+ Training and Testing on different distributions

Since Deep learning algorithms are very data hungry, some teams train on different distributions.

So being able to work with mismatched data is crucial.

You care more about how the program will do on the dev/test sets.

Option 1 => Combine and shuffle.
Option 2 => Combine at a smaller scale.

+ Bias and Variance with Mismatched Data Distribution

If data come with different distributions we can't always conclude that there is a variance error.

Trying to solve this problem could result in overfitting the dev-set.

Example: More general formulation.

	General Speech recognition	Specific speech recognition	
Human Level	"Human level" 4%	6%	↑ Avoidable Bias
Error on examples Trained on	"Training error" 7%	6%	↓
Error on examples not Trained on.	"Training-dev error" 10%	"Dev/Test error" 6%	↑ Variance

⟷ data mismatch

No Systematic Way To Solve it

+ Addressing Data Mismatch:

1. Carry out manual error analysis
   To try to understand the difference between training and dev/test sets.
   (usually only look only to the dev set).

2. Make training data more similar; or, collect more data similar to dev/ test set.

One way to obtain more data, is artificial data synthesis.