# Deep Neural Networks in Fully Connected CRF for Image Labeling with Social Network Metadata

Chengjiang Long Roddy Collins Eran Swears Anthony Hoogs
Kitware Inc.
28 Corporate Drive, Clifton Park, NY 12065
{chengjiang.long, roddy.collins, eran.swears, anthony.hoogs}@kitware.com

## Abstract

*We propose a novel method for predicting image labels by fusing image content descriptors with the social media context of each image. An image uploaded to a social media site such as Flickr often has meaningful, associated information, such as comments and other images the user has uploaded, that is complementary to pixel content and helpful in predicting labels. Prediction challenges such as ImageNet [6] and MSCOCO [19] use only pixels, while other methods make predictions purely from social media context [21]. Our method is based on a novel fully connected Conditional Random Field (CRF) framework, where each node is an image, and consists of two deep Convolutional Neural Networks (CNN) and one Recurrent Neural Network (RNN) that model both textual and visual node/image information. The edge weights of the CRF graph represent textual similarity and link-based metadata such as user sets and image groups. We model the CRF as an RNN for both learning and inference, and incorporate the weighted ranking loss and cross entropy loss into the CRF parameter optimization to handle the training data imbalance issue. Our proposed approach is evaluated on the MIR-9K dataset and experimentally outperforms current state-of-the-art approaches.*

## 1. Introduction

Multimedia data such as images and videos are being produced and shared at an unprecedented and accelerating pace in recent years. For example, on YouTube, video data is currently being uploaded at the rate of approximately 30 million hours a year. This drives a strong need to develop automatic tools to help users understand, organize, and retrieve images and videos from vast collections. While recent advances have been impressive, real-world multimedia, especially those shared on the image-sharing platform Flickr, can still be challenging to index and retrieve using only visual information, due to complex content, partial occlusion, and diverse styles and quality.

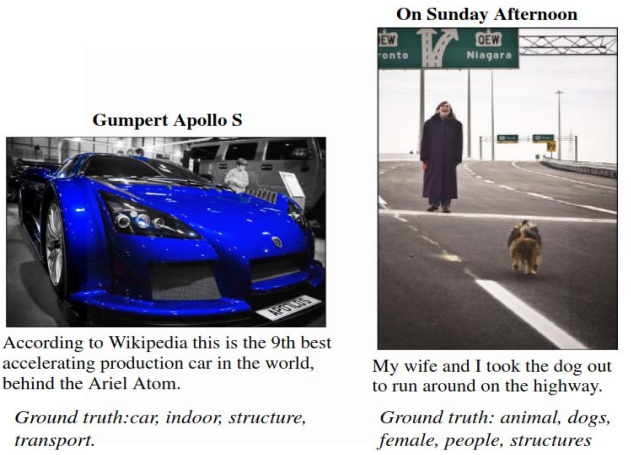



Figure 1: Two sample images with the title in bold, image description and the corresponding ground-truth labels in italic from the MIR-9K dataset. The goal of this paper is to make full use of such text information as well as the link-based metadata like user sets and image groups to boost the quality of image labeling.

Images in social media do not exist in isolation. As illustrated in Figure 1, a rich social multimedia database contains images, text information such as image title, description, and comments, as well as user information (*e.g.*, username, location, network of contacts), user image gallery, uploader-defined groups, and links between shared content. Most image recognition and label prediction methods depend entirely or primarily on pixel content and do not make full use of commonly-available multimedia information to aid in automatic image labeling. We hypothesize that using social media context jointly with pixel information should improve the state-of-the-art in image labeling. Furthermore, we seek to understand the relative contribution of pixels, text and other information in predicting image labels.

978-1-7281-1975-5/19/$31.00 ©2019 IEEE
DOI 10.1109/WACV.2019.00176

We define our problem as an automatic image labeling based on inferring content labels $Y$, conditioned on an image $I$, and other related metadata information $M$. Our proposed solution is illustrated in Figure 2, which introduces a novel deep fully connected Conditional Random Field framework (we call "DCRF") that uses deep neural networks to compute the joint probability $P(Y|I, M)$. CRFs have been commonly used in image segmentation problems where the model has one hidden node per pixel or grid-cell and a vector of hidden nodes for a single image. Instead, we abstract up one layer and define one hidden node per image, instead of per pixel, over the entire dataset of images to form an image relationship graph. This results in having the vector of hidden nodes over the whole dataset with one node per image.

For pixel content descriptors, we exploitation popular image classification CNNs to extract a visual feature vector for each image or CRF node. To incorporate image title, comments, captions, and other text, instead of using high-frequency words as tags [21, 12], we treat the text information as an unorganized and incoherent sentence and then fine-tune a popular network for sentence classification [15] as a text-level neural network to extract text features. In addition to textual similarity based on the text feature, we use associative metadata such as user sets and image groups to determine the edge weights in the fully connected CRF graph.

Our fully connected CRF establishes pairwise potentials on all pairs of images over the entire dataset. It combines the strengths of both CNN and CRF based graphical models in a unified framework. Inspired by Zheng *et al.* [33], we formulate a mean-field approximation inference [16] for the CRF and model it as a Recurrent Neural Network (RNN). Hence our DCRF is an end-to-end CNN-RNN framework, incorporating the advantages of both convolutional and recurrent neural networks while enabling standard back-propagation during training for network parameter learning.

In the most closely related work, McAuley *et al.* [21] has proposed a CRF framework using social-network metadata to solve the image labeling problem. Compared with McAuley's approach, we have two advantages. First, our image-level CNN makes full use of the existing popular CNN models to extract powerful visual features from images, which integrates of the advantages of CNN feature extraction for nodes in the CRF. Second, rather than exploring the relational model based on high-frequency co-occurring words as tags, we exploit our text-level CNN and associative metadata to construct the fully connected CRF graph. The experiment section shows that our method results in significant performance improvement.

