Introduction to Deep Learning Chapter 4: Error Analysis and Transfer Learning

Pao-Ann Hsiung

National Chung Cheng University

Contents

Error Analysis
Transfer Learning
Multi-task Learning

Contents

Error Analysis
Transfer Learning
Multi-task Learning

Error Analysis: Single Idea

錯誤標記要怎辦

如果錯誤有pattern

Some examples are mislabeled, what to do ?

Mislabeled dev set examples

- Get 100 mislabeled examples
- Count manually how many are for dogs mislabeled?
  - Only 5%? Not worth the effort!
  - 50%? Worth it!
    - Error rate: 10% ? 5%!
- What is the CEILING?

錯誤的結果

Error Analysis: Evaluate Multiple Ideas in Parallel 做一張表找出錯誤原因

Ideas of cat detection 錯誤原因較多的，找較多沒有問題的圖片再訓練，提高準確
Fix pictures of dogs being recognized as cats
Fix great cats (lions, panthers, etc.) being misrecognized
Improve performance on blurry images

Image	Dog	Great Cats	Blurry	Instagram	Comments
1	✓				Flying
2		✓			
3		✓	✓		Rainy day at zoo
					加所有錯誤
% of total	10%	45%	65%	12%	看原因

Do I need to bother about incorrectly labeled examples in training dataset?

x       

y 1 0 1 1 0 1 1

隨機錯誤影響不大，可以忽略

DL algorithms are quite robust to random errors in the training set, but not to systematic errors.

How about incorrectly labeled examples in the dev set?

Image	Dog	Great Cat	Blurry	Incorrectly labeled	Comments
…					
98				✓	Labeler missed cat in background
99		✓			
100				✓	Drawing of a cat; Not a real cat.
% of total	8%	43%	61%	6%	

Overall dev set error ---------- 10$ or 2%
Errors due to incorrect labels -------- 0.6% or 0.6%
Errors due to other causes ---------- 9.4% or 1.4%

花時間在此

Correcting incorrect dev/test set examples

資料來源相同，相同分配。training結果會很好
但testing結果可能會不好，資料來源不同。

Apply same process to your dev and test sets to make sure they continue to come from the same distribution

Consider examining examples your algorithm got right, as well as, ones it got wrong.

Train and dev/test data may now come from slightly different distributions.

將test的資料放入train的資料中訓練模型可以降低誤差。

High bias 在train時錯誤很大
High variance train與tes的錯誤誤差很大

Speech Recognition Example

如果要在雜音環境下做語音辨識　但是難以收集到有雜音的資料　自行加入雜音在資料中

- Noisy
  - Café
  - Car
- Accent
- Far fro
- Young
- Stuttering
- …

口音問題

Guideline:

Build your first system quickly, then iterate

- Set up dev/test set
- l system
- /ariance
- Error
- prioritize

analysis to
next steps.

找出bias或者variance產生的問題

How to create/distribute your data into training/dev/test sets?

Cat APP Example

Data from webpages

300,000 images

Data from mobile app

12,000 images

Option 1: Shuffle, Train 306,000, Dev 3,000, Test 3,000 BAD!

Option 2: No shuffle, Train 300,000(web)+6,000(app),
Dev 3,000 (app), Test 3,000 (app) GOOD!

Speech Recognition Example

Training

Purchased data

Smart speaker control

Voice keyboard

600,000 utterances

Dev/test

Speech activated rearview mirror

20,000 utterances

How to distribute data in training/dev/test sets?

Bias and Variance with Mismatched Data Distributions

Cat Classifier Example

Assume humans get nearly 0% error.
Training Error: 1%
Dev Error: 10%
High variance

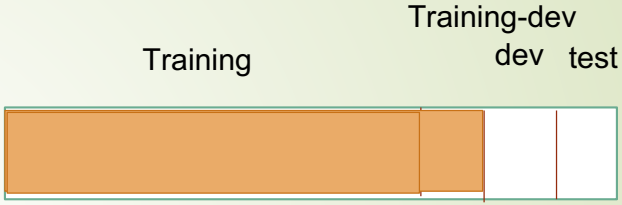
High Variance? May be, but 2 factors:
- Training Problem: did not see data in dev set
- Distribution Problem: Training vs. Dev Sets different

Overfitting problem

Regularization/Dropout/batch normalization

使用些方法但錯誤還是很大，有可能是Distribution Problem

Training-dev set: Same distribution as training set, but not used for training



Error	Case 1	Case 2	Case 3	Case 4
Training	1%	1%	10%	10%
Training-Dev	9%	1.5%	11%	11%
Dev	10%	10%	12%	20%
Variance	High			
Bias			High	High
Data Mismatch		Yes		Yes

