединяя, получим:

$$Z_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1,j+v-1}^m \cdot F_{uv}^{mk} \cdot G^k + b^k + g^k, \quad \forall k = 1 \dots K$$

- Т.о., мы получили свертку с параметрами O_{uv}^{mk}, o^k , где (подтягиваем параметры пакетной нормализации):
 - Ядро $O_{uv}^{mk} = F_{uv}^{mk} \cdot \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$,
 - Аддитивный член $o^k = b^k + \beta^k - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$.



Объединение свертки и пакетной нормализации (2)

- Объединяя, получим:

$$Z_{ij}^k = \sum_{m=1}^M \sum_{u,v=1}^{p,q} X_{i+u-1,j+v-1}^m \cdot F_{uv}^{mk} \cdot G^k + b^k + g^k, \quad \forall k = 1 \dots K$$

- Т.о., мы получили свертку с параметрами O_{uv}^{mk}, o^k , где (подтягиваем параметры пакетной нормализации):
 - Ядро $O_{uv}^{mk} = F_{uv}^{mk} \cdot \frac{\gamma^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$,
 - Аддитивный член $o^k = b^k + \beta^k - \frac{\gamma^k \mu_{avg}^k}{\sqrt{\sigma_{avg}^{2k} + \epsilon}}$.
- Из $X_{ij}^m \xrightarrow{F_{uv}^{mk}, b^k} Y_{ij}^k \xrightarrow{G^k, g^k} Z_{ij}^k$ получили $X_{ij}^m \xrightarrow{O_{uv}^{mk}, o^k} Z_{ij}^k$.



- **Вопрос:** Можно ли maxpooling представить как свертку?



- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)

- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)
- **Вопрос:** Можно ли average pooling представить как свертку?

- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)
- **Вопрос:** Можно ли average pooling представить как свертку?
- **Ответ:** Да, и рассмотрим на примере global average pooling (GAP):



- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)
- **Вопрос:** Можно ли average pooling представить как свертку?
- **Ответ:** Да, и рассмотрим на примере global average pooling (GAP):
 - Пусть двумерный (не обращаем внимание на карты) вход $X_{ij}, 1 \leq i \leq H, 1 \leq j \leq W$,



- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)
- **Вопрос:** Можно ли average pooling представить как свертку?
- **Ответ:** Да, и рассмотрим на примере global average pooling (GAP):
 - Пусть двухмерный (не обращаем внимание на карты) вход $X_{ij}, 1 \leq i \leq H, 1 \leq j \leq W$,
 - $GAP2D(X) = \frac{1}{HW} \sum_{i,j=1}^{H,W} X_{ij}$,



- **Вопрос:** Можно ли maxpooling представить как свертку?
- **Ответ:** Нет, так как операция взятия максимума — нелинейная (в то время как свертка всегда линейна)
- **Вопрос:** Можно ли average pooling представить как свертку?
- **Ответ:** Да, и рассмотрим на примере global average pooling (GAP):
 - Пусть двухмерный (не обращаем внимание на карты) вход $X_{ij}, 1 \leq i \leq H, 1 \leq j \leq W$,
 - $GAP2D(X) = \frac{1}{HW} \sum_{i,j=1}^{H,W} X_{ij}$,
 - Тогда свертка, соответствующая $GAP2D(X)$ — это свертка с ядром $F_{GAP} = \frac{1}{HW} \mathbb{1}_{i,j=1}^{H,W}$ без аддитивного члена, с размером, как у входа $H \times W$, применяемая без добавки (падинга) и в режиме “VALID”



- Вспомним три основных вида активации:

- Вспомним три основных вида активации:

- 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,

- Вспомним три основных вида активации:

① Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,

② Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,

- Вспомним три основных вида активации:

- 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
- 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
- 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.

- Вспомним три основных вида активации:
 - 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
 - 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
 - 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.
- Изначально все использовали $\sigma(x)$. Тем не менее, сейчас он почти не встречается. Почему?



- Вспомним три основных вида активации:
 - 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
 - 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
 - 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.
- Изначально все использовали $\sigma(x)$. Тем не менее, сейчас он почти не встречается. Почему?
- **Проблема:** выход $\sigma(x)$ — не центрирован в нуле.