Our main contributions are summarized as follows:

- We propose a novel deep fully connected CRF framework DCRF that uses deep neural networks for image labeling with social network metadata. Deep CCN image features are fused with text features and network linkage information in an end-to-end deep learning formulation.

- Instead of using high-frequency words as tags, we propose to use a text-level CNN to exploit textual information. The fully connected CRF graph is built based on the features extracted from the text-level convolutional neural network, as well as the link-based metadata like user sets and image groups.

- For both learning and inference, we model a mean field approximation inference [16] for the fully connected CRF as an RNN, to introduce the CNN-RNN formulation. We also incorporate the weighted ranking loss to handle the imbalance label distribution existing in the training data.

- We evaluate the proposed DCRF on the MIR-9K dataset and achieve significantly improved performance compared to previous state-of-the-art approaches.

## 2. Related work

The related work can be divided into two categories: *social media for labeling* and *CRF with deep neural networks*.

**Social media context for labeling**. A set of tags associated with each image is commonly used in multimodal classification settings. Guillamumin *et al.* [7] explored the relationship between tags and manual annotations to recover annotations using a combination of tags and image content. Lindstaedt *et al.* [20] and Sigurbjornsson *et al.* [26] studied the problem of recommending tags that were obtained from similar images and similar users. Sawant *et al.* [24] and Stone *et al.* [29] investigates friendship information between users for tag recommendation in social networks. EXIF and GPS are two commonly used sources of metadata that come directly from the camera [22, 18, 14, 13]. Such metadata can be used to help determine who captured the photo and where, and also provide informative signals for image labeling tasks. Our method differs from all these and also [21] in that we use a much larger range of social media information, including free-form text as well as links, with deep learning based pixel descriptors incorporated into our novel deep learning fully connected CRF framework.

**CRF with deep neural networks**. In recent years, there are several works about CRF with a convolutional neural network which incorporate CRF to model structures in both output and hidden feature layers in CNN. Chu Chao *et al.* [5] propose a CRF-CNN framework which can simultaneously model structural information in both output

and hidden feature layers in a probabilistic way, and apply it to human pose estimation. Shuai Zheng *et al.* [34] introduce a new form of convolutional neural network that combines the strengths of CNN and CRF-based probabilistic graphical modeling. Zheng *et al.* [33] models conditional random fields for image segmentation task as recurrent neural networks, with the node features extracted from a convolutional neural network. Chandra *et al.* [2] propose a structured prediction model that endows the Deep Gaussian Conditional Random Field with a densely connected graph structure. Chen *et al.* [3] propose a similarity learning approach for person re-identification by combining the CRF model with deep neural networks. In these works, CNNs are integrated into CRF models and perform as feature extractors. Similarly, the two CNNs (image-level CNN and text-level CNN) in our proposed DCRF framework also work as powerful feature extractors in image labeling using social metadata. However, in addition to this, our text-level CNN is also used to help build the fully connected CRF graph, and both the learning and inference is under the united CNN-RNN framework, which distinguishes our proposed DCRF from the existing approaches.

# 3. Proposed approach

In this section, we will describe in detail the proposed Deep fully connected CRF framework (as illustrated in Figure 2) with deep neural networks.

## 3.1. CRF framework

Our probability framework is based on a fully connected conditional random field (CRF). This captures both unary dependencies between image labels, $Y = \{y_1, y_2, \ldots, y_N\}$ (with binary value indicating if the image has this class label, $y_n = 1$ , or not, $y_n = 0$), and the input features (*e.g.*, image features and metadata), as well as the pairwise dependencies between pairs of labels and the input features to produce the conditional probability $P(Y|I, M) = P(Y|\mathbf{x}, M)$, where $\mathbf{x}$ are the raw image features derived from the image set $I$. The labels are treated as binary hidden nodes in the CRF and the image features $\mathbf{x}$, and metadata $M$, are used in the observation nodes. Therefore, the conditional probability of the fully connected CRF can be defined as:

$$\begin{aligned} P(Y|I, M) &= P(Y|\mathbf{x}, M) \\ &= \frac{1}{Z}\exp(\sum_{i=1}^{N} A(y_i, \mathbf{x}_i) \\ &\quad + \sum_{i=1}^{N}\sum_{\forall j \neq i} B(y_i, y_j, M)), \end{aligned} \tag{1}$$

where $Z$ is the normalization constant that depends on $\mathbf{x}$ and $M$, while $A$ is the unary function based on the image information $\mathbf{x}$, and $B$ is the pairwise potential function based on the metadata $M$. The unary potentials are single image potentials, while the pairwise potentials are between pairs of images. For simplicity, a separate binary CRF model can be learned for each category.

## 3.2. Unary function with image-level convolutional neural network

The goal of the image-level convolutional neural network (CNN) is to extract feature vectors that are compact, representative, and can capture the most related visual information for the decoder. The rapid development of deep convolutional neural networks have had great success in large-scale image recognition task [9, 28], object detection [23, 4, 30] and visual captioning [31, 1, 32]. High-level features can be extracted from upper or intermediate layers of a deep CNN network. Therefore, a set of well-tested CNN networks can be used in our framework.

We use VGG-19 [27] and ResNet-152 network [9] for our framework. In this paper, for each category, we modify the original network by changing the number of outputs in the last layer from 1000 to 2 and fine-tune to conduct the binary classification. $\mathbf{x}_i$ in Equation 1 is the feature vector extracted from the second last fully connected layer (*i.e.*, the 18-th layer in VGG-19 and the 151-th layer in ResNet-152) for $i$-th instance. Then we can define the unary potential function as

$$A(y_i, \mathbf{x}_i) = \mathbf{w}_A^{y_i}\mathbf{x}_i + \mathbf{b}_A^{y_i}, \tag{2}$$

where $y_i$ is either 1 or 0, and $\mathbf{w}_A^{y_i}$, $\mathbf{b}_A^{y_i}$ are the parameters we need to learn.

