LISTA TEÓRICA - SEGMENTAÇÃO SEMÂNTICA

THIAGO MALTA COUTINHO - 2014123335

① Porque segmentação semântica atribui uma classe a cada pixel e detecção de objetos atribui uma classe a um conjunto de pixels definidos pelo bounding box.

② Classificação de imagem tem o objetivo de dizer se há ou não uma classe na imagem.
Detecção de objetos determina a localização do objeto na imagem e o classifica.
Segmentação semântica classifica cada pixel da imagem
Segmentação de instância determina a localização do objeto na imagem e classifica os pixels do objeto.
OBS: Detecção de objetos determina a localização utilizando uma bounding box, mesmo todos os pixels da caixa não pertencentes ao objeto classificado.

③ Montar a rede e utilizar uma camada softmax na saída. O intervalo [0,1] será mapeado para um valor na escala de cinza [0,255]. Cada pixel recebe uma probabilidade e um valor de cinza no mapa final.

④ Efetuar recortes na imagem original e passar para a rede. O pixel central é o pixel que será classificado. Essa abordagem é custosa e alguns pixels de borda serão perdidos caso não se use padding.

⑤ Uma rede Fully connected com um neurônio para cada pixel não mantém conetamente a informação espacial dos pixels, deteriorando o desempenho da rede.

⑥ Downsampling é feito ao aplicar filtros convolucionais. O objetivo é extrair/gerar características completas que representam bem o problema e o torna separável/"classificável". Up sampling é feito com o objetivo de levar as features completas do downsampling e a previsão com o tamanho original da imagem, tornando a

informação utilizável.

(7)

$\begin{bmatrix} 5 & 6 \\ 3 & 2 \end{bmatrix}$; SAÍDA: 4×4

a) $\begin{bmatrix} 5 & 5 & 6 & 6 \\ 5 & 5 & 6 & 6 \\ 3 & 3 & 2 & 2 \\ 3 & 3 & 2 & 2 \end{bmatrix}$

b) $\begin{bmatrix} 5 & 0 & 6 & 0 \\ 0 & 0 & 0 & 6 \\ 3 & 0 & 2 & 0 \\ 0 & 0 & 0 & 6 \end{bmatrix}$

c) $\begin{bmatrix} 3 & 1 & 1 & 4 \\ 2 & 1 & 3 & 2 \\ 3 & 7 & 1 & 7 \\ 3 & 2 & 2 & 4 \end{bmatrix}$ MAX-POOL $\begin{bmatrix} 3 & 4 \\ 7 & 7 \end{bmatrix}$

$\begin{bmatrix} 5 & 6 \\ 3 & 2 \end{bmatrix}$ UNPOOLING $\begin{bmatrix} 5 & 0 & 0 & 6 \\ 0 & 0 & 0 & 0 \\ 0 & 3 & 0 & 2 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

d) $\begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix}$

$\rightarrow 5 \cdot \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 0 & 5 \\ 10 & 15 \end{bmatrix}$

$\rightarrow 6 \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 0 & 6 \\ 12 & 18 \end{bmatrix}$

$\rightarrow 3 \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 0 & 3 \\ 6 & 9 \end{bmatrix}$

$\rightarrow 2 \begin{bmatrix} 0 & 1 \\ 2 & 3 \end{bmatrix} = \begin{bmatrix} 0 & 2 \\ 4 & 6 \end{bmatrix}$

$\begin{bmatrix} 0 & 5+0 & 6 \\ 10+0 & 15+12+3+0 & 18+2 \\ 6 & 9+4 & 6 \end{bmatrix}$

$=$

$\begin{bmatrix} 0 & 5 & 6 \\ 10 & 30 & 20 \\ 6 & 13 & 6 \end{bmatrix}$

LISTA TEÓRICA - DETECÇÃO DE OBJETOS

THIAGO MALTA COUTINHO - 20144123335

① $y = \begin{bmatrix} 1 \\ 0,4 \\ 0,8 \\ 0,2 \\ 0,2 \\ 0 \\ 1 \\ 0 \end{bmatrix}$ → OBJECT? (P_c); → b_x; → b_y; → b_h; → b_w; } CLASSES ONE-HOT

② $y = [0\ ?\ ?\ ?\ ?\ ?\ ?\ ?]^T$

Os pesos da rede não serão atualizados.

