

Error analysis

Look at dev examples to evaluate ideas

Share



90% accuracy
→ 10% error

Should you try to make your cat classifier do better on dogs? ←

Error analysis:

- Get ~100 mislabeled dev set examples.
- Count up how many are dogs.

→ 5%
5/100

10%
↓
9.5%

"ceiling"

→ 50%
50/100

10%
↓
5%

Andrew Ng

Evaluate multiple ideas in parallel

Ideas for cat detection:

- 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	✓				Pitbull
2			✓		
3		✓	✓		Rainy day at zoo
⋮	⋮	⋮	⋮		
% of total	8%	43%	61%		

Andr

This gives us an estimate of how worthwhile it might be to work on various categories of misclassification.

e.g potential improvement is higher by improving performance on great cats or blurry images

Cleaning up incorrectly labelled data

Incorrectly labeled examples

Share



	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%
 Errors due incorrect labels 0.6% ←
 Errors due to other causes 9.4% ←

Handwritten notes on the right:
 2%
 0.6%
 1.4%

Correcting incorrect dev/test set examples

Share

- 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. (2%)
- Train and dev/test data may now come from slightly different distributions.

Build your first system quickly, then iterate

Speech recognition example



- • Noisy background
 - • Café noise
 - • Car noise
 - • Accented speech
 - • Far from microphone
 - • Young children's speech
 - • Stuttering *uh, ah, um, ...*
 - • ...
- • Set up dev/test set and metric
 - Build initial system quickly
 - Use Bias/Variance analysis & Error analysis to prioritize next steps.

Training & testing on different distributions

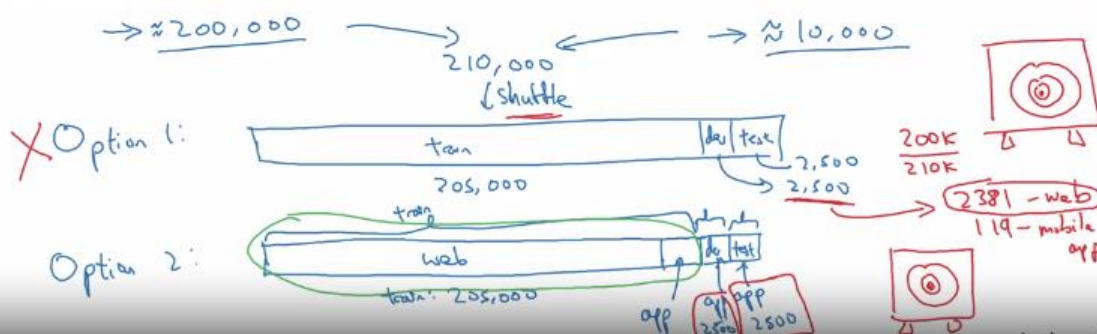
In dev and test set, keep the data you really care about, i.e. the ones that come from the distribution that you'll use in practice.

Cat app example

Data from webpages



care about this
Data from mobile app



Speech recognition example

Speech activated rearview mirror



Training

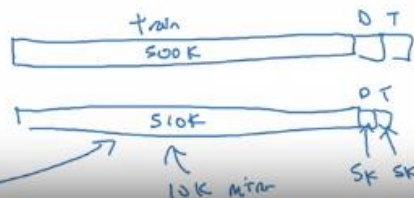
Purchased data $\downarrow \downarrow$ x, y
 Smart speaker control
 Voice keyboard

... 500,000 utterances

Dev/test

Speech activated rearview mirror

$\Rightarrow 20,000$



Bias & variance with mismatched data distributions

If dev set error is much higher than training error, we could say we have a high variance problem where our model doesn't generalize well on dev set.

But If train & dev sets come from different distributions, we can no longer say this. Maybe the dev set is just much harder to classify, because it comes from a different distribution.

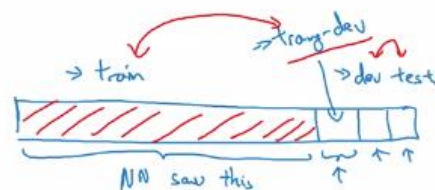
We can use a dev subset out of the train data, which we can be assured comes from same distribution as train data.

Cat classifier example

Assume humans get $\approx 0\%$ error.

Training error 1%
 Dev error 10%

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



Training error	1%	\uparrow variance	1%
\rightarrow Training-dev error	9%		1.5%
\rightarrow Dev error	10%		10%
			\uparrow data mismatch
		Variance	
Human error	0%	\uparrow Avoidable bias	10%
Training error	10%	\downarrow bias	10%
Training-dev error	11%		11%
Dev error	12%		20%
		\uparrow Data mismatch	
		Bias	Bias

Bias/variance on mismatched training and dev/test sets

Human level	4%		4%
Training set error	7%	↑ avoidable bias	7%
Training-dev set error	10%	↑ variance	10%
→ Dev error	12%	↓ data mismatch	6%
→ Test error	12%	↓ degree of similarity to dev set.	6%

More general formulation

Residual mirror

	General speech recognition	Residual mirror speech data.	
Human level	"Human level" 4%	<u>6%</u>	↑ avoidable bias
Error on examples trained on	"Training error" 7%	6%	↑ Variance
Error on examples not trained on	"Training-dev error" 10%	"Dev/Test error" <u>6%</u>	
		↔ data mismatch	