- Вспомним три основных вида активации:
 - 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
 - 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
 - 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.
- Изначально все использовали $\sigma(x)$. Тем не менее, сейчас он почти не встречается. Почему?
- **Проблема:** выход $\sigma(x)$ — не центрирован в нуле.
- **Решение:** использовать $\tanh(x)$.



- Вспомним три основных вида активации:
 - 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
 - 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
 - 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.
- Изначально все использовали $\sigma(x)$. Тем не менее, сейчас он почти не встречается. Почему?
- **Проблема:** выход $\sigma(x)$ — не центрирован в нуле.
- **Решение:** использовать $\tanh(x)$.
- Однако это не избавляет от главной проблемы — **исчезающих градиентов**:
 - 1 Производная $\sigma'(x) = \sigma(x)(1 - \sigma(x))$,



- Вспомним три основных вида активации:
 - 1 Сигмоида $\sigma(x) = \frac{1}{1+\exp(-x)}$,
 - 2 Гиперболический тангенс $\tanh(x) = 2\sigma(2x) - 1$,
 - 3 Rectified Linear Unit $ReLU(x) = \max(0, x)$.
- Изначально все использовали $\sigma(x)$. Тем не менее, сейчас он почти не встречается. Почему?
- **Проблема:** выход $\sigma(x)$ — не центрирован в нуле.
- **Решение:** использовать $\tanh(x)$.
- Однако это не избавляет от главной проблемы — **исчезающих градиентов**:
 - 1 Производная $\sigma'(x) = \sigma(x)(1 - \sigma(x))$,
 - 2 Для любых больших по модулю x $\sigma(x)$ стремится к 1 или 0, и соответственно его производная — всегда к нулю.



- $ReLU(x) = \max(0, x)$ дает нулевую производную только при отрицательных x ,

¹<https://stats.stackexchange.com/a/422579>

О ReLU¹

- $ReLU(x) = \max(0, x)$ дает нулевую производную только при отрицательных x ,
- $ReLU(x)$ при $x > 0$ дает константную производную (равную 1),

¹<https://stats.stackexchange.com/a/422579>

О ReLU¹

- $ReLU(x) = \max(0, x)$ дает нулевую производную только при отрицательных x ,
- $ReLU(x)$ при $x > 0$ дает константную производную (равную 1),
- $ReLU(x)$ потрясающе эффективен в реализации на конечном устройстве.

¹<https://stats.stackexchange.com/a/422579>

О ReLU¹

- $ReLU(x) = \max(0, x)$ дает нулевую производную только при отрицательных x ,
- $ReLU(x)$ при $x > 0$ дает константную производную (равную 1),
- $ReLU(x)$ потрясающе эффективен в реализации на конечном устройстве.
- Иллюстрация:

Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



tanh

$$\tanh(x)$$



ReLU

$$\max(0, x)$$



¹<https://stats.stackexchange.com/a/422579>

О ReLU¹

- $ReLU(x) = \max(0, x)$ дает нулевую производную только при отрицательных x ,
- $ReLU(x)$ при $x > 0$ дает константную производную (равную 1),
- $ReLU(x)$ потрясающе эффективен в реализации на конечном устройстве.
- Иллюстрация:

Sigmoid

$$\sigma(x) = \frac{1}{1+e^{-x}}$$



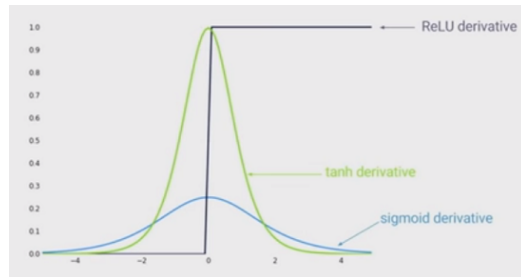
tanh

$$\tanh(x)$$



ReLU

$$\max(0, x)$$



¹<https://stats.stackexchange.com/a/422579>

Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)
- Важность карты агрегируется через MaxPool / AvgPool по пространственным размерностям: $F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : C \times 1 \times 1$

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)
- Важность карты агрегируется через MaxPool / AvgPool по пространственным размерностям: $F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : C \times 1 \times 1$
- После чего применяется двухслойный перцептрон, при этом для уменьшения количества параметров применяется сжимающе-разжимающее отображение с параметром r : $Y_1 = W_1 RELU(W_0 F_{agg}), W_0 : C/r \times C, W_1 : C \times C/r$