③ $L(w) = \begin{cases} \sum_i (y_i - \hat{y}_i)^2, & \text{se } y_1 = 1 \\ (y_1 - \hat{y}_1)^2, & \text{se } y_1 = 0 \end{cases}$

④ Como é uma classificação binária, a última camada será uma regressão logística. Apenas b_x e b_y são necessários, visto que as dimensões da lata são fixas.

$y = [P_L\ b_x\ b_y]^T$

$b_x, b_y \rightarrow (x, y) \rightarrow$ centro da bounding box/lata.

$P_L \rightarrow$ Probabilidade de haver uma lata.

⑤ Como é uma imagem 2D, cada ponto tem coordenada (x, y), logo $2N$.

⑥ É necessário fornecer rótulos ONE-HOT e as bounding boxes com coordenadas (b_x, b_y) associados ao centro e b_w e b_h associadas à largura da caixa e altura, respectivamente.

(7) Aumentar o stride reduz o custo computacional pois menos valores são processados mas diminui a acurácia, pois menos informação é processada. As alternativas são: convolução dilatada, sliding window convolucional e region proposal.

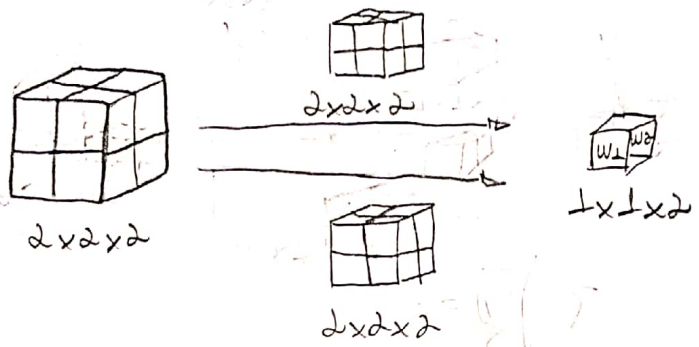
(8) R-CNN faz a proposição de regiões e então o feature extraction.
Já o Fast R-CNN faz o feature extraction na imagem completa e region proposal na imagem original. O ROI-POOL é utilizado para alinhar o mapa com a ROI e transformar as dimensões para o tamanho de saída necessário.
A velocidade do forward é aumentada significativamente.

(9) A faster R-CNN utiliza uma rede CNN para propor as regiões de interesse, diferente da Fast R-CNN, que utiliza Selective Search.

(10) ROI POOLING com alinhamento produz um mapa de ativação de saída compatível com a entrada de um classificador, podendo ser uma CNN. Ele padroniza a dimensão de saída para diferentes tamanhos de ROI.

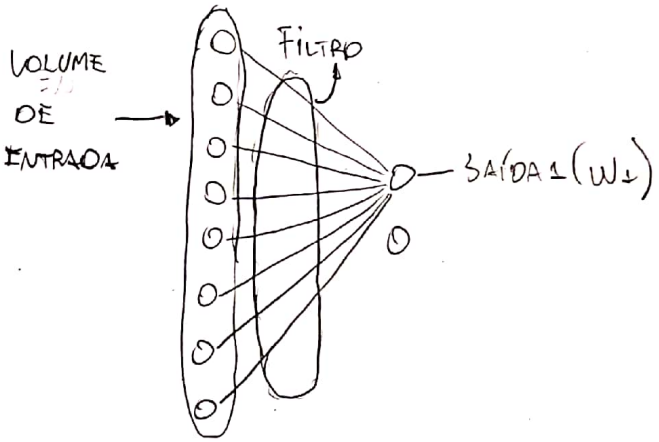
LISTA TEÓRICA - DETECÇÃO / YOLO

THIAGO MATTA LOUTINHO - 2014123335

① 

$$W_k = g\left(\sum_{i,j,c=1}^{2} w_{ijc} F_{ijck} + b_k\right)$$

São 8 entradas, (2x2x2)x2 pesos, 2 saídas. Cada filtro é equivalente a um neurônio e os valores F_{ijck} são as conexões da entrada com o filtro/neurônio correspondente. É uma camada FC com 2 neurônios.



