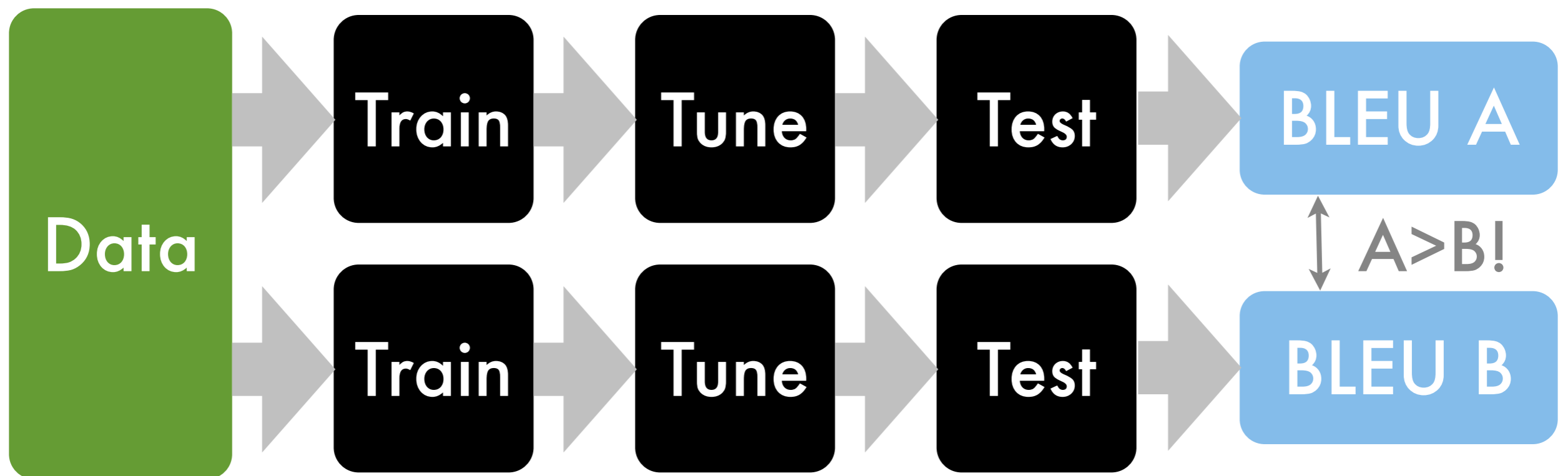


Better Hypothesis Testing for MT: Controlling for Optimizer Instability

Jonathan Clark, Chris Dyer,
Alon Lavie, & Noah Smith
Carnegie Mellon University
ACL – June 21, 2011

