Probabilistic Graphical Models

A Brief Overview of Trustworthy Machine Learning

Haohan Wang
Lecture 28, April 29, 2020
A Brief Overview of Trustworthy Machine Learning

- New Challenges of Modern Machine Learning
  - Empirical Observations
  - Trade-off between Accuracy and Robustness
- Cross-domain Robust Models
  - Generalization Analysis of Domain Adaptation
  - Method Overview – It’s mostly about Invariance
  - Domain Generalization and Beyond
- Adversarial Robust Models
  - The Attack vs. Defense Arm Race Highlights
  - Adversarial Training and Its Recent Developments
  - Certified Robustness and Generalization Analysis
New Challenges of Modern Machine Learning
What if I told you…

- We can predict whether you use chopsticks from your genome.
  - HLA-A1 gene
  - (Vilhjálmsson and Nordborg, 2013)

- We can predict whether you speak English or Welsh from your genome
  - (Weale et al 2002)

- Trustworthy statistical models research dates to decades ago…
  - E.g., population stratification in GWAS (Devlin, B. and Roeder, K. 1999)
Behind the Glory of Deep Learning

- Adversarial example
  - (Szegedy et al 2013)

- CNN’s tendency in superficial statistics
  - (Jo and Bengio 2017)
  - (Geirhos et al 2019)
The Misalignment between CNN and Human

- CNN captures high-frequency information
  - (Wang et al, 2020)
The Misalignment between CNN and Human

- CNN captures high-frequency information
  - (Wang et al, 2020)
The New Challenge of Modern Machine Learning

- The misalignment between models and the nature/human
  - (Wang et al, 2020)

The central problem of today’s lecture
The Tradeoff between Accuracy and Robustness

- It depends on how one defines “robustness”, but roughly

  - Formal Discussions
    - Tsipras et al 2019
    - Zhang et al 2019
    - Yang et al 2019
    - Wang et al 2020
Why robustness is valued when we have accuracy

- What the current academy is good at
Why robustness is valued when we have accuracy

- What the world needs

Data collection
The Solution

- To summarize it in one sentence
  - Machine learning is about generalization

\[ R(h) \leq R_{emp}(h) + f(|H|, N, \delta) \]

- Cross-domain Robustness
  - Almost all about hypothesis space (i.e., inductive bias, regularization)

- Adversarial Robustness
  - The dominant solution increases the number of samples

The central solution of today’s lecture
Cross-Domain Robustness: Domain Adaptation, Domain Generalization, and Beyond
Domain Adaptation

- Model is trained on one distribution/domain, but tested on a “similar but different” distribution/domain
  - Supervised domain adaptation
    - Some labelled samples in the test distribution are available during training
  - Unsupervised domain adaptation
    - Only unlabeled samples in the test distribution are available during training
How is the generalization changed in UDA

- $\mathcal{H}$-divergence: a measure of two distributions given a hypothesis class
  - $I(h)$: the set of hypotheses that are characteristic functions for the domain
  - $\mathcal{H}$-divergence between the two distributions
    \[
d_{\mathcal{H}}(\mathcal{D}, \mathcal{D}') = 2 \sup_{h \in \mathcal{H}} |\Pr_{\mathcal{D}}[I(h)] - \Pr_{\mathcal{D}'}[I(h)]|
\]
  - Can be estimated from finite samples
    - $U$ and $U'$ are samples of size $m$:
      \[
d_{\mathcal{H}}(\mathcal{D}, \mathcal{D}') \leq \hat{d}_{\mathcal{H}}(U, U') + 4\sqrt{d \log(2m) + \log(\frac{2}{\delta})} \quad \frac{1}{m}
\]
  - Estimated from a domain classifier
    - For a symmetric hypothesis class both $h$ and $1-h$ are in the hypothesis class

\[
\hat{d}_{\mathcal{H}}(U, U') = 2 \left(1 - \min_{h \in \mathcal{H}} \left[ \frac{1}{m} \sum_{x:h(x)=0} I[x \in U] + \frac{1}{m} \sum_{x:h(x)=1} I[x \in U'] \right] \right)
\]
How is the generalization changed in UDA