## 3.3. Pairwise potential with text-level convolutional neural network and other meta information

Unlike previous work that tries to make full use of text information in the metadata by exploring the co-occurrence of high frequently used words as tags, we treat all the texts including title, description and comments information associated with an image as an unorganized incoherent sentence or a bag of words. Then we can train a text-level convolutional neural network to extract the feature vectors. In principle, any sentence convolutional neural networks can be used in our framework. To make it simple, we use Kim's sentence network [15], which is composed of one convolutional layer, one pooling layer, one dropout layer, one fully-connected layer and output with softmax activation function.

In this paper, we extract the 128-dimensional dropout layer to measure the similarity between any two images at the text-level. We define text similarity as

$$S_{text}(i, j) = \exp(-\frac{|\mathbf{x}_i^{text} - \mathbf{x}_j^{text}|^2}{2\theta_{text}}), \tag{3}$$

where the degree of nearness and similarity is controlled by the $\theta_{text}$ parameter.

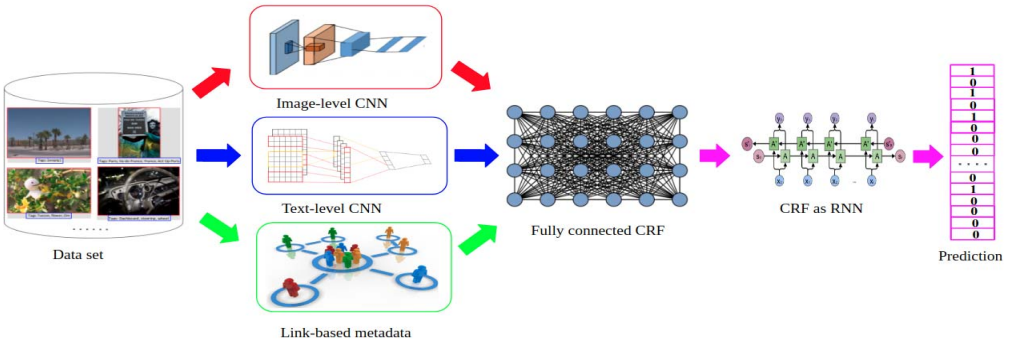



Figure 2: The pipeline of the proposed Deep fully connected CRF framework with deep neural networks for image labeling using social network metadata. The node features are extracted from an image-level convolutional neural network (CNN), and the edges are built based on the textual similarity based on the feature extracted from a text-level CNN, as well as the similarity determined by link-based metadata like user sets and image groups. For both learning and inference, we resort to a mean field approximation inference [16] for the CRF and model it as an RNN. We learn the parameter via a stochastic gradient descent RMSProp under the united CNN-RNN framework. Based on the learned CNN-RNN framework, we can predict the final 24-dimensional binary label vector directly at the testing stage.

Besides the text information, we also use the link-based metadata such as user sets and image groups [12]. A user set associates with a collection of images uploaded or collected by the same user. Image groups are community-curated, and are usually images belonging to the same concept, scene or event are uploaded and shared by the social network users. Both user sets and image groups have vocabularies, *i.e.*, $T_{set}$ and $T_{group}$, and each image $\mathbf{x}_i$ has two subsets, $t_i^{set}$ and $t_i^{group}$. We calculate the distance between any two nodes/images using the Jaccard similarity between their user sets and image groups as:

$$d(i,j,M_{set}) = 1 - |t_i^{set} \cap t_j^{set}|/|t_i^{set} \cup t_j^{set}|,$$

$$d(i,j,M_{group}) = 1 - |t_i^{group} \cap t_j^{group}|/|t_i^{group} \cup t_j^{group}|$$

and get the corresponding similarities

$$S_{set}(i,j) = \exp(-\frac{d(i,j,M_{set})^2}{2\theta_{set}}), \tag{4}$$

$$S_{group}(i,j) = \exp(-\frac{d(i,j,M_{group})^2}{2\theta_{group}}), \tag{5}$$

where both $\theta_{set}$ and $\theta_{group}$ are the parameter to control the degree of similarity.

Intuitively, two nodes/images which are more similar are more likely to share the same labels and should be able to affect each other more than others. Therefore, similarity is close related to the pairwise potential and we can define the pairwise potential as

$$B(y_i, y_j, M)) = \mu(y_i, y_j) \sum_{k \in \{text, set, group\}} \mathbf{w}_B^k S_k(i,j), \tag{6}$$

where $\mu(\cdot,\cdot)$ is label compatibility function, and $\mathbf{w}_B^{text}$, $\mathbf{w}_B^{set}$, $\mathbf{w}_B^{group}$ are the $2 \times 2$ parameters to be learned from the training data.

### 3.4. Learning and inference under neural network

Both image-level CNN and text-level CNN are trained separately. With the pre-trained networks, we are able to extract the node features and textual features to build the fully connected CRF, in which we also take user sets and image groups into account. Unfortunately, the parameters $\theta_{text}$, $\theta_{set}$ and $\theta_{group}$ cannot be calculated efficiently since their gradients involve a sum of non-Gaussian kernels, which are not amenable to the same acceleration techniques. Therefore, we resort to take grid search on a holdout validation set for determining all these three parameters, and learn the parameter $\mathbf{w} = [\mathbf{w}_A, \mathbf{w}_B]$, $\mathbf{b}_A$ and $\mu(\cdot,\cdot)$ only.

