Multi-view, Multi-label Learning with Deep Neural Networks

Research Proficiency Exam
September 7, 2016

ZIQIAO GUAN
DEPT. OF COMPUTER SCIENCE
STONY BROOK UNIVERSITY
Overview

Introduction

Deep Neural Network
  ◦ Artificial Neural Network
  ◦ Deep Belief Network
  ◦ Convolutional Neural Network
  ◦ Recurrent Neural Network

Multi-view and Multi-label Learning
  ◦ Multi-view Learning
  ◦ Multi-label Learning

Applications

Conclusion
What is Deep Learning?

**Multi-layer** representation methods to learn features **from data**

SIFT Descriptor [Lowe 1999]

AlexNet and the learned filters [Krizhevsky 2012]


Why We Care About Structure?

Deep models take a long time to train and are prone to local minima

Incorporate higher level understanding

Training a ConvNet

Multi-label correlations [Che 2015]

Z. Che et al. Deep Computational Phenotyping. In KDD '15.
Overview

Introduction

**Deep Neural Network**
- Artificial Neural Network
- Deep Belief Network
- Convolutional Neural Network
- Recurrent Neural Network

Multi-view and Multi-label Learning
- Multi-view Learning
- Multi-label Learning

Applications

Conclusion
Artificial Neural Network

Input  Hidden  Output

Sigmoid \( \sigma(x) = \frac{1}{1+e^{-x}} \)
Backpropagation

Forward pass:
\[ x \rightarrow g(x) \rightarrow f(g(x)) \rightarrow F(x) \]

Backward pass:
\[ l \rightarrow f'(g(x)) \rightarrow f'(g(x))g'(x) \rightarrow F'(x) \]

Backpropagation

Extend the network to an **Error node E**.

**Gradient descent:**

\[ \Delta w_i = -\gamma \frac{\partial E}{\partial w_i}, \]

where

\[ \frac{\partial E}{\partial w_i} = \frac{\partial f(w_i o_i, \cdot)}{\partial w_i} = o_i \frac{\partial f}{\partial w_i} \triangleq o_i \delta_i, \]

\( \delta_i \) is computed via BP.
Problem with BP

Requires a large amount of labeled data
Very slow in multi-layer networks
Tends to get stuck in local minima

... and support vector machines worked better

Support vector machine and the kernel trick [Cortes 1995]

Deep Belief Network

Training DBN

Unsupervised greedy layer-wise training
- Start from the bottom and treat the adjacent two layers as RBM
- Maximize the probability of training samples
- Go up the DBN and repeat

*Supervised training (fine-tuning)

Training Deep Auto-encoders

Train the encoder side as stacked RBMs

“Unroll”

Fine-tune with backpropagation

Convolutional Neural Network

AlexNet [Krizhevsky 2012]

ConvNet Operations

Convolutional layer
Nonlinear activation
Pooling layer
Convergence and Overfitting

Speedup convergence with normalizations

Prevent overfitting with dropout


Batch Normalization

Input whitening improves training speed
- Zero means, unit variations
- But intermediate layers still undergo covariate shift

Perform “whitening” at the intermediate layers
- The network needs to account for the normalization step

Batch Normalization [Ioffe 2015]
- Normalize per mini-batch $\hat{x} = \frac{x - E[x]}{\sqrt{\text{Var}[x]}}$
- Scale and shift the normalized value with learned parameters to preserve representation power $y = \gamma \hat{x} + \beta$

---

Notable Architectures

AlexNet [Krizhevsky 2012]
VGGNet [Simonyan 2015]
GoogLeNet [Szegedy 2015]
ResNet [He 2016]

AlexNet

Stacked convolution layers
ReLU and dropout
GPU training

VGGNet

Depth is the key (16/19 layers)

Homogeneous structure with 3x3 convs and 2x2 pooling

Showed fully connected layers could be removed

GoogLeNet

Introduces the Inception module as a computation unit on a sliding window
- Inspired by “Network in Network” [Lin 2014]
- Neurons that fire together, wire together