Addressing data mismatch

Addressing data mismatch

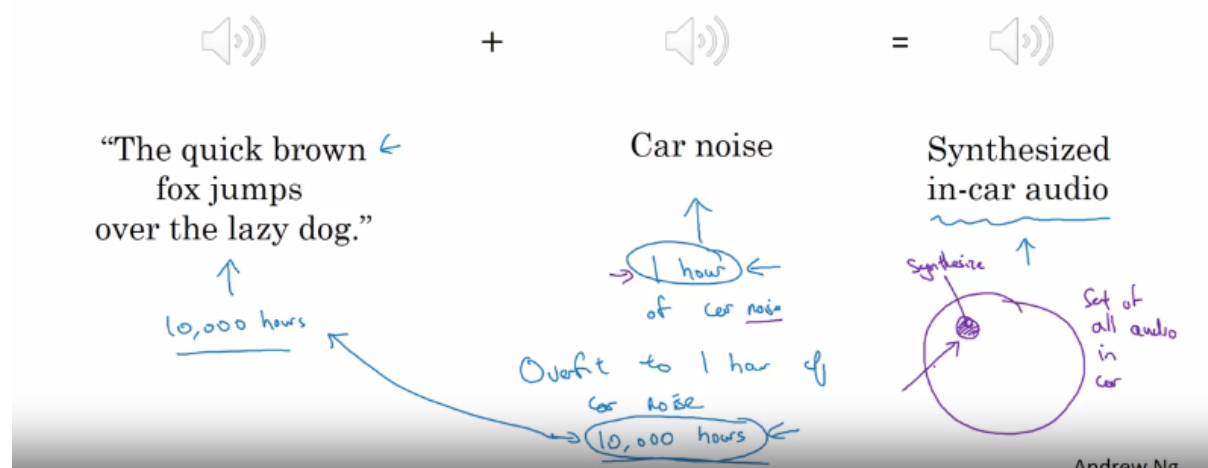
- • Carry out manual error analysis to try to understand difference between training and dev/test sets

E.g. noisy - car noise street numbers

- • Make training data more similar; or collect more data similar to dev/test sets

E.g. Simulate noisy in-car data

Artificial data synthesis



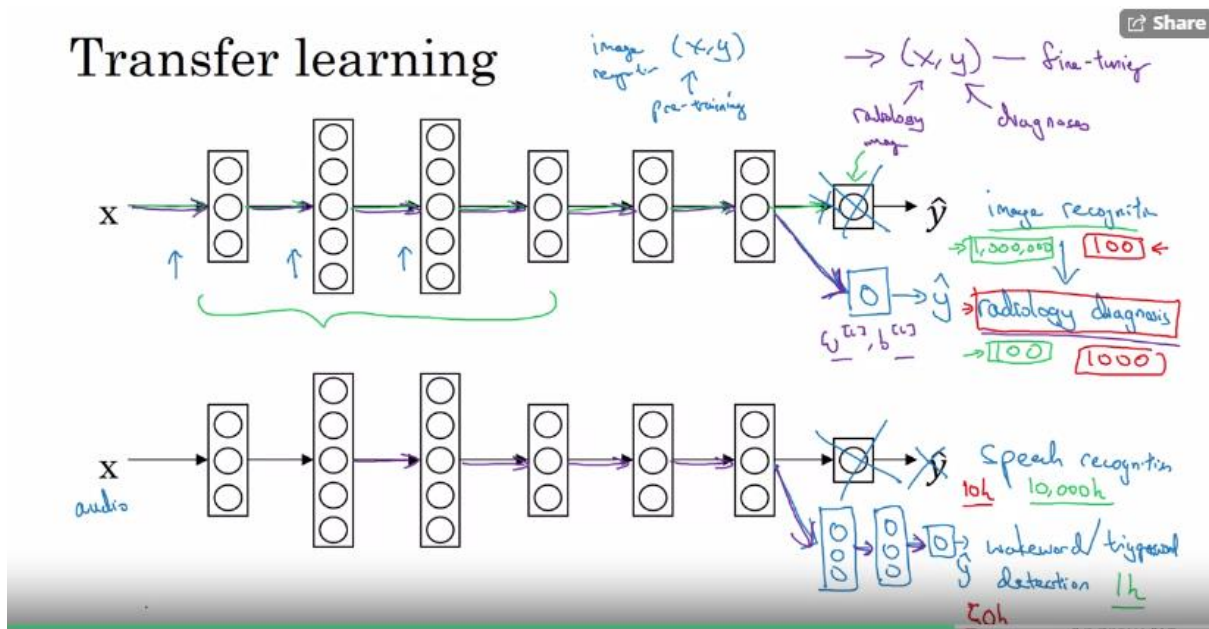
Artificial data synthesis

Car recognition:



if you synthesize just a very small subset of these cars, then to the human eye, maybe the synthesized images look fine. But you might overfit to this small subset you're synthesizing

Transfer learning



When transfer learning makes sense

Transfer from A \rightarrow B

- 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.