**Algorithm 1:** The outline of our proposed DCRF algorithm

**Input:** $I$ and $M$
**Output:** $Q$

$\mathbf{x} \leftarrow \text{CNN}_{image}(I)$
$\mathbf{x}^{text} \leftarrow \text{CNN}_{text}(M)$
$\mathbf{t}^{set}, \mathbf{t}^{group} \leftarrow M$
$U \leftarrow \mathbf{w}_A \mathbf{x} + \mathbf{b}_A$
$Q_i(y) \leftarrow \frac{1}{Z_i} \exp\{U_i(y)\}$
**while** *not converged* **do**
  $\tilde{Q}_i^{(k)}(y) \leftarrow \sum_{\forall j \neq i} S_k(i,j) Q_j(y)$ for all $k$
  $\check{Q}_i(y) \leftarrow \sum_k \mathbf{w}_B^k \tilde{Q}_i^{(k)}(y)$
  $\hat{Q}_i(y) \leftarrow \sum_{y'} \mu(y,y') \check{Q}_i(y)$
  $\breve{Q}_i(y) \leftarrow U_i(y) - \hat{Q}_i(y)$
  $Q_i(y) \leftarrow \frac{1}{Z_i} \exp\{\breve{Q}_i(y)\}$
**end**

We resort to a mean field approximation inference [16] which computes a distribution $Q(\mathbf{X})$ that minimizes the KL-divergence $D(Q||P)$ among all the approximated distributions $Q$ that can be expressed as a product of independent marginals, $Q(\mathbf{X}) = \prod_i Q_i$ [1] and

$$Q_i(y) = \frac{1}{Z_i} \exp\{-\mathbf{w}_A^y \mathbf{x}_i - Q_i'(y)\}, \tag{7}$$

where

$$Q_i'(y) = \sum_{y'} \mu(y, y') \sum_{k \in \{text, set, group\}} \mathbf{w}_B^k \sum_{\forall j \neq i} S_k(i, j) Q_j(y).$$

The update equation in Equation 7 leads to the inference steps, as seen in Algorithm 1. Inspired by the spirits in Zheng *et al.*'s work [33], we can implement the algorithm as a combination framework with both CNN and RNN. To be specific, line 1 and line 2 are associated with image-level CNN and text-level CNN, respectively. Line 3 can be regarded as a pre-processing step. Line 4 can be modeled as a fully connected layer. Line 5 is a softmax layer with unary potential as input. Line 7 can be regarded as linear combination of matrix multiplications, since the parameters $\theta_{text}$, $\theta_{set}$ and $\theta_{group}$ are determined by grid search validation and therefore $S_k(\cdot, \cdot)$ is fixed during running the algorithm 1. Line 8 can be implemented as a convolution with a 1x1 filter with three input channels and one output channel. Line 9 is another convolutional layer in which the number of both input and output channels are both two for the binary classification case. Line 10 is element-wise subtraction from the unary potential $U_i$. Line 11 is another softmax layer. Obviously, the layers associated with line 7 to line 11 construct a recurrent neural network (RNN).

Both learning and inference can be conducted under the united CNN-RNN framework which we implement in PyTorch. For the loss function, besides the L2 regularization on $\mathbf{w}_A$, we use weighted binary cross entropy and pairwise ranking loss

$$\begin{aligned} Loss(Q, Y) = & \sum_{i=1}^{N} -\frac{y_i}{N_+} \log Q_i(y_i) - \frac{1 - y_i}{N_-} \log Q_i(1 - y_i) \\ & + \sum_{i=1}^{N} \frac{y_i}{N_+} (1 - (Q_i(y_i) - Q_i(1 - y_i)) \\ & + \sum_{i=1}^{N} \frac{1 - y_i}{N_-} (1 - (Q_i(1 - y_i) - Q_i(y_i)) + \lambda \|\mathbf{w}_A\|_2, \end{aligned} \tag{8}$$

to handle the possible imbalance distribution of positive/negative instances in the training data and ensure a good probability ranking. Note that $N_+$ and $N_-$ in Equation 8 are the number of positive training instances and the number of the negative training instances, respectively. $\lambda$ is the regularization parameter and we set 0.1 for VGG-19 and 0.001 for ResNet-152. We initialize all parameters using the method of [8] and optimize using stochastic gradient descent RMSProp with a fixed learning rate of 0.1.

[1] For simplicity, we use $Q_i$ to indicate $Q(\mathbf{x}_i)$ and therefore $Q_i(y)$ indicate the probability of $\mathbf{x}_i$ being labeled as label $y$.

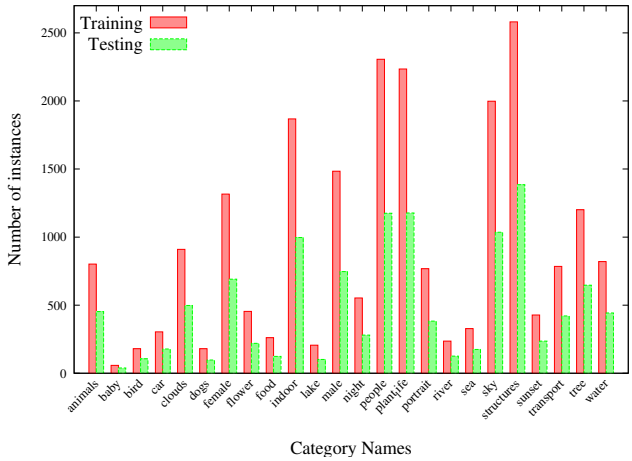



Figure 3: The number of instances per category used for both training and testing among 24 categories on the full MIR-9K dataset.