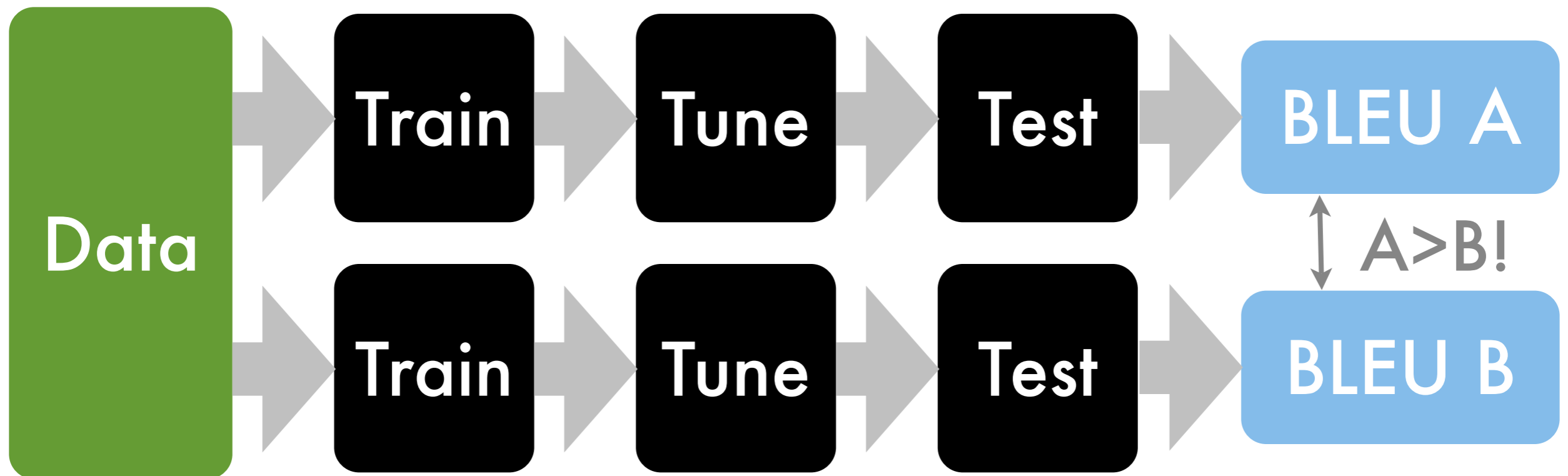
A Tale of 2 BLEU Scores

- Jon runs an experiment: $A > B$?



A Tale of 2 BLEU Scores

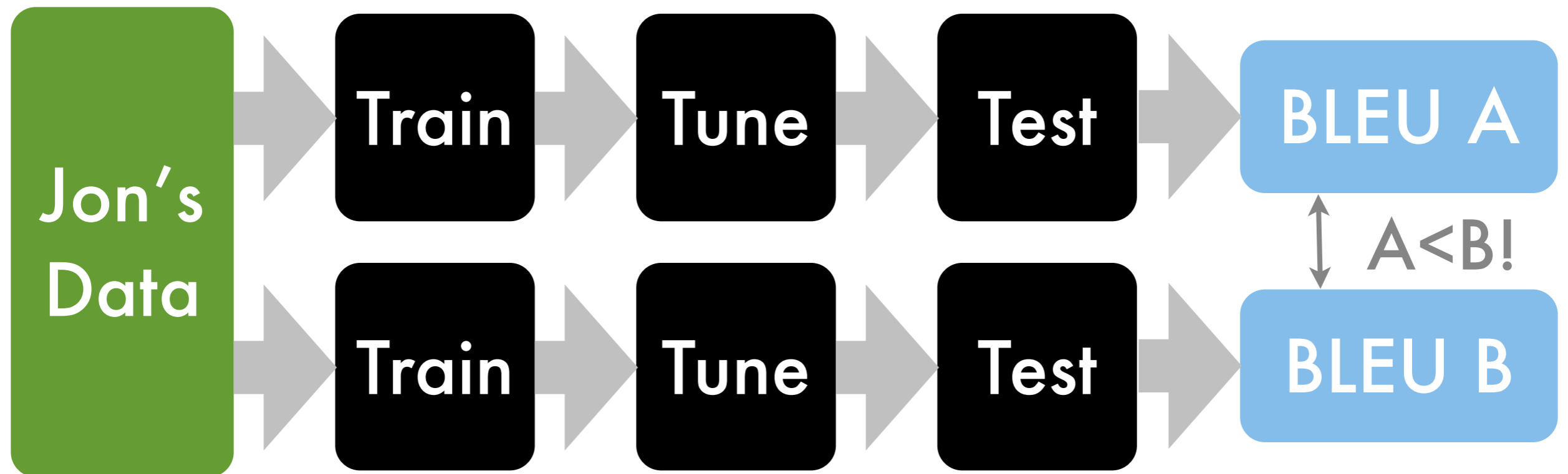
- Jon runs an experiment: $A > B$?



- If $\text{BLEU A} > \text{BLEU B}$, we conclude A is better...
...right?

