Deep Convolutional Ranking for Multilabel Image Annotation

Xie, Jiayu
Liu, Xiao

Introduction

- features based on Deep Neural Networks
- a significant performance gain could be obtained by combining convolutional architectures with approximate top-k ranking objectives
- multiple labels to one image

Figure 1: Sample images from the NUS-WIDE dataset, where each image is annotated with several tags.

Dataset

NUS-WIDE: Image from Flickr, 81 tags
Each image is resized to 256*256, then 220*220 patches are extracted from the whole image, at the center and the four corners to provide an augmentation of the dataset

Flickr tag:

uk hairy nature field animal scotland ginger cow cattle unfound coo mugdock highlandcow

NUS-WIDE tag:

animal cloud cow grass

Network Architecture

Five convolutional layers, some followed by max-pooling layers
three fully-connected layers with final softmax

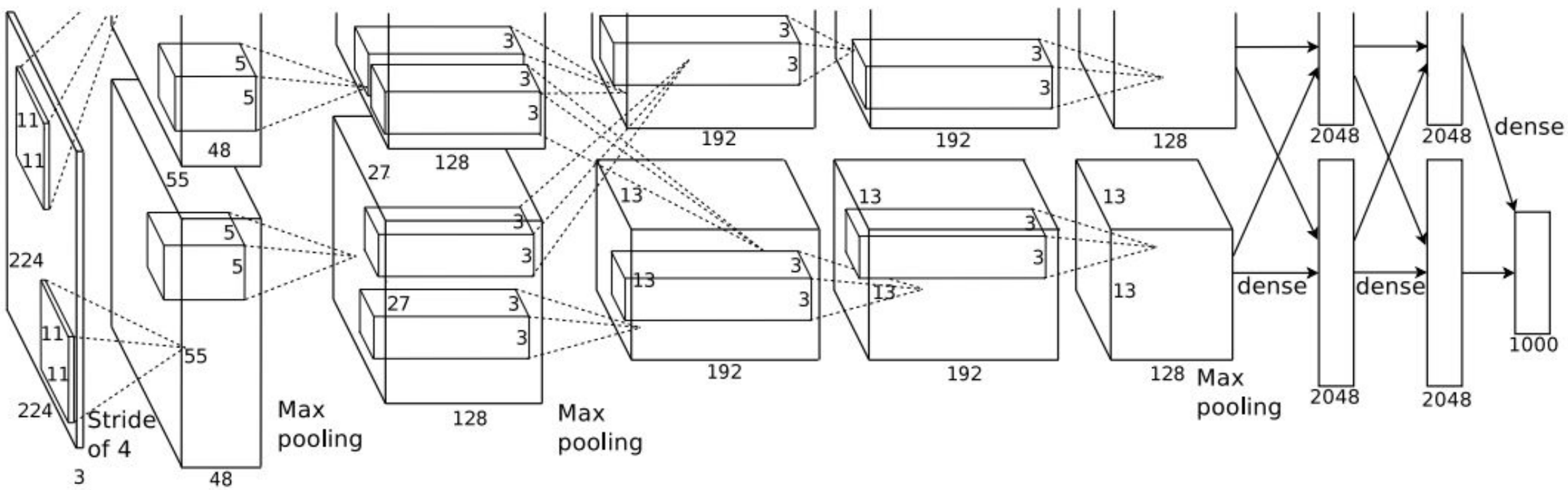

Figure 2: An illustration of the architecture of our CNN, explicitly showing the delineation of responsibilities between the two GPUs. One GPU runs the layer-parts at the top of the figure while the other runs the layer-parts at the bottom. The GPUs communicate only at certain layers. The network's input is 150,528-dimensional, and the number of neurons in the network's remaining layers is given by 253,440–186,624–64,896–64,896–43,264–4096–4096–1000.

Multilabel Ranking Losses (1/3)

Softmax

The output of f (x) is a scoring function of the data point x

- $f_j(x_i)$ means the activation value for image x_i and class j
- c_+ denotes the number of positive labels for each image
- m maybe is batch-size

$$p_{ij} = \frac{\exp(f_j(\boldsymbol{x}_i))}{\sum_{k=1}^{c} \exp(f_k(\boldsymbol{x}_i))},$$

$$J = -\frac{1}{m} \sum_{i=1}^{n} \sum_{j=1}^{c} \bar{p}_{ij} \log(p_{ij}) = -\frac{1}{m} \sum_{i=1}^{n} \sum_{j=1}^{c_+} \frac{1}{c_+} \log(p_{ij})$$

Multilabel Ranking Losses (2/3)

Pairwise Ranking

- The output of f (x) is a scoring function of the data point x
- $f_j(x_i)$ means the activation value for image x_i and class j
- c_+ is the positive labels and c_ is the negative labels
- rank the positive labels have higher scores than negative labels

$$J = \sum_{i=1}^{n} \sum_{j=1}^{c_+} \sum_{k=1}^{c_-} \max(0, 1 - f_j(\boldsymbol{x}_i) + f_k(\boldsymbol{x}_i)),$$