ResNet

Super deep ConvNets
  ◦ 152-layer on ImageNet and 1000-layer on CIFAR-10

Features shortcut connections that pass identity mappings across layers
  ◦ Observation: Deeper networks have higher training errors
  ◦ Assumption: Nonlinear layers have difficulties approximating identity mappings
  ◦ Solution: Directly feed the identity downstream and learn the residue

Heavy use of batch normalization

Recurrent Neural Network
RNN Variants

Bidirectional RNN [Schuster 1997]  
Deep Bidirectional RNN [Amodei 2016]


Backpropagation Through Time

BPTT for $\mathbf{W}$:

$$\frac{\partial E_n}{\partial \mathbf{W}} = \frac{\partial E_n}{\partial y_n} \frac{\partial y_n}{\partial s_n} \frac{\partial s_n}{\partial \mathbf{W}}$$

$$\Rightarrow$$

$$\frac{\partial E_n}{\partial \mathbf{W}} = \sum_{k=0}^{n} \frac{\partial E_n}{\partial y_n} \frac{\partial y_n}{\partial s_n} \frac{\partial s_n}{\partial s_k} \frac{\partial s_k}{\partial \mathbf{W}}$$

The vanishing gradient problem

Understanding LSTM Networks. URL http://colah.github.io/posts/2015-08-Understanding-LSTMs/.
Gated Recurrent Unit

Understanding LSTM Networks. URL http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Overview

Introduction

Deep Neural Network
◦ Artificial Neural Network
◦ Deep Belief Network
◦ Convolutional Neural Network
◦ Recurrent Neural Network

Multi-view and Multi-label Learning
◦ Multi-view Learning
◦ Multi-label Learning

Applications

Conclusion
“Multi-” Objective Learning

Multi-view Learning (MVL) or multimodal learning
- Learning from data represented by multiple distinct datasets

Multi-instance Learning (MIL)
- Learning from data in bags that obtain their labels from bag instances

Multi-label Learning (MLL)
- Learning from data with multiple non-mutually-exclusive labels (attributes)

Multi-task Learning (MTL)
- Learning data representations for multiple related tasks

“Multi-” Objective Learning

MVL  MIL

MLL  MTL
“Multi-” Objective Learning

**Multi-view Learning (MVL)** or multimodal learning
- Learning from data represented by multiple distinct datasets

**Multi-instance Learning (MIL)**
- Learning from data in bags that obtain their labels from bag instances

**Multi-label Learning (MLL)**
- Learning from data with multiple non-mutually-exclusive labels (attributes)

**Multi-task Learning (MTL)**
- Learning data representations for multiple related tasks
Deep MVL

Deep multi-view representation learning
- Learn a common low dimensional feature space

Network co-training
- Hybrid network for joint inference
Split Auto-encoder

\[
\min_{W_f, W_g, w_q} \frac{1}{N} \sum_{i=1}^{N} (\|x_i - p(f(x_i))\|^2 + \|y_i - q(f(x_i))\|^2)
\]

[Ngiam 2011]

Deep Canonical Correlation Analysis

Canonical Correlation Analysis. Let \((X_1, X_2) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}\) denote a pair of random vectors, CCA finds pairs of linear projections of the two views \((w_1', w_2')\) that are maximally correlated:

\[
(w_1^*, w_2^*) = \arg\max_{w_1', w_2'} \text{corr}(w_1'X_1, w_2'X_2)
\]

DCCA. Perform CCA on \((f(X_1), g(X_2))\) with DNNs \(f, g\)

---


DCCAE

[Wang 2015]

Heterogeneous Network Embedding

Learn from links across heterogeneous networks
- Image-text, image-image, text-text

Project deep representations of different views into common space
- $p'(X) = U p(X)$
- $q'(z) = V q(z)$

Minimize the loss in common feature space
- $\min \frac{1}{N_{II}} \sum L(p'(X_i), p'(X_j)) + \frac{\lambda_1}{N_{TT}} \sum L(q'(z_i), q'(z_j)) + \frac{\lambda_2}{N_{IT}} \sum L(p'(X_i), q'(z_j))$

[Chang 2015]