- symmetric difference hypothesis space
  
  \[ g \in \mathcal{H} \Delta \mathcal{H} \iff g(x) = h(x) \oplus h'(x) \text{ for some } h, h' \in \mathcal{H}. \]
  
  \[ d_{\mathcal{H} \Delta \mathcal{H}}(\mathcal{D}_S, \mathcal{D}_T) = 2 \sup_{h, h' \in \mathcal{H}} \left| \Pr_{x \sim \mathcal{D}_S} [h(x) \neq h'(x)] - \Pr_{x \sim \mathcal{D}_T} [h(x) \neq h'(x)] \right| \]

- Generalization analysis
  - (Ben-David et al 2010)

  \[ \epsilon_T(h) \leq \epsilon_S(h) + \frac{1}{2} \hat{d}_{\mathcal{H} \Delta \mathcal{H}}(\mathcal{U}_S, \mathcal{U}_T) + 4 \sqrt{\frac{2d \log(2m') + \log(\frac{2}{\delta})}{m'}} + \lambda \]
  
  \[ \lambda = \epsilon_S(h^*) + \epsilon_T(h^*) \]
  
  \[ h^* = \arg\min_{h \in \mathcal{H}} \epsilon_S(h) + \epsilon_T(h) \]

A hypothesis space that is small enough so that the model does not differentiate source and target distribution.
Domain Adversarial Neural Network

- Find a classifier that
  - has small error on the source distribution
  - unable to differentiate source and target distributions
  - (Ganin et al 2016)

\[
e_T(h) \leq \epsilon_S(h) + \frac{1}{2} d_{\mathcal{H}_2}(\mathcal{U}_S, \mathcal{U}_T) + 4 \sqrt{\frac{2d \log(2m') + \log(\frac{\delta}{\delta})}{m'}} + \lambda
\]
Invariance regularization is not panacea

- Invariant representation and small source risk are not sufficient
  - (Zhao et al, 2019)

- Labelling function also matters

\[
\epsilon_T(h) \leq \epsilon_S(h) + d_{\hat{H}}(D_S, D_T) + \min\{\mathbb{E}_{D_S}[|f_S - f_T|], \mathbb{E}_{D_T}[|f_S - f_T|]\}
\]

Distance between labelling functions
Domain Adaptation Method Highlights

- In the deep learning era:
  - Essentially, to design an inductive bias that learns an invariance representation between source and target distributions
  - How to force the representations to be invariant?
    - Through a distance metric
      - e.g., Maximum Mean Discrepancy (MMD) (Rozantsev et al. 2016)
    - Through an adversarial classifier
      - e.g., Domain Adversarial Neural Network (DANN) (Ganin et al. 2016)
    - Through reconstruction
      - e.g., Deep Reconstruction Classification Network (DRCN) (Ghifary et al. 2016)
Domain Adaptation Method Highlights

- With the power of samples
  - Self-Training for Gradual Domain Adaptation
  - (Kumar et al. 2020)

- Practical insights
  - There must be an incentive for the model to update itself
  - Regularization is necessary
  - Label-sharpening (instead of probabilistic labels) is necessary
Domain Generalization

- Are models trained for domain adaptation good enough for a general real-world deployment?
  - Probably not. --- there might be other distributions that are not considered as target distributions during training.

- Domain Generalization
  - Train with samples from different domains, and test it with a new distribution
Domain Generalization Methods Highlights

- Taking advantage of the domain identifications
  - Central assumption:
    - The model that can generalize similarly and well across different training distributions will be able to generalize well to an arbitrary target distribution
- Method Development
  - Forcing invariance across different distributions
    - e.g., select-additive learning (Wang et al., 2017)
    - Ensemble of classifiers (one for each domain)
      - e.g., best-source forward (Mancini et al., 2018)
  - Meta-learning
    - e.g., iteratively split training domains to virtual training and testing domains (Li et al., 2018)
Are models trained for domain generalization good enough for a general real-world deployment?