Multilabel Ranking Losses (3/3)

Weighted Approximate Ranking (WARP)

- The output of f (x) is a scoring function of the data point x
- $f_j(x_i)$ means the activation value for image x_i and class j
- c_+ is the positive labels and c_ is the negative labels
- rank the positive labels have higher scores than negative labels

$$J = \sum_{i=1}^{n} \sum_{j=1}^{c_+} \sum_{k=1}^{c_-} L(r_j) \max(0, 1 - f_j(\boldsymbol{x}_i) + f_k(\boldsymbol{x}_i)).$$

$$L(r) = \sum_{j=1}^{r} \alpha_j, \text{ with } \alpha_1 \geq \alpha_2 \geq \ldots \geq 0.$$

Baseline

Visual Feature:

- GIST (Gradient-domain Image STitching): 960d
- SIFT (Scale-Invariant Feature Transform): 5000d *6

  2 methods (dense, Harris corner) *

  3 descriptors (SIFT, CSIFT, RGBSIFT)
- HOG (Histogram of Oriented Gradient): 5000d
- Color (RGB histogram): 512d

Use Kernel PCA to reduce each dimensionality to 500d

(36,472 ⇒ 4,500 dimensional feature vector)

Learning Algorithms:

- Visual Feature + kNN
- Visual Feature + SVM

Experiments

method / metric	per-class recall	per-class precision	overall recall	overall precision	$N+$
Upper bound	97.00	44.87	82.76	66.49	100.00
Visual Feature + kNN	19.33	**32.59**	53.44	42.93	91.36
Visual Feature + SVM	18.79	21.51	35.87	28.82	82.72
CNN + Softmax	31.22	31.68	59.52	47.82	98.76
CNN + Ranking	26.83	31.93	58.00	46.59	95.06
CNN + WARP	**35.60**	31.65	**60.49**	**48.59**	**96.29**

Table 1: Image annotation results on NUS-WIDE with $k = 3$ annotated tags per image. See text in section 5.4 for the definition of "Upper bound".

method / metric	per-class recall	per-class precision	overall recall	overall precision	$N+$
Upper bound	99.57	28.83	96.40	46.22	100.00
Visual Feature + kNN	32.14	**22.56**	66.98	32.29	95.06
Visual Feature + SVM	34.19	18.79	47.15	22.73	96.30
CNN + Softmax	48.24	21.98	74.04	35.69	98.76
CNN + Ranking	42.48	22.74	72.78	35.08	97.53
CNN + WARP	**52.03**	22.31	**75.00**	**36.16**	**100.00**

Table 2: Image annotation results on NUS-WIDE with $k = 5$ annotated tags per image. See text in section 5.4 for the definition of "Upper bound".

$$\text{per-class recall} = \frac{1}{c}\sum_{i=1}^{c}\frac{N_i^c}{N_i^g}, \qquad \text{per-class precision} = \frac{1}{c}\sum_{i=1}^{c}\frac{N_i^c}{N_i^p}$$

$$\text{overall recall} = \frac{\sum_{i=1}^{c} N_i^c}{\sum_{i=1}^{c} N_i^g}, \qquad \text{overall precision} = \frac{\sum_{i=1}^{c} N_i^c}{\sum_{i=1}^{c} N_i^p}.$$

Result (WARP)

Image	Ground truth	Predictions	Image	Ground truth	Predictions
	Boat Cloud Ocean Vehicle Water	Lake Ocean Cloud Sky Water		Grass	Cloud House Animal Grass Sky
	Cloud Flower plant Sky Valley	Grass Plant Cloud Flower Sky		Road	Grass Animal Bird Food Toy
	Beach Rock Sky Sunset Water	Rock Water Ocean Cloud Sky		Cloud Sky Snow Sunset Tree	Tree Snow Cloud Sky Water
	Animal Cloud Cow Grass	Horse Animal Cloud Grass Sky		Cloud Mountain Rock Sky	Valley Mountain Building Sky Cloud

Figure 4: Qualitative image annotation results obtained with WARP.

