Exercícios Deep Learning
Aula 21 e 22

October 3, 2019

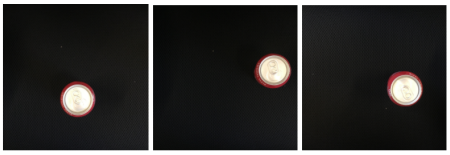
1 Detecção de Objetos

1- Você está construindo um algoritmo de classificação e localização de objetos de 3 classes. As classes são: pedestre (c = 1), carro (c = 2), motocicleta (c = 3). Qual seria o rótulo aproximado da imagem a seguir? Considere a imagem como sendo um quadrado de lado igual a 1.

2- Continuando com o problema anterior, qual deve ser o rotulo y da imagem abaixo? Lembre-se de que "?" Significa "não se importa", o que significa que a função de perda da rede neural não se importa com o que a rede neural fornece para esse componente da saída.

3- O somatório dos erros quadráticos é uma possível função de perda que pode ser aplicada ao trabalhar com detecção de objetos. Escreva a fórmula de perda considerando os dois casos mostrados nos Exercícios 1 e 2, ou seja, quando na imagem existe um objeto que pertence a uma das classes e quando não há nenhum.

4- Você está trabalhando em uma tarefa de automação de fábrica. Seu sistema verá uma lata de refrigerante descendo por uma correia transportadora, e você quer que ela tire uma foto e decida se (i) há uma lata de refrigerante na imagem e, em caso afirmativo, (ii) sua bounding box. Como a lata de refrigerante é redonda, a bounding box é sempre quadrada e o refrigerante sempre aparece com o mesmo tamanho na imagem. Há no máximo uma lata de refrigerante em cada imagem. Aqui estão algumas imagens típicas no seu conjunto de treinamento:



Qual é o conjunto mais apropriado de unidades de saída para sua rede neural? Considere a função de ativação que poderá ser utilizada nessa tarefa de classificação (i) e a dimensão do vetor de saída bem como o que cada posição representa.

5- Se você construir uma rede neural que recebe como entrada uma imagem da face de uma pessoa e imprima N pontos de referência na face (suponha que a imagem de entrada sempre contenha exatamente uma face), quantas unidades de saída a rede terá?

6- Ao treinar um dos sistemas de detecção de objetos, você precisa de um conjunto de treinamento que contenha muitas imagens do(s) objeto(s) que você deseja detectar. Além das imagens e o rótulo dos objetos, explique quais outras informações precisam ser fornecidas para o treinamento supervisionado da rede de detecção de objetos.

7- Suponha que você esteja aplicando um classificador de janelas deslizantes (implementação não-convolucional). Quais são as consequências de aumentar o stride em termos de acurácia e custo computacional? Qual seria uma boa alternativa para o uso de janelas deslizantes?

8- Qual a principal diferença entre o método R-CNN e o Fast R-CNN. Qual é a vantagem de utilizar o último em relação ao primeiro?

9- Qual a principalmente diferença entre a Fast R-CNN e a Faster R-CNN?

10- Qual o objetivo da camada de "RoI Pooling"? Porque ela é necessária em arquiteturas que utilizam de abordagens baseadas em regiões?

Solução

1- y=[1,0.3,0.7,0.3,0.3,0,1,0], onde:

- 1, no caso a imagem contem um objeto
- Posição b_x a 30% da largura total da imagem
- Posição b_y a 70% da altura total da imagem
- Tamanho total do objeto (carro) de $b_w = 0.3$ em relação a largura total da imagem
- Tamanho total do objeto (carro) de $b_h = 0.3$ em relação a altura total da imagem
- Vetor de classes apenas ativado na classe carro: $c_1 = 0, c_2 = 1, c_3 = 0$

2- y=[0,?,?,?,?,?,?,?], nessa imagem não é encontrada nenhuma das classes reconhecidas pelo algoritmo.

3-
$L(\hat{y}, y) = (\hat{y_1} - y_1)^2 + (\hat{y_2} - y_2)^2 + ... + (\hat{y_8} - y_8)^2$ se $y_1 = 1$.
$L(\hat{y}, y) = (\hat{y_1} - y_1)^2$ se $y_1 = 0$.

4- Unidade logística, b_x e b_y. A unidade logística vai servir apenas para a tarefa de classificação, enquanto as coordenadas b_x e b_y nos dizem a posição do objeto. Considerando que o refrigerante aparece sempre com o mesmo tamanho, b_h e b_w são desnecessários.

5- 2N, Cada ponto de referencia deve ser ancorado com b_x, b_y.

6- As caixas delimitadoras precisam ser fornecidas no conjunto de treinamento.

7- Aumentar o stride aumenta a eficiência computacional, porém diminui a acurácia. Duas possíveis alternativas são: implementação convolucional de sliding window ou não classificar todas as regiões (region propposal).

8- O R-CNN usa o algoritmo de segmentação selective search para propor regiões de interesse, i.e. regiões que possivelmente contém um objeto. Cada uma dessas regiões é achatada (warped) e passada por uma rede convolucional para obter um mapa de features correspondente (um processo muito custoso). O Fast R-CNN passa a imagem uma única vez pela ConvNet, obtendo um mapa de features. As regiões selecionadas a partir da imagem pelo Selective Search são então "recortadas" no mapa de features e, em seguida, tem sua dimensão reduzida através da operação de RoI Pooling. Passar a imagem uma única vez pela ConvNet reduz substancialmente o tempo da detecção de objetos.

9- O gargalo da Fast R-CNN é a proposta de regiões através do selective search. O Faster R-CNN usa uma rede convolucional para propor regiões (region proposal network ou RPN), que pode ser vista como uma rede cujo papel é determinar se há um objeto centrado em um determinado pixel. Isto torna a Faster R-CNN bem mais eficiente.

10- O "RoI Pooling" transforma regiões propostas de tamanhos diferentes em uma lista de matrizes com o mesmo tamanho. Isso torna possível utilizar redes convolucionais para classificar as regiões.