② $IoU = \frac{A \cap B}{A \cup B}$; $A \cap B = 1$; $A \cup B = 9$ $\Rightarrow IoU = 1/9$

③ $p(x) \leq 0,4 \rightarrow$ DESCARTA

$IoU > 0,5 \rightarrow$ SOBREPOSIÇÃO

5 CAIXAS NO TOTAL

$p(\text{MOTORCYCLE}) = 0,58 \rightarrow OK$

$p(\text{PEDESTRIAN}) = 0,98 \rightarrow OK$

$p(\text{CAR1}) = 0,126 \rightarrow X$

$p(\text{CAR2}) = 0,63 \rightarrow$ ~~OK~~

$p(\text{CAR2}) = 0,173 \rightarrow OK$

$IoU > 0,5 \rightarrow OK$

$p(\text{TREE}) = 0,46 \rightarrow OK$

$p(\text{TREE}) = 0,76 \rightarrow OK$

$IoU > 0,5 \rightarrow X$

④ $P(x) \leq 0,6$

P(MOTORCYCLE) = 0,58 → X

P(CAR 2) = 0,62 → ~~OK~~ ⎫
P(CAR 2) = 0,73 → OK ⎭ IoU > 0,5 OK

3 CAIXAS NO TOTAL

P(CAR 1) = 0,26 → X

P(PEDESTRIAN) = 0,98 → OK

P(TREE) = 0,46 → X

P(TREE) = 0,74 → OK

⑤ Objetos distintos sobrepostos

Bounding box com maior probabilidade não é o mais adequado.

⑥ Apenas a célula com anchor que identifica o objeto.

⑦ $\{[P_c\ b_x\ b_y\ b_h\ b_w\ \text{ONE-HOT}_{20}] \times 5\}\ (\text{grid}) = (2 \times 5) \times (19 \times 19)$

⑧ YOLO nem sempre produz os melhores resultados. Ele possui dificuldades com proporções distorcidas, limitações de tamanho de grid.

⑨ Porque a sobreposição entre os objetos resultará em probabilidade baixa para o objeto sobreposto, que será filtrado pelo non-max suppression.

⑩ Objetos de tamanhos diferentes e em posições diversas serão detectados.

⑪ Manualmente → simplicidade

k-means → melhor ajuste das anchor boxes.

LISTA TEÓRICA – AUTOENCODERS

THIAGO MALTA COUTINHO – 2014123335

(1) Aprendizado supervisionado possui dados rotulados, já o não-supervisionado, não.

Supervisionado: classificação, regressão.

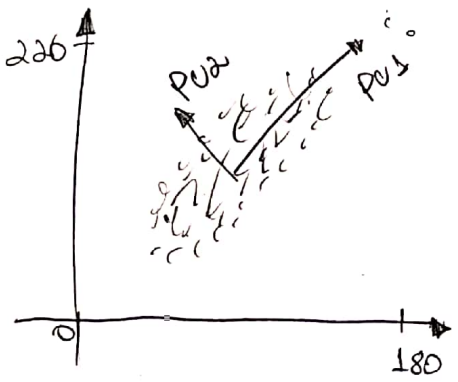
Não-supervisionado: Clustering, redução de dimensionalidade.