- Probably not. – the partitions of domains may not always be available in the real world.

Beyond Domain Generalization

- Train with samples from some domains, and test it with a new distribution.
Beyond Domain Generalization

- Now, we probably have a problem that is real enough
  - But how to solve it then?

- Find a hypothesis space that pick less superficial signals
  - Case Studies:
    - Neural GLCM (Wang et al. 2019a)
    - Patch-wise Adversarial Regularization (Wang et al. 2019b)
Learning Robust Representations by Projecting Superficial Statistics

- The challenge
  - Neural networks learns textural signals
    - (Jo and Bengio 2017)
    - And many examples we have seen
A Superficial Statistics Learner

- Existing computer vision techniques
  - SURF (Bay et al., 2006)
  - LBP (He & Wang, 1990)
  - GLCM (Haralick et al., 1973)
- Semantic vs. Textural Experiments
  - Digits (MNIST, SVHN, MNIST-M, USPS)
    - Digit classification vs. Domain classification
  - Rotated MNIST
    - Digit classification vs. Rotation classification
  - FFT Kernelled MNIST
    - Digit classification vs. Kernel classification

<table>
<thead>
<tr>
<th></th>
<th>LBP</th>
<th>SURF</th>
<th>GLCM</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Digit</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semantic</td>
<td>0.179</td>
<td>0.563</td>
<td>0.164</td>
</tr>
<tr>
<td>Textural</td>
<td>0.527</td>
<td>0.809</td>
<td>0.952</td>
</tr>
<tr>
<td><strong>Rotated</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semantic</td>
<td>0.155</td>
<td>0.707</td>
<td>0.214</td>
</tr>
<tr>
<td>Textural</td>
<td>0.121</td>
<td>0.231</td>
<td>0.267</td>
</tr>
<tr>
<td><strong>FFT</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Semantic</td>
<td>0.710</td>
<td>0.620</td>
<td>0.220</td>
</tr>
<tr>
<td>Textural</td>
<td>0.550</td>
<td>0.200</td>
<td>0.490</td>
</tr>
</tbody>
</table>
Gray-Level Co-Occurrence Matrix

- Count the number of pixel pairs under a certain direction

```
1 1 7 5 3 2
5 1 6 1 2 5
8 8 6 8 1 2
4 3 4 5 5 1
8 7 8 7 6 2
7 8 6 2 6 2
```

Image

Pixel

GLCM
Gray-Level Co-Occurrence Matrix

- Count the number of pixel pairs under a certain direction
Gray-Level Co-Occurrence Matrix

- Count the number of pixel pairs under a certain direction

Image

Pixel

GLCM

© Eric Xing @ CMU, 2005-2020
Gray-Level Co-Occurrence Matrix

- Count the number of pixel pairs under a certain direction

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>7</td>
<td>5</td>
<td>3</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>1</td>
<td>6</td>
<td>1</td>
<td>2</td>
<td>5</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>8</td>
<td>8</td>
<td>6</td>
<td>8</td>
<td>1</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>5</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>7</td>
<td>8</td>
<td>7</td>
<td>6</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>7</td>
<td>8</td>
<td>6</td>
<td>2</td>
<td>6</td>
<td>2</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

GLCM

© Eric Xing @ CMU, 2005-2020
Neural Gray-Level Co-Occurrence Matrix

- Built into a neural network
  - Enable end-to-end learning
  - Differentiate weights
Synthetic Experiments: TESTING HEX Method

- Office data set
  - Webcam (W)
  - Amazon (A)
  - DSLR (D)
- Testing with the Beyond Domain Generalization setting

<table>
<thead>
<tr>
<th>Train</th>
<th>Test</th>
<th>Baseline</th>
<th>HEX</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+W</td>
<td>D</td>
<td>0.405±0.016</td>
<td>0.343±0.030</td>
</tr>
<tr>
<td>D+W</td>
<td>A</td>
<td>0.112±0.008</td>
<td>0.147±0.004</td>
</tr>
<tr>
<td>A+D</td>
<td>W</td>
<td>0.400±0.016</td>
<td>0.378±0.034</td>
</tr>
</tbody>
</table>
OFFICE DATA SET: A CLOSER LOOK