加入training錯誤還是很大

Bias/Variance on Mismatched Training and Dev/test Sets

Error	Case 1	Case 2
Human Level	4%	4%
Training Set	7%	7%
Training-Dev	10%	10%
Dev	12%	6%
Test	12%	6%

Avoidable Bias

Variance

Data Mismatch

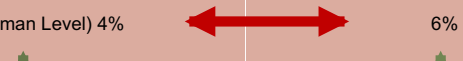
Overfitting to Dev Set

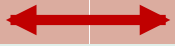
Training Set difficult

Dev/Test Set easier

General Information

Error	General speech recognition	Rearview mirror speech data
Human level	(Human Level) 4%	6%
Examples trained on	(Training Error) 7%	6%
Examples not trained on	(Training-Dev Error) 10%	(Dev/Test Error) 6%





Addressing Data Mismatch

Addressing data mismatch

Carry out manual error (人為錯誤) analysis to try to understand difference between training and dev/test sets

- Dev data might have noisy background noise, while training data might be more clear

Make training data more similar; or collect more data similar to dev/test sets (盡量收集相同資料)

Artificial data synthesis

"The quick brown fox jumps over the lazy dog."

+

Car noise

=

Synthesized in-car audio

10,000 hours of speech
1 hour of car noise
If you repeat the 1 hour of car noise 10,000 times, then the network model might overfit to the 1 hour car noise.

Contents

Error Analysis
Transfer Learning
Multi-task Learning

1 遷移學習提出的背景及歷史

1、遷移學習提出背景

在機器學習、深度學習和數據挖掘的大多數任務中，我們都會假設training和inference時，採用的數據服從相同的分布（distribution）、來源於相同的特徵空間（feature space）。但在現實應用中，這個假設很難成立，往往遇到一些問題：

1、帶標記的訓練樣本數量有限。比如，處理A領域（target domain）的分類問題時，缺少足夠的訓練樣本。同時，與A領域相關的B（source domain）領域，擁有大量的訓練樣本，但B領域與A領域處於不同的特徵空間或樣本服從不同的分布。

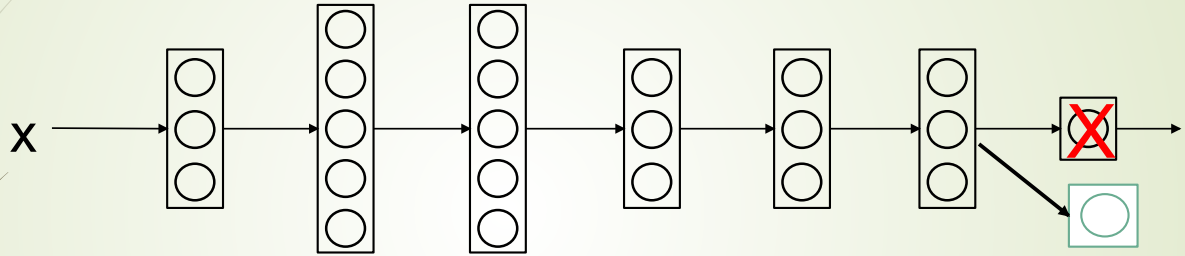
2、數據分布會發生變化。數據分布與時間、地點或其他動態因素相關，隨著動態因素的變化，數據分布會發生變化，以前收集的數據已經過時，需要重新收集數據，重建模型。

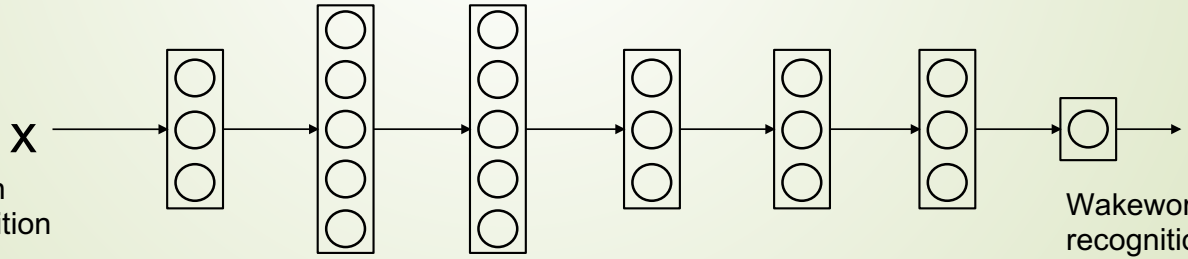
這時，知識遷移（knowledge transfer）是一個不錯的選擇，即把B領域中的知識遷移到A領域中來，提高A領域分類效果，不需要花大量時間去標註A領域數據。遷移學習，做為一種新的學習範式，被提出用於解決這個問題。

原文網址：https://kknews.cc/zh-tw/education/ov5klnp.html

Transfer Learning

Cat Recognition ?
Radiology Diagnosis





Speech recognition

Wakeword recognition e.g ., Hey Siri!