(2)

$$\Sigma\vec{v} = \lambda\vec{v} \quad ; \quad \vec{v} = \{+1, -1, 1\rangle/\sqrt{3}$$

$$\Sigma = \begin{bmatrix} 1 & \rho & \rho \\ \rho & 1 & \rho \\ \rho & \rho & 1 \end{bmatrix} \qquad \Sigma\vec{v} = \begin{bmatrix} 1+\rho+\rho \\ 2\rho+1 \\ 2\rho+1 \end{bmatrix} \rightarrow (2\rho+1) \rightarrow \text{AUTOVALOR}$$

(3)



(4) A ideia principal é aprender codificar a variável de entrada em um espaço latente e decodificá-la com a menor perda de informação possível. Nesse processo o espaço latente é aprendido e pode ser utilizado para diversas aplicações como feature engineering. O autoencoder tem a estrutura de encoder e decoder, ambas são redes neurais.

(5).
- Detecção de anomalias baseado no erro de reconstrução.
- Feature engineering a partir do espaço latente aprendido.
- Pré-treinamento para uma rede que será concatenada na saída do decoder

(6) Apenas aprenderá a função identidade com alguns ruídos.

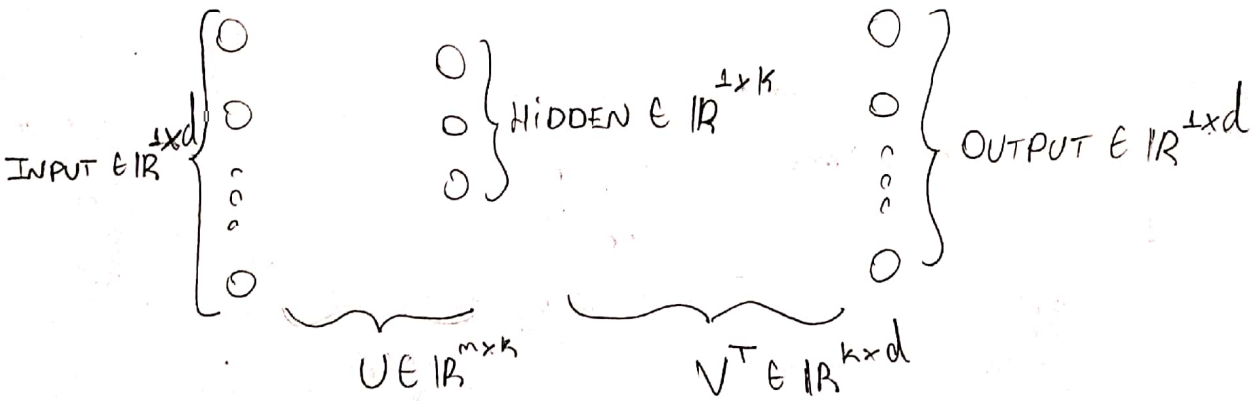
(7) Evitar que o autoencoder aprenda a função identidade ou um mapeamento direto. O objetivo é aprender um espaço latente.

(8) $\|x\|^2 = \sum_{i=1}^{m} (x_i)^2 \leq 1$ → normalização do espaço de entrada.

Para que o produto interno ou a semelhança dos vetores seja a maior possível, o vetor w também possuirá norma unitária.

(9) Como o encoder é uma convolução, o decoder terá uma convolução transposta com filtro de mesmo tamanho que o filtro de entrada.

(10) $D \in \mathbb{R}^{n \times d}$
$U \in \mathbb{R}^{n \times k}$
$V \in \mathbb{R}^{d \times k}$
→ $D \approx UV^T$



LISTA TEÓRICA - VAE

THIAGO MALTA LOUTINHO - 2014123335

① Sparse autoencoder: mapeia a entrada para um espaço de maior dimensão. Esse espaço é esparso e o objetivo é gerar vetores base por meio da LOSS com regularização L1.

Denoising Autoencoder: o objetivo é filtrar ruídos presentes na entrada dos dados. Cada um tipo de ruído que se quer filtrar, ele é adicionado a entrada do modelo e a saída é a entrada sem ruído. A rede então aprende a remover os ruídos.