- Distribution similarities between D and W
- Two classifier:
  - C1: classifying object based on object and background
  - C2: classifying object based on object, ignoring background

<table>
<thead>
<tr>
<th>Train</th>
<th>Test</th>
<th>Baseline (C1)</th>
<th>HEX (C2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+W</td>
<td>D</td>
<td>0.405±0.016</td>
<td>0.343±0.030</td>
</tr>
<tr>
<td>D+W</td>
<td>A</td>
<td>0.112±0.008</td>
<td>0.147±0.004</td>
</tr>
<tr>
<td>A+D</td>
<td>W</td>
<td>0.400±0.016</td>
<td>0.378±0.034</td>
</tr>
</tbody>
</table>
Learning Robust Presentations by Penalizing the Local prediction power

- The challenge
  - Neural networks can predict through local signals, that do not align well with the annotation of the dataset
Patch-wise Adversarial Regularization

C channels
$M \times N$ patches

$L$ logits
Patch-wise Adversarial Regularization

C channels
$M \times N$ patches

(Reduce Gradient)

C inputs for every patch

$C \times L$

$L$ logits

$M \times N \times L$ logits

© Eric Xing @ CMU, 2005-2020
Results – ImageNet Sketch

ImageNet-Sketch Data set
- ImageNet-Sketch data set consists of 50000 images, 50 images for each of the 1000 ImageNet classes.

<table>
<thead>
<tr>
<th>Results</th>
<th>AlexNet-PAR</th>
<th>AlexNet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>prediction</td>
<td>prediction</td>
</tr>
<tr>
<td>stethoscope</td>
<td>0.6608</td>
<td>hook</td>
</tr>
<tr>
<td>tricycle</td>
<td>0.9260</td>
<td>safety pin</td>
</tr>
<tr>
<td>Afghan hound</td>
<td>0.8945</td>
<td>swab (mop)</td>
</tr>
<tr>
<td>red wine</td>
<td>0.5999</td>
<td>goblet</td>
</tr>
</tbody>
</table>
Adversarial Robustness

\[ x + 0.007 \times \text{sign}(\nabla_x J(\theta, x, y)) \]

= 

"panda" 
57.7% confidence

"nematode" 
8.2% confidence

"gibbon" 
99.3% confidence
What are adversarial examples

- Recap, with the help of the most popular example
- What are adversarial examples?
  - Samples with carefully crafted patterns added to an original sample, and
    - look identical to the original image to human,
    - but predicted by a model significantly different from the original sample.
    - This sounds quite magic, but not rigorous enough...
  - With sample $x$, model $f(\theta)$, the adversarial example $x'$ is
    \[
    x' = \arg \min_{x'} f(x';\theta) \neq f(x;\theta) d(x, x')
    \]
    - A result of an optimization, $d(,)$ is a distance
    - Hopefully, this does not sound too magic anymore.
    - How magic it looks to human depends on the choice of $d(,)$
What are adversarial examples

- Why are there adversarial examples?
  - Adversarial examples are direct outcomes of the fact that models are capturing superficial signals
    - (Ilyas et al. 2019)
    - (Wang et al. 2020)

- Why is adversarial robustness a popular topic
  - Hmm..., one reason might actually be it appears magic
  - The alignment between human and model may play a significant role in the real-world deployment of machine learning

- As a popular topic
  - Tons of efforts made to defend a model against adversarial examples
  - Unfortunately, even more efforts made to generate new adversarial examples...
The Attack vs. Defense Arm Race Highlights

- **Attack Methods**
  - FGSM
    - (Goodfellow et al. 2014)
  - PGD
    - (Madry et al. 2017)