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)
- Важность карты агрегируется через MaxPool / AvgPool по пространственным размерностям: $F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : C \times 1 \times 1$
- После чего применяется двухслойный перцептрон, при этом для уменьшения количества параметров применяется сжимающе-разжимающее отображение с параметром r : $Y_1 = W_1 RELU(W_0 F_{agg}), W_0 : C/r \times C, W_1 : C \times C/r$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

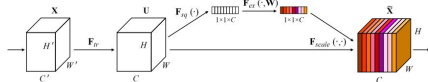
Сверточные механизмы внимания: поканальный (1)

- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)
- Важность карты агрегируется через MaxPool / AvgPool по пространственным размерностям: $F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : C \times 1 \times 1$
- После чего применяется двухслойный перцептрон, при этом для уменьшения количества параметров применяется сжимающе-разжимающее отображение с параметром r : $Y_1 = W_1 RELU(W_0 F_{agg}), W_0 : C/r \times C, W_1 : C \times C/r$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$
- В конце исходный тензор $X : C \times H \times W$ в каждой пространственной размерности поэлементно перемножается на тензор внимания $Y : C \times 1 \times 1$

²Hu, Jie, Li Shen, and Gang Sun. "Squeeze-and-excitation networks." 2017

Сверточные механизмы внимания: поканальный (1)

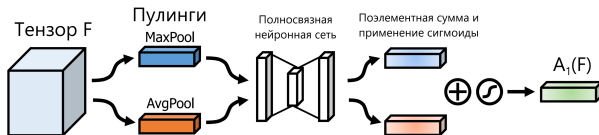
- Вводится механизм внимания на конкретные карты
- *Внимание* — это обычно мультипликативный коэффициент $a \in [0, 1]$ (вес)
- Важность карты агрегируется через MaxPool / AvgPool по пространственным размерностям: $F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : C \times 1 \times 1$
- После чего применяется двухслойный перцептрон, при этом для уменьшения количества параметров применяется сжимающе-разжимающее отображение с параметром r : $Y_1 = W_1 RELU(W_0 F_{agg}), W_0 : C/r \times C, W_1 : C \times C/r$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$
- В конце исходный тензор $X : C \times H \times W$ в каждой пространственной размерности поэлементно перемножается на тензор внимания $Y : C \times 1 \times 1$
- Популярность этот вид внимания приобрел под названием “Squeeze-and-Excitation”²



²Hu, Jie, Li Shen, and Gang Sun. “Squeeze-and-excitation networks.” 2017

Сверточные механизмы внимания: поканальный (2)

- На иллюстрации ниже сделана комбинация MaxPool и AvgPool через сумму
- При этом двухслойный перцептрон один и тот же



Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)

³Wang, Fei, et al. "Residual attention network for image classification." 2017

Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)
- Важность позиции агрегируется так же — через MaxPool / AvgPool, но уже не по пространственным размерностям, а поканально:

$$F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : 1 \times H \times W$$

³Wang, Fei, et al. "Residual attention network for image classification." 2017

Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)
- Важность позиции агрегируется так же — через MaxPool / AvgPool, но уже не по пространственным размерностям, а поканально:
$$F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : 1 \times H \times W$$
- После чего применяется обычная двумерная свертка W размерности $k \times k$ для сглаживания: $Y_1 = W * F_{agg}$

³Wang, Fei, et al. "Residual attention network for image classification." 2017

Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)
- Важность позиции агрегируется так же — через MaxPool / AvgPool, но уже не по пространственным размерностям, а поканально:
$$F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : 1 \times H \times W$$
- После чего применяется обычная двухмерная свертка W размерности $k \times k$ для сглаживания: $Y_1 = W * F_{agg}$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$

³Wang, Fei, et al. "Residual attention network for image classification." 2017

Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)
- Важность позиции агрегируется так же — через MaxPool / AvgPool, но уже не по пространственным размерностям, а поканально:
$$F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : 1 \times H \times W$$
- После чего применяется обычная двумерная свертка W размерности $k \times k$ для сглаживания: $Y_1 = W * F_{agg}$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$
- В конце исходный тензор $X : C \times H \times W$ в каждой карте поэлементно перемножается на тензор внимания $Y : 1 \times H \times W$