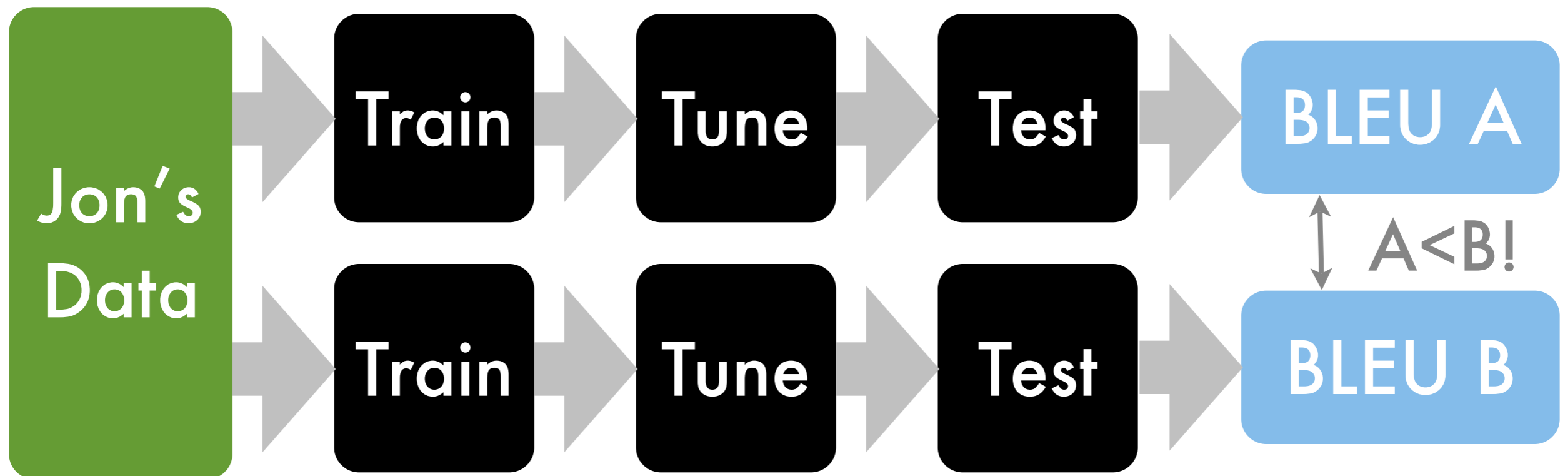
A Tale of 2 BLEU Scores

- Chris repeats Jon's experiment



A Tale of 2 BLEU Scores

- Chris repeats Jon's experiment



- But Jon found that $A > B$...
- Who's wrong?

Outline

- Why do these outcomes differ?
- Current practice: Ignore instability
- Experiments: Does instability really matter?
- Recommendation: *Measure* instability
(software provided @ the link below)

Why Outcomes Differ

- Intentional changes (**experimental variables**):
 - Tokenization, TM, LM, etc.

Why Outcomes Differ

- Intentional changes (**experimental variables**):
 - Tokenization, TM, LM, etc.
- Unintentional changes (**extraneous variables**):
 - Test sets
 - Optimizer initialization
 - Optimizer strategy
(MERT, MCMC, MIRA)

Why Outcomes Differ

- Intentional changes (**experimental variables**):
 - Tokenization, TM, LM, etc.
- Unintentional changes (**extraneous variables**):
 - Test sets } **Well-explored: Bootstrap resampling, Approximate randomization, etc.**
 - Optimizer initialization
 - Optimizer strategy
(MERT, MCMC, MIRA)

Why Outcomes Differ

- Intentional changes (experimental variables):
 - Tokenization, TM, LM, etc.
- Unintentional changes (extraneous variables):
 - Test sets } **Well-explored:** Bootstrap resampling, Approximate randomization, etc.
 - Optimizer initialization
 - Optimizer strategy (MERT, MCMC, MIRA) } **Well-known, Not Well-explored → Formalize**

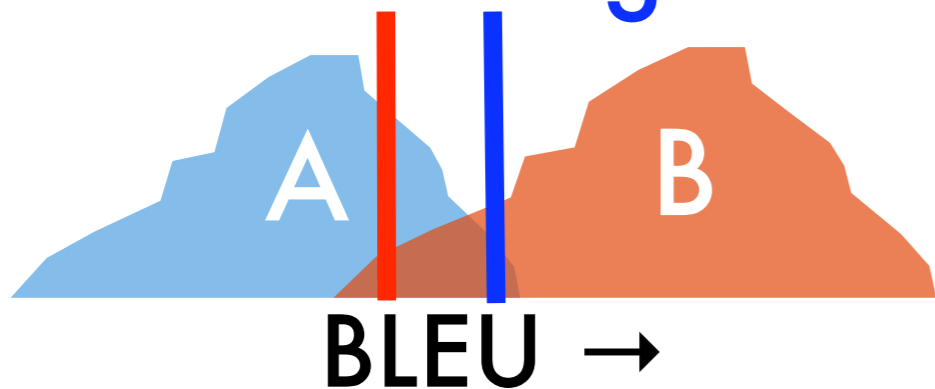
Current Practice: Instability

- Run your favorite optimizer *once* and report
 - *Single* sample from distribution over weights

Current Practice: Instability

- Run your favorite optimizer *once* and report
 - *Single* sample from distribution over weights

Low B < High A!



