

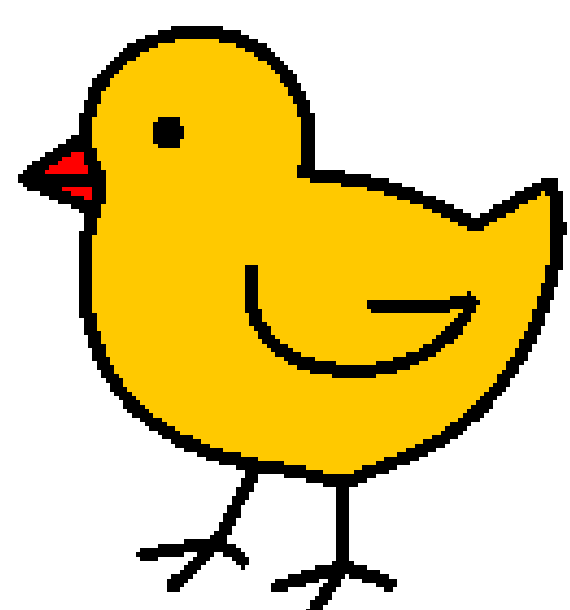
Kuriatko alebo pes?

Nový algoritmus na bipartíciu príznakových máp



Abstrakt

Jedným zo spôsobov, akým dokáže systém umelej inteligencie pomenovať predmet na obraze je spriahnutie extraktora príznakov z obrazu s extraktorom príznakov textu. V takomto spoločnom priestore obrazových a textových príznakov je potom pre príznaky obrazu hľadať akým príznakom textu sú podobné. Robí to takto napríklad model CLIP, pričom používa kosínusovú podobnosť medzi príznakovým vektorom textu a klasifikačným príznakovým vektorom obrazu. Tento prístup je pomerne ľahké pomýliť, napríklad keď modelu prezentujeme obrázok kuriatka a pod ním text "pes", povie, že na obrázku je pes. Pritom ako vedľajší produkt vzniká mapa príznakov jednotlivých regiónov obrazu. Naším vkladom do problematiky je nový algoritmus, ktorý túto mapu dokáže rozložiť na časti, ktorá zodpovedá kuriatku a časť, ktorá zodpovedá nápisu "pes". Takáto bipartícia je v princípe NP-ťažký problém, pre ktorý navrhli Shi a Malik v roku 2000 približný algoritmus, kvadratický od počtu regiónov. Prerobili sme ho na lineárny a dali mu podobu, v ktorej môže byť zabudovaný priamo do neurónovej siete. Vďaka bipartícii obrázka je potom možné použiť pôvodný model tak, aby jednu časť pomenoval správne: "kuriatko" a druhú správne: "pes".



DOG

dog 90% bird 8%

Keď porovnáme kosínusovou podobnosťou príznakový vektor tohto obrazu s príznakovými vektormi pomenovaní kategórii dátovej sady COCO a z 80 možností vyberieme dve s najvyššou pravdepodobnosťou, vyjde nám, že na tomto obrázku je s vysokou pravdepodobnosťou pes a len s pomerne nízkou vták.

Hoci je tento obrázok kontroverzný, človeku nerobí problém vnímať, že na obrázku je vták i nápis „pes“



bird 97% person 0%
dog 96% hotdog 2%

Keď mapu príznakov na ktoré obrázok premení model CLIP, preškálovanú zo 7×7 na pôvodné rozlíšenie obrázka podrobíme bipartícii, rozdelí obrázok najprv na okrajovú a významovú časť. Keď potom delíme významovú časť, dostaneme približne časť v ktorej je kuriatko a v ktorej je nápis „pes“.

Keď potom klasifikujeme tieto časti, podľa vyššej pravdepodobnosti vidíme, že na obrázku máme naozaj dve rôzne veci: vtáka a psa

Algorithm 1 Fast Normalized Cut

```
1: function UNPR( $u, v$ )
2:   return  $v(v^T u)$ 
3: end function
4:
5: function FAST_NCUR( $features, mask, iterations = 4$ )
6:    $\triangleright features$  is a map of the shape  $C \times H \times W$ 
7:    $\triangleright mask$  is a bool map of the shape  $H \times W$ 
8:    $f \leftarrow features[mask]$   $\triangleright$  picking up features into  $N \times C$ 
9:    $y \leftarrow init()[mask]$   $\triangleright$  initialization of  $H \times W$  with 1 or  $-1(0)$ , into  $N$ 
10:   $d \leftarrow UNPR(1_N, f)$   $\triangleright$  initialize parameters of the projection
11:   $z_0^{norm} \leftarrow normalize(d^{\frac{1}{2}})$ 
12:   $pr \leftarrow (d^{-\frac{1}{2}} 1_C^T) \circ f$ 
13:  for  $i \leftarrow 1 \rightarrow iterations$  do  $\triangleright$  power iteration
14:     $y \leftarrow UNPR(y, pr) - UNPR(y, z_0^{norm})$   $\triangleright$  projection of  $y$ 
15:     $y \leftarrow normalize(y)$   $\triangleright$  normalization of  $y$ 
16:  end for
17:   $bipartition[:, :] \leftarrow UNKNOWN$   $\triangleright$  a bool map  $H \times W$ 
18:   $bipartition[mask] \leftarrow y \geq 0$   $\triangleright$  distribution of  $N$  into  $H \times W$ 
19:  return  $bipartition$   $\triangleright$  returns a bool map of the shape  $H \times W$ 
20: end function
```

Prínosy

- Algoritmus pre približný výpočet normalizovaného rozdelenia $O(t f n)$ miesto doterajších $O(t f n^2)$, navyše t malé (t .. počet iterácií, f .. počet príznakov, n .. počet regiónov / pixelov)
- Aplikácie tohto algoritmu ako stavebného prvku hlbokých neurónových sietí

Referencie

Shi, J., & Malik, J. (2000). *Normalized cuts and image segmentation*. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), 888–905

Wang, X., Girdhar, R., Yu, S. X., & Misra, I. (2023). *Cut and Learn for Unsupervised Object Detection and Instance Segmentation (CutLER)*. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2023)* (pp. 3124–3134).