Results

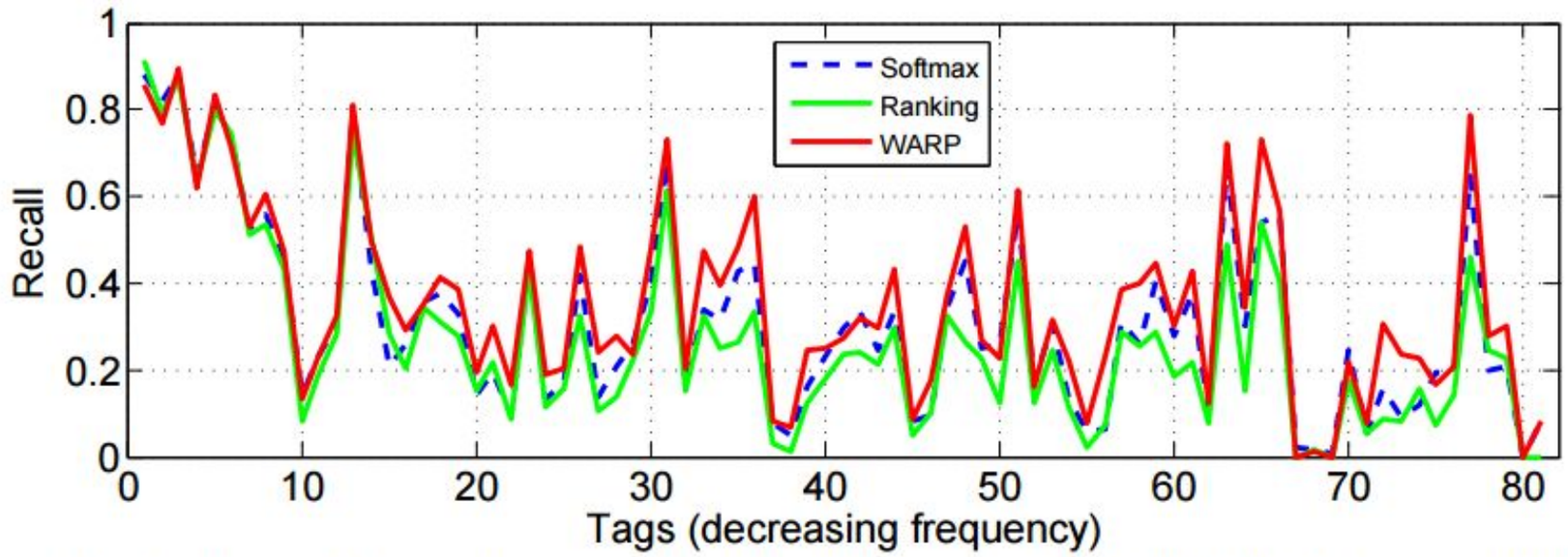

Figure 2: Analysis of per-class recall of the 81 tags in NUS-WIDE dataset with $k = 3$.

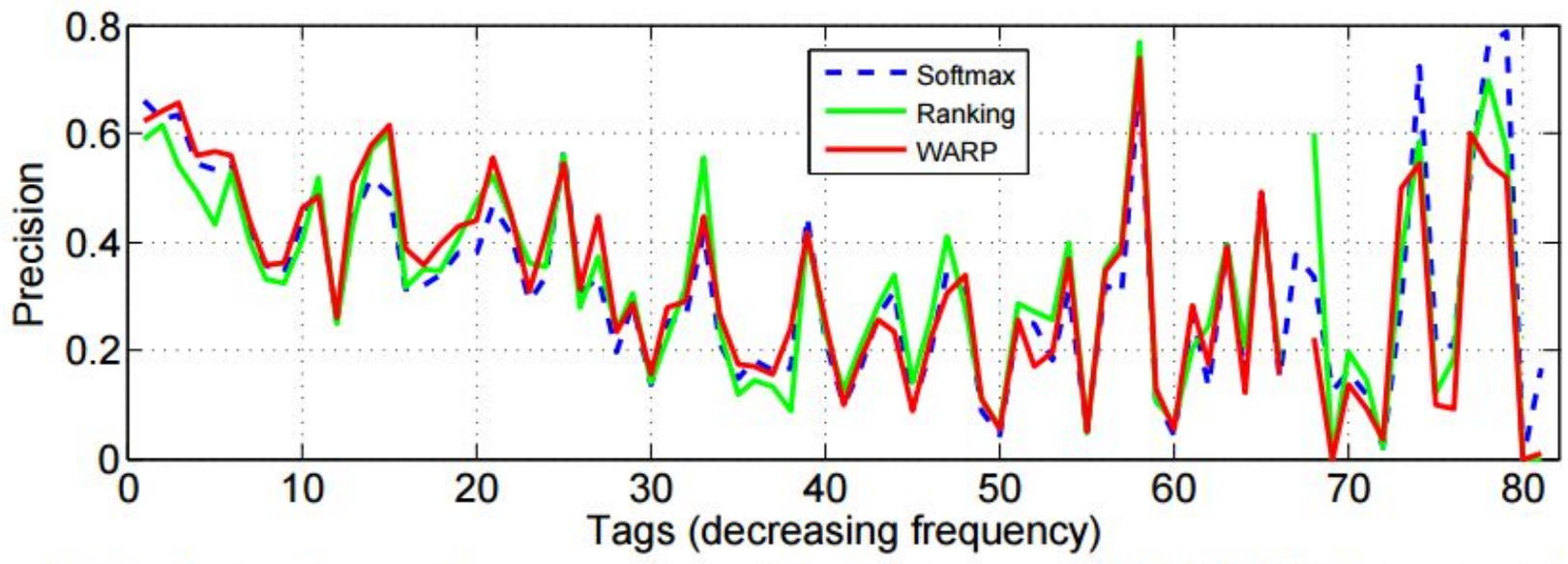

Figure 3: Analysis of per-class precision of the 81 tags in NUS-WIDE dataset with $k = 3$.

Questions?