Variational Autoencoder: O modelo aprende a distribuição dos dados de entrada a partir da modelagem de uma distribuição de variáveis latentes da entrada.

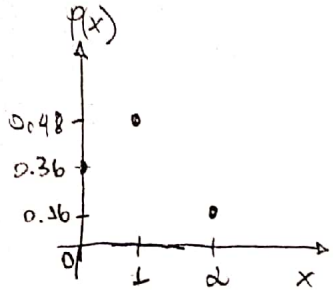
Contractive Autoencoder: A função de perda é desenvolvida para que pequenas variações na entrada produza pequenas variações no espaço latente.

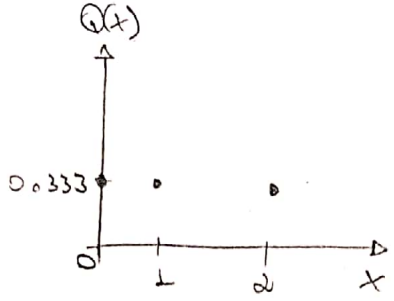
② Um modelo generativo é um modelo que aprende a distribuição dos dados, possibilitando gerar novos exemplos da distribuição aprendida. O autoencoder é um modelo que mapeia uma entrada para uma saída através de um espaço latente, ele não aprende as distribuições dos dados. O autoencoder variacional aprende as distribuições do espaço latente e da saída, por isso é generativo.

③ $$l_i(\theta,\phi) = -E_{z\sim g_\theta(z|x_i)}\left[\log p_\phi(x_i|z)\right] + KL\left(q_\theta(z|x_i)\,\|\,p(z)\right)$$

④ $$D_{KL}(P\|Q) = -\sum_x P \log Q + \sum_x P \log P$$

(5)





$$D_{KL}(Q\|P) = \sum_x P_Q(x) \log \frac{P_Q(x)}{P_P(x)} = 0.333 \cdot \left[\log\left(\frac{0.333}{0.36}\right) + \log\left(\frac{0.333}{0.48}\right) + \log\left(\frac{0.333}{0.16}\right)\right]$$

$$D_{KL}(P\|Q) = 0.36 \log\left(\frac{0.36}{0.333}\right) + 0.48 \log\left(\frac{0.48}{0.333}\right) + 0.16 \log\left(\frac{0.16}{0.333}\right)$$

$$\boxed{D_{KL}(Q\|P) \neq D_{KL}(P\|Q)}$$

---

(7)

$$\mu_\phi(x) = ReLU(2\,ReLU(3x-1))$$

$$\Sigma_\phi(x) = ReLU(ReLU(3x+1)+1)$$

$$\mu_\theta(z) = ReLU(-z+12)$$

$$\Sigma_\theta(z) = ReLU(2z-10)$$

a) $x = 2$

$\varepsilon = [0.5 \;\; 0 \;\; -0.5]$

$z_i = \mu_\phi(x) + \Sigma_\phi(x) \odot \varepsilon_i$

$z_1 = \underbrace{Relu(2 \cdot Relu(6-1))}_{10} + \underbrace{Relu(Relu(6+1)+1)}_{8} \cdot 0.5 = 10+4 = 14$

$z_2 = 10 + 8.0 = 10$

$z_3 = 10 + 8 \cdot -0.5 = 10-4 = 6$

$\vec{z} = [14 \;\; 10 \;\; 6]$

---

b) $x_i = \mu_\theta(z), \Sigma_\theta(z)$

$x_1 = [Relu(-z+12), Relu(2z-10)] = [0, 18]$

$x_1 = N(0, 18)$

$x_2 = N(2, 10)$

$x_3 = N(6, 2)$

⑥ Separando a parte determinística da aleatória. A parte determinística é a média, o desvio padrão e a incerteza multiplicativa. A parte estocástica é modelada como um "ruído" normal com média zero e desvio padrão unitário.

⑧ $p(x)$ é intratável, pois não conseguimos estimá-lo diretamente.

⑨ ???