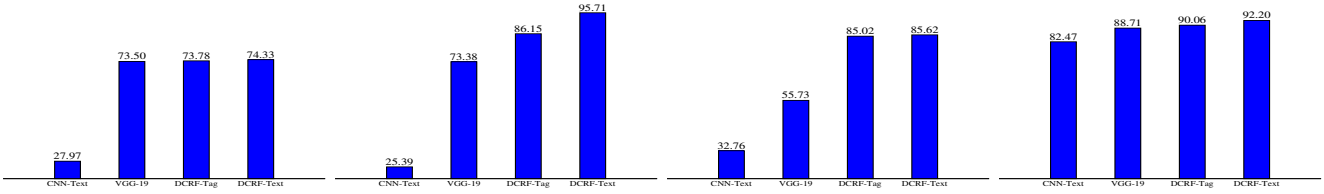
## 4. Experiment

We conduct experiments to verify the effectiveness of the proposed approach on the MIR-9K dataset, a subset of the MIRFLICKR [11] dataset which is available under Creative Commons licenses. It worths mentioning here that we do not evaluate on datasets like NSU-WIDE dataset [25] since there are only tag words available on the official website, without the original text information (*e.g.*, image title, description and comments), which prevent us from evaluating our text-level CNN in Section 3.3. The MIR-9K dataset contains 6000 training instances and 3182 testing instances with 24 categories: animals, baby, bird, car, clouds, dogs, female, flower, food, indoor, lake, male, night, people, plant life, portrait, river, sea, sky, structures, sunset, transport, tree, and water. It involves a set of 3,213 users, a collection of 34,942 words and 17,687 image groups. The data distribution is shown in Figure 3, where there are imbalance issues among different categories.

For measurement metrics, we report the average precision (AP), recall, precision and accuracy over all 24 categories for the sake of comparison with published algorithms.

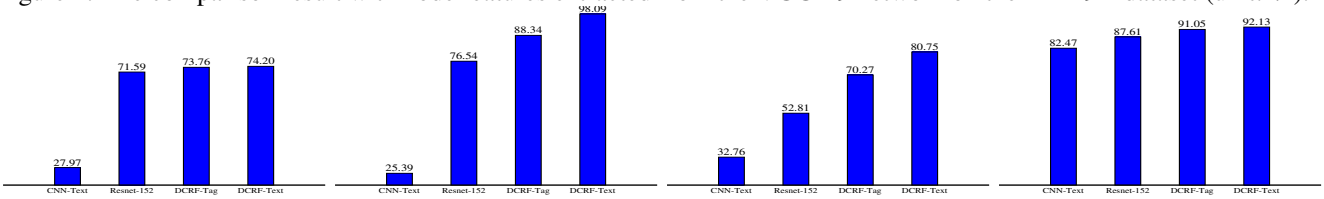
### 4.1. Effectiveness of the text-level CNN

To evaluate the effectiveness of the text-level CNN to our proposed DCRF, we first define the CRF only with the textual similarity defined in Equation 3, and denote the method




(a) AP (b) Recall (c) Precision (d) Accuracy

Figure 4: The comparison result with node features extracted from the VGG-19 network on the MIR-9K dataset (unit: %).




(a) AP (b) Recall (c) Precision (d) Accuracy

Figure 5: The comparison result with node features extracted from the ResNet-152 network on the MIR-9K dataset (unit: %).

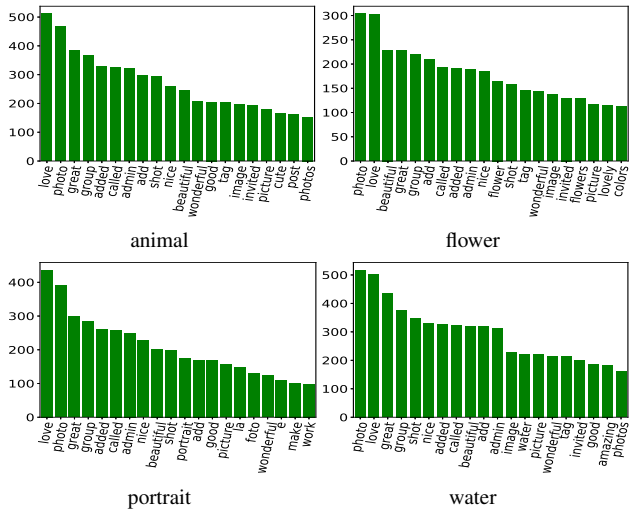



Figure 6: Visualization of top 20 tag words appearing in 4 categories on the positive training instances of the MIR-9K dataset.

as DCRF-Text. We then replace the textual similarity with the tag Jaccard similarity calculated in [12] (with 5000 high frequent occurring words as tags) for any two nodes/images to build the CRF, and mark the competing algorithm as DCRF-Tag. Note that both DCRF-Text and DCRF-Tag use the same visual node features obtained from the VGG-19 or ResNet-152 network. We also provide two baselines that use text information and image information independently, *i.e.*, CNN-Text, VGG-19, and ResNet-152, respectively.

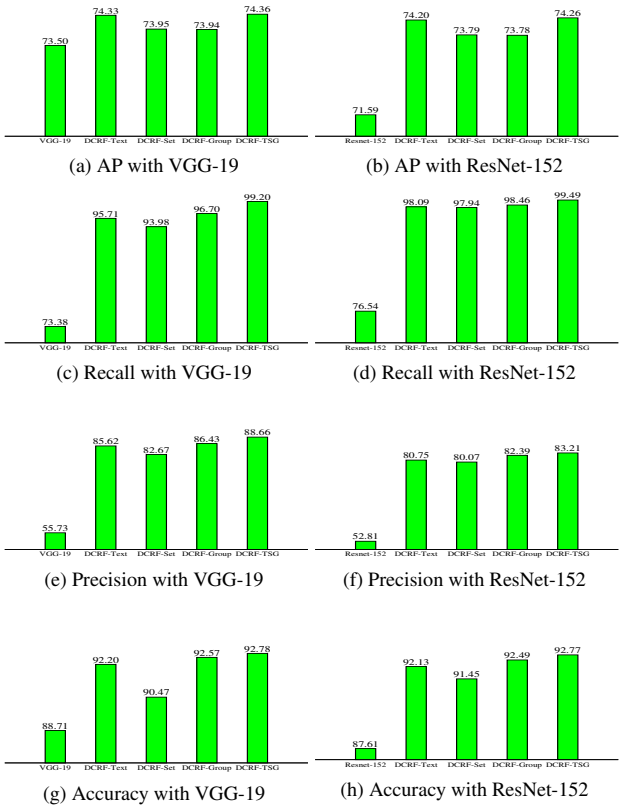
The performance results are summarized in Figure 4 and 5. As we can see, (a) both VGG-19 and ResNet-152 perform much better than CNN-Text in AP, recall, precision, and accuracy, which indicates image information is more helpful for image labeling when compared with the text information. (b) Also, both DCRF-Text and DCRF-Tag work better than CNN-Text, VGG-19 and ResNet-152 in all four metrics, which indicates the text information complementary to image information and useful for improving the labeling accuracy. (c) Regardless of using the VGG-19 or ResNet-152 network to extract the node/image features, DCRF-Text outperforms DCRF-Tag, which clearly demonstrate that the text-level CNN of our DCRF is better able to explore the underlying information in text than just relying on the top frequent words as tags.