Network Co-training

Joint Top Layers [Ha 2016]  

Network Cascading [Vinyals 2015]

Joint Top Layers

DMVH [Kang 2012]  
MV-DNN [Elkahky 2015]  
HMM-FNN [Wu 2015]

Network Cascading

NIC [Vinyals 2015]  m-RNN [Mao 2015]  [Kiros 2015]

J. Mao et al. Deep Captioning with Multimodal Recurrent Neural Networks (m-RNN). In ICLR 2015.
Deep MLL

Early model: BP-MLL [Zhang 2006]
- Fully connected network
- N-way multi-label output with loss function
  \[ E_i = \frac{1}{|C_+||C_-|} \sum_{(k,l) \in C_+ \times C_-} \exp(- (c_k^i - c_l^i)) \]

With deep learning practice [Nam 2014]
- Cross entropy instead of the exponential rank loss
- ReLU instead of hyperbolic tangent
- Dropout
- AdaGrad [Duchi 2011]

Multi-label Ranking Loss

Softmax loss
- Compute softmax posterior probability \( p_l = \frac{c_l}{\sum_k c_k} \)
- Compute KL-divergence between prob. and ground truth \( E = \frac{1}{n} D_{KL}(p||y) \)

Pairwise ranking loss \( E = \sum_i \sum_{j \in C_i^+} \sum_{k \in C_i^-} \max(0,1 - c_j^i + c_k^i) \)
- Does not directly optimize top-k accuracy

Weighted approximate ranking \( E = \sum_i \sum_{j \in C_i^+} \sum_{k \in C_i^-} L(r_j) \max(0,1 - c_j^i + c_k^i) \)

Multi-label Graph Regularization

Label correlation

Multi-label Network

Z. Che et al. Deep Computational Phenotyping. In KDD '15.
Multi-label Graph Regularization

Given similarity matrix $A$, $C = \text{diag}((\sum_j A_{ij})_i)$, Laplacian $L = C - A$

$$\text{tr}(\beta^T L \beta) = \frac{1}{2} \sum_{ij} A_{ij} \|\beta_i - \beta_j\|^2$$

Laplacian regularization

$$E = \sum_i E_i + \lambda R(\Theta) + \text{tr}(\beta^T L \beta)$$

Z. Che et al. Deep Computational Phenotyping. In KDD '15.
Weakly Supervised Learning

Weak supervision: obtaining the complete label set is prohibitive [Shankar 2015]

Weakly Supervised Learning

Pseudo-label for semi-supervised learning [Lee 2013]
- Pick the class with maximum predicted prob. for unlabeled samples
- Increase the weight of pseudo-label loss during training
- Equivalent to entropy regularization

Deep-carving [Shankar 2015]
- Compute per-class average spatial response at each filter
- Compute a similarity score for each image vs. each class among all filters as pseudo-label
- Update pseudo-labels every few epochs

Regional Object Detection

Localization + Classification
R-CNN

R-CNN: Regions with CNN features

1. Input image
2. Extract region proposals (~2k)
3. Compute CNN features
4. Classify regions

[Girshick 2014]

Fast R-CNN

Faster R-CNN


[Ren 2015]
Hypothesis-CNN-Pooling

Overview

Introduction

Deep Neural Network
  ◦ Artificial Neural Network
  ◦ Deep Belief Network
  ◦ Convolutional Neural Network
  ◦ Recurrent Neural Network

Multi-view and Multi-label Learning
  ◦ Multi-view Learning
  ◦ Multi-label Learning

Applications

Conclusion
Applications

E-commerce item categorization [Ha 2016]
Deep semantic ranking based hashing [Zhao 2015]
Dense image captioning [Johnson 2016]
Preliminary work: X-ray scattering image classification
E-commerce Item Categorization