³Wang, Fei, et al. "Residual attention network for image classification." 2017

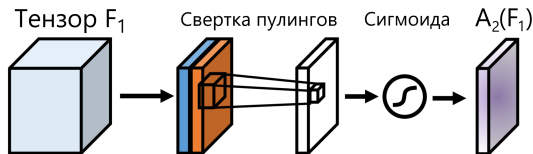
Сверточные механизмы внимания: пространственный (1)

- Вводится также механизм внимания на конкретные пространственные позиции (но уже без учета карт!)
- Важность позиции агрегируется так же — через MaxPool / AvgPool, но уже не по пространственным размерностям, а поканально:
$$F_{agg} = AGG(X), X : C \times H \times W, F_{agg} : 1 \times H \times W$$
- После чего применяется обычная двухмерная свертка W размерности $k \times k$ для сглаживания: $Y_1 = W * F_{agg}$
- Коэффициент внимания вычисляется через сигмоид: $Y = \sigma(Y_1)$
- В конце исходный тензор $X : C \times H \times W$ в каждой карте поэлементно перемножается на тензор внимания $Y : 1 \times H \times W$
- Одно из первых применений этого вид внимания прошло под названием “Residual Attention”³

³Wang, Fei, et al. “Residual attention network for image classification.” 2017

Сверточные механизмы внимания: пространственный (2)

- На иллюстрации ниже сделана комбинация MaxPool и AvgPool через конкатенацию карт
- Итоговая свертка имеет размерность уже не $1 \times k \times k$, а $2 \times k \times k$



Сверточные механизмы внимания: комбинация

- Оказывается, можно комбинировать поканальный и пространственный механизмы внимания

⁴Woo, Sanghyun, et al. "Cbam: Convolutional block attention module." 2018

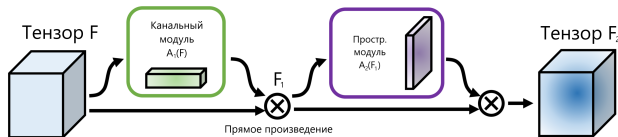
Сверточные механизмы внимания: комбинация

- Оказывается, можно комбинировать поканальный и пространственный механизмы внимания
- При этом наилучшие результаты давал подход, где сначала применяется поканальный, а затем – пространственный механизмы внимания

⁴Woo, Sanghyun, et al. "Cbam: Convolutional block attention module." 2018

Сверточные механизмы внимания: комбинация

- Оказывается, можно комбинировать поканальный и пространственный механизмы внимания
- При этом наилучшие результаты давал подход, где сначала применяется поканальный, а затем – пространственный механизмы внимания
- Наибольшую известность такой подход получил с названием “Convolutional Block Attention Module”⁴



⁴Woo, Sanghyun, et al. "Cbam: Convolutional block attention module." 2018

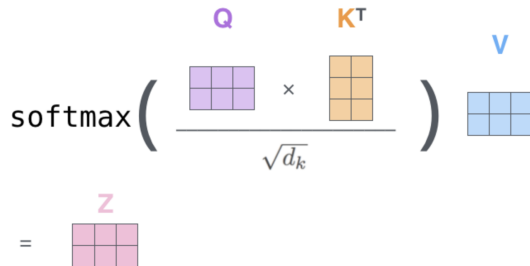
Внимание на себя (self-attention)

- Тем не менее, на данный момент наибольшей популярностью пользуется механизм внимания, предназначенный для обработки естественных языков, под названием self-attention⁵

⁵Vaswani, Ashish, et al. "Attention is all you need." 2017

Внимание на себя (self-attention)

- Тем не менее, на данный момент наибольшей популярностью пользуется механизм внимания, предназначенный для обработки естественных языков, под названием self-attention⁵
- To be coming soon...



The diagram illustrates the self-attention calculation in matrix form. It shows a purple 3x3 matrix labeled 'Q' multiplied by an orange 3x3 matrix labeled 'K^T'. The result of this multiplication is divided by the square root of the dimension $\sqrt{d_k}$. This entire operation is enclosed in a 'softmax' function. The result of the softmax is then multiplied by a blue 3x3 matrix labeled 'V'. The final result is a pink 3x3 matrix labeled 'Z'.

$$\text{softmax}\left(\frac{Q \times K^T}{\sqrt{d_k}}\right) V = Z$$

The self-attention calculation in matrix form

⁵Vaswani, Ashish, et al. "Attention is all you need." 2017



Спасибо за внимание!