To analyze why our proposed DCRF-Text is able to outperform DCRF-Tag, we visualize the top 20 words in decreasing order of frequency occurring among 6000 training examples. Due to space limitations, we only present 4 categories in Figure 6. For more information, please refer to our supplementary. We observe that the top frequently co-occurring words such as "love", "photo", "great", "group", "added", "nice" *et. al* convey little information relative to any of the prediction 24 categories. Instead, we resort to a text-level CNN to explore the underlying information and use the textual similarity based on the features extracted from the text-level CNN to build the fully connected CRF graph, which explains why our proposed approach DCRF makes slightly better use of text information.

### 4.2. Effectiveness of the metadata for image labeling

In Section 4.1, text information has been proved to able to boost the performance for the image labeling accuracy. We also want to see whether the link-based metadata such as user sets and image groups, can produce positive effect on the image labeling. Using the same node features extracted from either the VGG-19 or ResNet-152 network, we first define the CRF graph with a single type of metadata (*i.e.*, text, user sets and image groups) and get three versions of DCRF: DCRF-Text, DCRF-Set and DCRF-Group. Then we define the CRF with the combined these three types of metadata together, and denote the combined version as

DCRF-TSG.



(a) AP with VGG-19 (b) AP with ResNet-152

(c) Recall with VGG-19 (d) Recall with ResNet-152

(e) Precision with VGG-19 (f) Precision with ResNet-152

(g) Accuracy with VGG-19 (h) Accuracy with ResNet-152

Figure 7: The comparison result with node features extracted from the VGG-19 and ResNet-152 networks (unit: %).

We summarize the results in Figure 7. As we expect, each of all these DCRF-Text, DCRF-Set and DCRF-Group perform better than VGG-19 and ResNet-152, which suggests that all these three types of metadata are helpful for image annotation. Among these three types of metadata, using text information provides the greatest improvement in AP compared to the other two. However, combining all three types into the DCRF-TSG model produces the greatest performance in all four metrics, regardless of using the VGG-19 or ResNet-152 network as node feature extractor. Such observations demonstrate that the metadata as text, user sets, and image groups are complementary to each other and can be used for boosting the quality of image labeling.

### 4.3. Compare with state-of-the-art approach

In addition to the text-level CNN model $\text{CNN}_{text}$ [15], and four image-level popular CNN models $\text{AlexNet}_{img}$ [17], $\text{VGG-19}_{img}$ [27], $\text{ResNet-152}_{img}$ [9] and $\text{DenseNet-201}_{img}$ [10], we compare our proposed DCRF with the most closely related work, *i.e.*, McAuley *et al*'s CRF algorithm [21], denoted as McAuley-CRF, which explores the social-network metadata such as image groups and comments, and utilizes structured learning techniques to learn model parameters. We also compare with one deep learning related work, *i.e.*, Johnson *et al*'s neighbor-based CNN algorithm [12], denoted as Johnson-NCNN, which use image metadata in a nonparametric manner to generate neighborhoods of related images using Jaccard similarity and then uses a deep learning to blend visual information from the image and its neighbors.

Table 1: The performance comparison among the competing algorithms (AP: average precision, REC: recall, PRE: precision, ACC: accuracy, unit: %).

| | AP | REC | PRE | ACC |
|---|---|---|---|---|
| $\text{CNN}_{text}$ [15] | 27.97 | 25.39 | 32.76 | 82.47 |
| $\text{AlexNet}_{img}$ [17] | 62.54 | 76.30 | 40.25 | 74.56 |
| $\text{VGG-19}_{img}$ [27] | **73.50** | **77.38** | **55.73** | **88.71** |
| $\text{ResNet-152}_{img}$ [9] | 71.59 | 76.54 | 52.82 | 87.62 |
| $\text{DenseNet-201}_{img}$ [10] | 63.26 | 72.55 | 42.93 | 85.06 |
| McAuley-CRF [21] | 54.73 | 40.75 | 59.44 | 83.1 |
| John-$\text{NCNN}_{vgg}$ [12] | **73.78** | **61.18** | 79.01 | **92.57** |
| John-$\text{NCNN}_{res}$ [12] | 72.90 | 50.59 | **81.39** | 91.87 |
| $\text{DCRF}_{vgg}$-BCE | 74.13 | 92.66 | 85.86 | 92.50 |
| $\text{DCRF}_{vgg}$-RLoss | 74.29 | 93.12 | 88.18 | 92.61 |
| $\text{DCRF}_{vgg}$-BCE+RLoss | **74.36** | 99.20 | **88.66** | **92.78** |
| $\text{DCRF}_{res}$-BCE | 74.05 | 91.52 | 74.69 | 91.74 |
| $\text{DCRF}_{res}$-RLoss | 74.09 | 94.38 | 77.59 | 91.93 |
| $\text{DCRF}_{res}$-BCE+RLoss | 74.26 | **99.49** | 83.21 | 92.77 |