- **Defense Methods**
  - TRADES
    - (Zhang et al. 2019)
  - Adversarial Training
    - (Madry et al. 2017)
The Attack vs. Defense Arm Race Highlights

- Fast Gradient Sign Method (FGSM)
  - (Goodfellow et al. 2014)
  - Intuitively: a reverse way of training
    - Training (one epoch) is to update parameters to decrease loss, according to the gradient.
    - FGSM is to update the data to increase loss, according to the sign of the gradient.

\[ x' = x + \epsilon \text{sign}(\nabla_x l(x, y; \theta)) \]
The Attack vs. Defense Arm Race Highlights

- PGD (Projected Gradient Descent) attack
  - (Madry et al. 2017)
  - Fun fact, it’s previously invented under the name Basic Iterative Method (BIM)
  - Algorithm: Performing FGSM multiple times, with smaller step size.
    - Start with a random perturbation within the $l_p$ norm ball
    - Perform FGSM: $x' = x + \epsilon \text{sign}(\nabla_{x}l(x, y; \theta))$
    - Project back to the $l_p$ norm ball
    - Repeat multiple iterations

- Probably the most powerful attack method nowadays ($l_p$ norm distance)
The Attack vs. Defense Arm Race Highlights

- TRADES Defense
  - (Zhang et al. 2019)
  - Take advantage of the worst-case example
    - To identify the worst-case example
      - Apply attack during training to identify $x'$
      - Regularize the distance of embedding between $x$ and $x'$
        - KL divergence regularization over softmax
Competition: NeurIPS 2018 Adversarial Vision Challenge

The methodology is the foundation of our entry to the NeurIPS 2018 Adversarial Vision Challenge in which we won the 1st place out of ~2,000 submissions, surpassing the runner-up approach by 11.41% in terms of mean $l_2$ perturbation distance.
The Attack vs. Defense Arm Race Highlights

- Adversarial Training as a Defense
  - (Madry et al. 2017)
  - Summarize in one sentence:
    - augment the training data with generated adversarial examples during training
  - The most popular defense method
    - When integrated with PGD
      - Impressive empirical performance and simplicity

As we see in next slide.

The community is making it easier:
- A faster (than PGD) way to generate the adversarial examples (that are equally powerful to the ones generated by PGD)
  - Adversarial training for free. Shafahi et al. 2019
  - Fast is better than free. Wong et al. 2020
Perceptually Aligned Representations from Adversarial Training w. PGD

- With PGD + adversarial training, adversarial examples are not that magical any more
- (Santurkar et al. 2019)
Perceptually Aligned Representations from Adversarial Training w. PGD

- With PGD + adversarial training, the same model may fulfill multiple tasks
  - (Santurkar et al. 2019)
The Attack vs. Defense Arm Race

- With the impressive performance of PGD + adversarial training, can we conclude the development of defense methods?
  - Maybe not: It's simple, not not simple enough
  - Definitely not: there might be other attack methods can penetrate the defended model
    - So… is this race endless?
      - Fortunately, the community moves to provably robust models.
Certified Robust Models

- Certified Robustness through Randomized Smoothing
  - (Cohen et al. 2019)
  - Randomized smoothing method
    \[ g(x) = \arg \max_{c \in Y} \mathbb{P}(f(x + \varepsilon) = c) \]
    
    where \( \varepsilon \sim \mathcal{N}(0, \sigma^2 I) \)

- Certified Robustness
  - Over \( l_2 \) norm distance
Looking into the Future

Data

“semantic”

“superficial”

label

what a model picks up
A Brief Overview of Trustworthy Machine Learning

- New Challenges of Modern Machine Learning
  - Empirical Observations
  - Trade-off between Accuracy and Robustness
- Cross-domain Robust Models
  - Generalization Analysis of Domain Adaptation
  - Method Overview – It’s mostly about Invariance
  - Domain Generalization and Beyond
- Adversarial Robust Models
  - The Attack vs. Defense Arm Race Highlights
  - Adversarial Training and Its Recent Developments
  - Certified Robustness and Generalization Analysis
References

References

References