When transfer learning makes sense

Task A and B have the same input x.

You have a lot more data for Task A than Task B.

Low level features from A could be helpful for learning B.

- Image recognition helpful for radiology diagnosis (放射診斷)
  - Difficult to get that many radiology data
- Speech recognition helpful for wakeword recognition

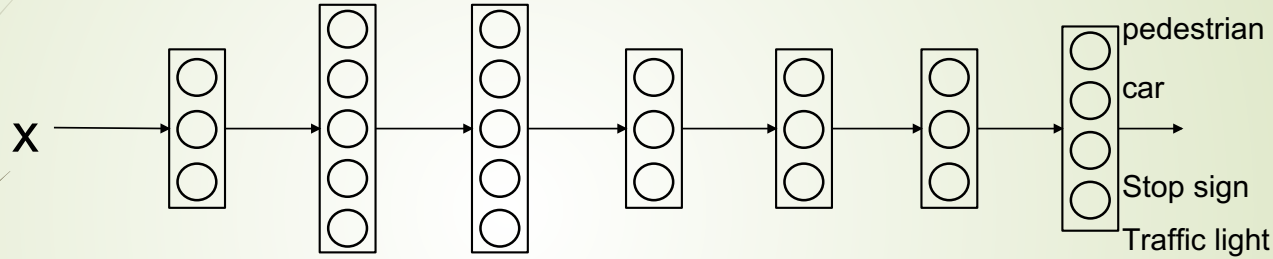
Contents

Error Analysis
Transfer Learning
Multi-task Learning

多目標學習（Multi-Task Learning）是一個在機器學習領域存在許久的課題，我們總希望能夠訓練出一個精簡優雅的模型，同時預測多個目標，而不是對於每一個目標都設計單獨一套模型。舉個例子來說：想要訓練電腦判斷一張圖片是否有包含人的同時，我們也想要電腦順便給我們更多的資訊，像是人的姿態、四肢的位置、實際的身高跟距離。在現今資訊爆炸以及AI不斷進化的時代，未來這樣的多目標學習只會越來越普及，人類的大腦很厲害一下子就能做到舉一反三，但想要設計並有效訓練機器達成多目標預測並不容易。

Simplified Autonomous Driving Example

Neural Network Architecture



When multi-task learning makes sense

- Training on a set of tasks that could benefit from having shared lower-level features
  - Example: all are features of a road
- Usually, amount of data you have for each task is quite similar
  - 1000 training samples for each task, so if there are 100 tasks to learn, then you will have 1000 x 100 = 100,000 training samples
- Can train a big enough neural network to do well on all the tasks
  - Will not work well if the network is not large enough

End-to-end deep learning

What is end-to-end learning?

Speech recognition example

Traditional stages:

Audio (x) → features → Phonemes → Words → Transcript (y)

End-to-end Deep Learning

Audio → Transcript (a single neural network)

Depends on the amount of training data we have:

3,000 hours: pipelined approach might work better

10,000 to 100,000 hours: end-to-end deep learning works better

Face Recognition

Normally, face recognition is divided into 2 stages

- Face detection: where the face is located?
- Face identification: identical to which face in database?

2-stage face recognition is more efficient and more accurate than 1-stage end-to-end learning because

- Not that many data for end-to-end learning; 2-stages have more data samples for each stage
- 2 simpler sub-problems

More examples

Machine Translation
English ? Text Analysis ? … ? French
Large dataset for English, French translations
End-to-end deep learning works well

Estimating child's age
Image ? Bones ? Age
Not enough end-to-end data (image ? age)
multi-step approach works better in this application

Pros and cons of end-to-end deep learning

Pros:

- Let the data speak. x ? y
- Less hand-designing of components needed

Cons

- May need large amount of data. (x, y)
- Excludes potentially useful hand-designed components
  - 注入 To inject manually-designed features/components is useful when data is not not large enough (could be double-edged sword)

Applying end-to-end deep learning

Key question: Do you have sufficient data to learn a function of the complexity needed to map x to y?

- Example: estimating age from hand image is much more complex; however, locating the bones in the hand image is much simpler, so you don't need that much data

Autonomous Driving

Image → cars, pedestrains, etc. → route → steering

Use DL to learn individual components (image to cars pedestrians, etc.)

For other parts: use motion planning and control

Cannot use DL today to do end-to-end learning for this application: image → steering