For fair comparison, we provide two versions of deep learning models (*i.e.*, the VGG-19 and ResNet-152 networks) for Johnson-NCNN, and mark them as Johnson-$\text{NCNN}_{vgg}$ and Johnson-$\text{NCNN}_{res}$, respectively. Obviously, our DCRF has two versions, *i.e.*, $\text{DCRF}_{vgg}$ and $\text{DCRF}_{res}$. To better show the effectiveness of the loss function we use in Section 3.4, we get six versions, *i.e.*, $\text{DCRF}_{vgg}$-BCE, $\text{DCRF}_{res}$-BCE, $\text{DCRF}_{vgg}$-RLoss, $\text{DCRF}_{res}$-RLoss, $\text{DCRF}_{vgg}$-BCE+RLoss, $\text{DCRF}_{res}$-BCE+RLoss, in which "BCE" indicates binary cross entropy and "RLoss" means pairwise ranking loss in the binary classification cases.

The results are summarized in Table 1, from which we can observe: (a) all four image-level CNNs perform better $\text{CNN}_{text}$, and VGG-19 and ResNet-152 are the top 2 image-level CNN models on the MIR-9K dataset; (b) our proposed DCRF significantly outperforms the McAuley-CRF approach, which shows the big advantages of using deep neural networks in CRF; (c) with the same deep node features extracted from either the VGG-19 or ResNet-152 network, all versions of our DCRF are able to obtain improvement in all four metrics when compared to using tex-

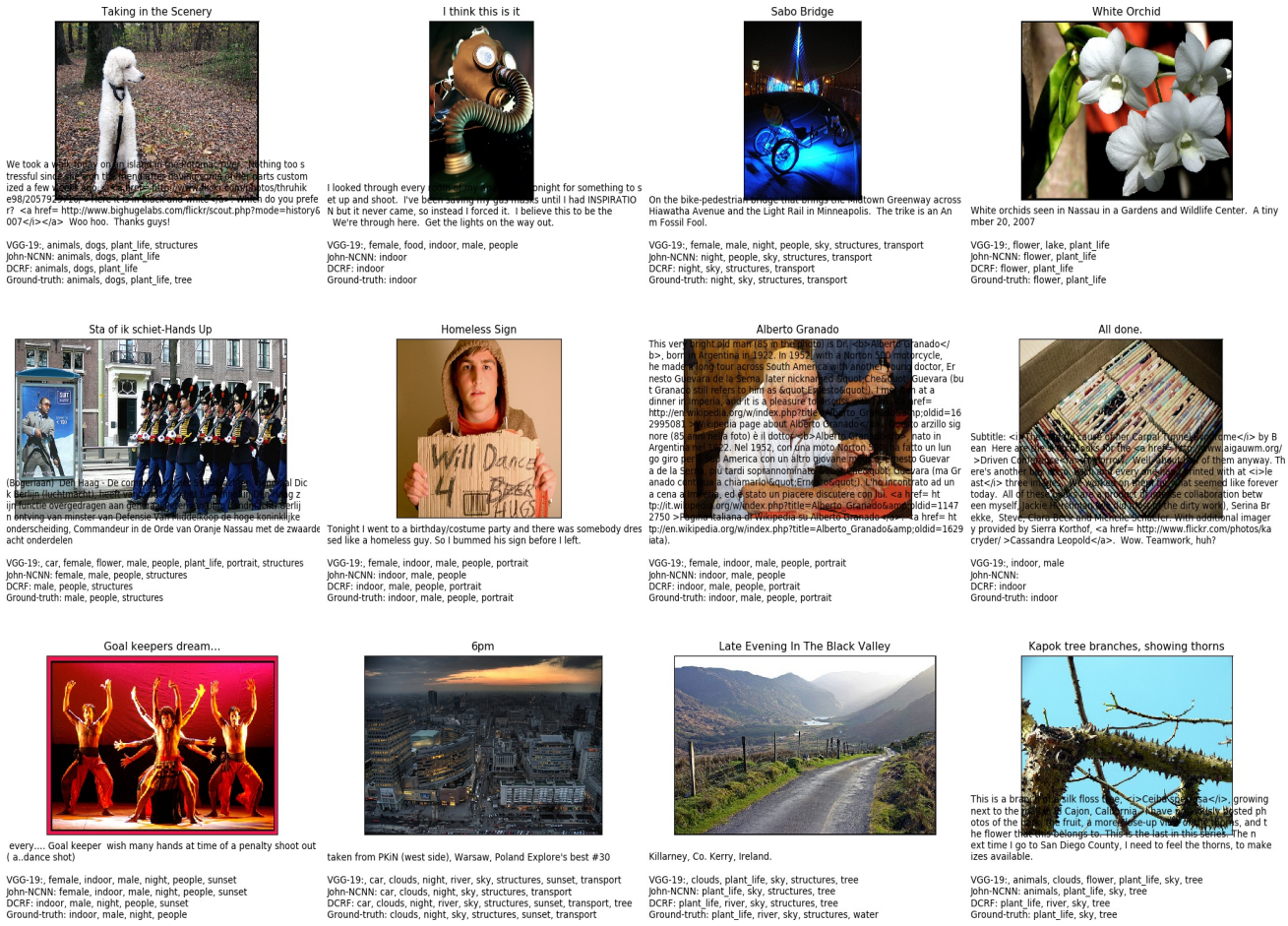



Figure 8: Visualization of image labeling on some testing examples. For each example, the title is above the image, and the text below the image is the corresponding description. The bottom four rows are prediction labels by VGG-19, John-NCNN$_{vgg}$, DCFR$_{vgg}$ and the corresponding ground-truth labels.

tual or visual information only; (d) rank loss function works a little better than cross entropy loss function in all four metrics; (e) the performance of our DCRF$_{vgg}$-BCE+RLoss and DCRF$_{res}$-BCE+RLoss are slightly higher than those of John-NCNN$_{vgg}$ and John-NCNN$_{res}$ in AP and accuracy and significantly higher in recall and precision, and our DCRF$_{vgg}$-BCE+RLoss achieves the best performances in AP, precision and accuracy.