**Problem:** classify newly registered items on e-commerce websites based on manual metadata

**Input:** \{item name, brand name, high-level category, shopping mall ID, manufacturer, image\}

**Output:** category label

**Challenge:** Noisy data, long-tail small categories

E-commerce Item Categorization

**Model:** multiple-RNN with joint fully connected layers

---

Deep Semantic Ranking Based Hashing

**Problem:** Learn a set of hash functions to encode multi-label image similarity

Hash function $h: \mathbb{R}^D \rightarrow \{-1,1\}$,

Learn $h(x) = [h_1(x), h_2(x), ..., h_K(x)]$ s.t. $|h(x)-h(y)| \sim |L(x)-L(y)|$, $L(.)$ is the multi-label vector.
Deep Semantic Ranking Based Hashing

\[ h(x; w) = \text{sign}(w^T [f_a(x); f_b(x)]) \]

Dense Image Captioning

Dense Image Captioning

**X-ray Scattering Image Classification**

**Problem:** predict the attributes of x-ray scattering images

- Explore the advantage brought by going deep
- Understand the difference between x-ray images and natural images
- Exploit the multi-learning potential

X-ray Classify: Multi-Learning Potential

Multi-label Correlation

Image and q-integral curve
X-ray Classify: Method

Use simulation software to generate synthetic image data for training
15 major attributes (synthetic), 9 in experimental data
Log image in AlexNet with cross entropy loss
X-ray Classify: Results

Table 4.1: Mean Average Precision on Synthetic Dataset

<table>
<thead>
<tr>
<th>Category</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diffuse low-q</td>
<td>0.7823</td>
</tr>
<tr>
<td>Diffuse high-q</td>
<td>0.7624</td>
</tr>
<tr>
<td>Halo</td>
<td>0.7718</td>
</tr>
<tr>
<td>Higher orders</td>
<td>0.8943</td>
</tr>
<tr>
<td>Rings</td>
<td>0.8926</td>
</tr>
<tr>
<td>BCC</td>
<td>0.0186</td>
</tr>
<tr>
<td>FCC</td>
<td>0.0750</td>
</tr>
<tr>
<td>Hexagonal</td>
<td>0.1821</td>
</tr>
<tr>
<td>Lamellar</td>
<td>0.2725</td>
</tr>
<tr>
<td>Symmetry halo</td>
<td>0.4758</td>
</tr>
<tr>
<td>Symmetry ring</td>
<td>0.4055</td>
</tr>
<tr>
<td>Circular beamstop</td>
<td>0.3016</td>
</tr>
<tr>
<td>Wedge beamstop</td>
<td>0.6206</td>
</tr>
<tr>
<td>Linear beamstop</td>
<td>0.6402</td>
</tr>
<tr>
<td>Beam off image</td>
<td>0.8273</td>
</tr>
</tbody>
</table>

Table 4.2: Mean Average Precision on Experimental Dataset

<table>
<thead>
<tr>
<th>Category</th>
<th>mAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Diffuse low-q</td>
<td>0.5545</td>
</tr>
<tr>
<td>Diffuse high-q</td>
<td>0.1425</td>
</tr>
<tr>
<td>Halo</td>
<td>0.2414</td>
</tr>
<tr>
<td>Higher orders</td>
<td>0.6365</td>
</tr>
<tr>
<td>Rings</td>
<td>0.7485</td>
</tr>
<tr>
<td>Symmetry ring</td>
<td>0.0119</td>
</tr>
<tr>
<td>Circular beamstop</td>
<td>0.5649</td>
</tr>
<tr>
<td>Linear beamstop</td>
<td>0.3325</td>
</tr>
<tr>
<td>Beam off image</td>
<td>0.8750</td>
</tr>
</tbody>
</table>
X-ray Classify: Future Directions

Hierarchical classification
  - Major visual pattern vs. style variation
  - Visual vs. physical meaning

Transform and 1D information

Multi-scale classification
Conclusion

Deep neural networks have demonstrated great versatility with heavy modifications for multi-learning problems
  ◦ End-to-end models work well, are easy to train and able to incorporate problem intuitions

Understanding of the data still counts; but it is not a comeback to hand crafted features
  ◦ Deep multi-learning is one step further ahead from generic deep models

Future work directions
  ◦ New features and structures for MVL
  ◦ New computation units and layers
  ◦ Manifold learning for structure-aware MLL
Thank you for your attention

Q&A