### 4.4. Visualization

For better understanding of our porposed DCRF, we visualize some testing examples with DCRF$_{vgg}$ and take VGG-19 and John-NCNN$_{vgg}$ as baselines in Figure 8. As we can see, both DCRF$_{vgg}$ and John-NCNN$_{vgg}$ benefit from the metadata information for improving the quality of image labeling. Overall, our proposed DCRF achieves the higher quality of labels. Moreover, our proposed DCRF$_{vgg}$ is able to predict some categories that are not clear or even occluded in images, such as "car" and "tree" in the middle two examples at the bottom row.

## 5. Conclusion

In this paper, we propose a novel deep fully connected CRF based framework DCRF with deep neural networks for image labeling using social network metadata. In such a framework, CNNs are used to extract powerful visual features for nodes/images and textual features to explore the underlying information embedded in text. The fully connected CRF graph is built based on the textual similarity and the link-based metadata like user sets and image groups. With the mean-field approximation modeled as an RNN, our proposed framework DCRF becomes a joint end-to-end CNN-RNN formulation, which combines the strengths of both CNNs and RNNs. The experimental evaluation on the MIR-9K dataset demonstrates that our proposed DCRF framework outperforms state-of-the-art approaches [21, 12]. Our future work includes investigating more effective meta information, and improving the efficiency of the current DCRF framework to handle more complicated real-world application problems.

## References


[1] J. Aneja et al. Convolutional image captioning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[2] S. Chandra, N. Usunier, and I. Kokkinos. Dense and low-rank gaussian crfs using deep embeddings. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017.

[3] D. Chen et al. Group consistent similarity learning via deep crf for person re-identification. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[4] Y. Chen et al. Domain adaptive faster r-cnn for object detection in the wild. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[5] X. Chu et al. CRF-CNN: modeling structured information in human pose estimation. In *Advances in Neural Information Processing Systems (NIPS)*, 2016.

[6] J. Deng et al. ImageNet: A Large-Scale Hierarchical Image Database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[7] M. Guillaumin et al. Multimodal semi-supervised learning for image classification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[8] K. He et al. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[9] K. He et al. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[10] G. Huang et al. Densely connected convolutional networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2017.

[11] M. J. Huiskes and M. S. Lew. The mir flickr retrieval evaluation. In *ACM International Conference on Multimedia Information Retrieval (MIR)*, 2008.

[12] J. Johnson et al. Love thy neighbors: Image annotation by exploiting image metadata. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[13] D. Joshi et al. Using geotags to derive rich tag-clouds for image annotation. In *Social Media Modeling and Computing*. 2011.

[14] E. Kalogerakis et al. Image sequence geolocation with human travel priors. In *IEEE International Conference on Computer Vision (ICCV)*, 2009.

[15] Y. Kim. Convolutional neural networks for sentence classification. *arXiv*, 2014.

[16] P. Krähenbühl and V. Koltun. Efficient inference in fully connected crfs with gaussian edge potentials. In *Advances in neural information processing systems (NIPS)*, 2011.

[17] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira et al., editors, *Advances in Neural Information Processing Systems (NIPS)*. 2012.

[18] Y. Li et al. Landmark classification in large-scale image collections. In *ICCV*, pages 1957–1964, 2009.

[19] T. Lin et al. Microsoft COCO: common objects in context. In *The European Conference on Computer Vision (ECCV)*, 2014.

[20] S. Lindstaedt et al. Recommending tags for pictures based on text, visual content and user context. In *International Conference on Internet and Web Applicants and Services (ICIW)*, 2008.

[21] J. J. McAuley and J. Leskovec. Image labeling on a network: Using social-network metadata for image classification. In *The European Conference on Computer Vision (ECCV)*, 2012.

[22] M. M. M. Patil et al. Survey on image based information retrieval using geo-tagging. 2017.

[23] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems (NIPS)*. 2015.

[24] N. Sawant, R. Datta, J. Li, and J. Z. Wang. Quest for relevant tags using local interaction networks and visual content. In *The ACM SIGMM International Conference on Multimedia Information Retrieval (MIR)*, 2010.

[25] T. seng Chua et al. Nus-wide: A real-world web image database from national university of singapore. In *The International Conference on Image and Video Retrieval (CIVR)*, 2009.

[26] B. Sigurbjrnsson and R. van Zwol. Flickr tag recommendation based on collective knowledge. In *The International Conference on World Wide Web (WWW)*, 2008.

[27] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv*, 2014.

[28] P. Stock and M. Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *The European Conference on Computer Vision (ECCV)*, 2018.

[29] Z. Stone, T. Zickler, and T. Darrell. Autotagging facebook: Social network context improves photo annotation. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW) on Internet Vision*, 2008.

[30] P. Tang et al. Weakly supervised region proposal network and object detection. In *The European Conference on Computer Vision (ECCV)*, 2018.

[31] S. Venugopalan et al. Translating videos to natural language using deep recurrent neural networks. In *North American Chapter of the Association for Computational Linguistics Human Language Technologies (NAACL-HLT)*, 2015.

[32] H. Yu et al. Fine-grained video captioning for sports narrative. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[33] S. Zheng et al. Conditional random fields as recurrent neural networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.

[34] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *IEEE International Conference on Computer Vision (ICCV)*, 2015.