- High baseline (A) +
low experimental (B)

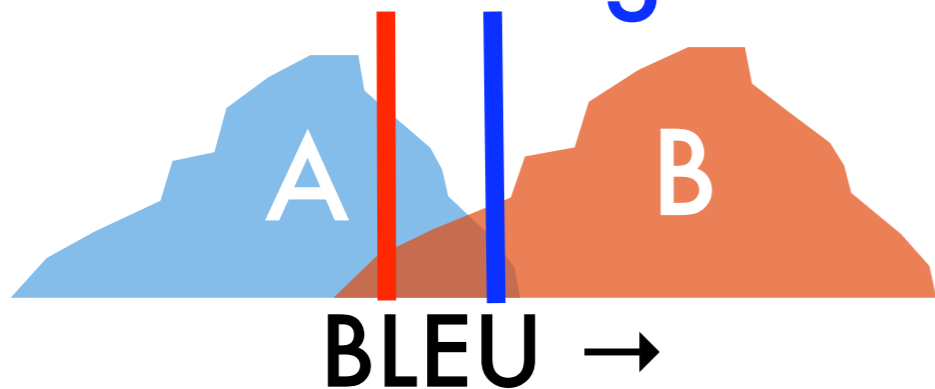


Throw away a
good idea

Current Practice: Instability

- Run your favorite optimizer *once* and report
- *Single* sample from distribution over weights

Low B < High A!

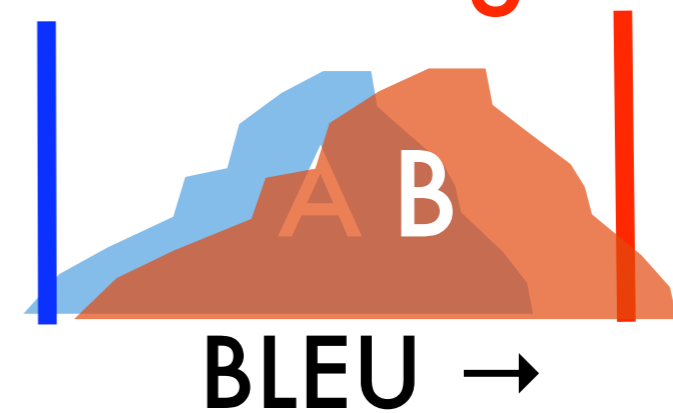


- High baseline (A) + low experimental (B)



Throw away a good idea

Low A < High B!



- Low baseline (A) + high experimental (B)



Report a spurious improvement

Experiments

- Does this variation really show up?
- Does it affect comparisons to baseline systems?

Single System Variation

System	BLEU Avg			
BTEC Chinese-English				
A	48.4			
B	49.9			
WMT German-English				
A	18.5			
B	18.7			


* See paper for BLEU, METEOR, & TER

Single System Variation

System	BLEU Avg	S_{sel}		
BTEC Chinese-English				
A	48.4	1.6		
B	49.9	1.5		
WMT German-English				
A	18.5	0.3		
B	18.7	0.3		

* See paper for BLEU, METEOR, & TER

- Well-known bootstrap resampling
- Variance over slightly different test sets:

	Sent 1	Sent 1	Sent 1
	Sent 2	Sent 2	Sent 1
	Sent 3	Sent 2	Sent 2
BLEU:	22.0	23.0	24.0
Variance (in BLEU):			
		1.0	

Single System Variation

System	BLEU Avg	S_{sel}		
BTEC Chinese-English				
A	48.4	1.6		
B	49.9	1.5		
WMT German-English				
A	18.5	0.3		
B	18.7	0.3		

- Run optimizer many times (sampling)
- Run BTEC MERT **300X**
- Run WMT MERT **50X**

Single System Variation

System	BLEU Avg	S_{sel}		
BTEC Chinese-English				
A	48.4	1.6		
B	49.9	1.5		
WMT German-English				
A	18.5	0.3		
B	18.7	0.3		

	Opt 1	Opt 2	Opt 3
Corpus BLEU:	22.0	23.0	24.0
Variance (in BLEU):		1.0	

* See paper for BLEU, METEOR, & TER

Single System Variation

System	BLEU Avg	S_{sel}	S_{dev}	
BTEC Chinese-English				
A	48.4	1.6	0.2	
B	49.9	1.5	0.1	
WMT German-English				
A	18.5	0.3	0.0	
B	18.7	0.3	0.0	

* See paper for BLEU, METEOR, & TER

- Optimizer not guaranteed to find global optimum
- Good fit on tuning data
- About the same score (not always same weights)

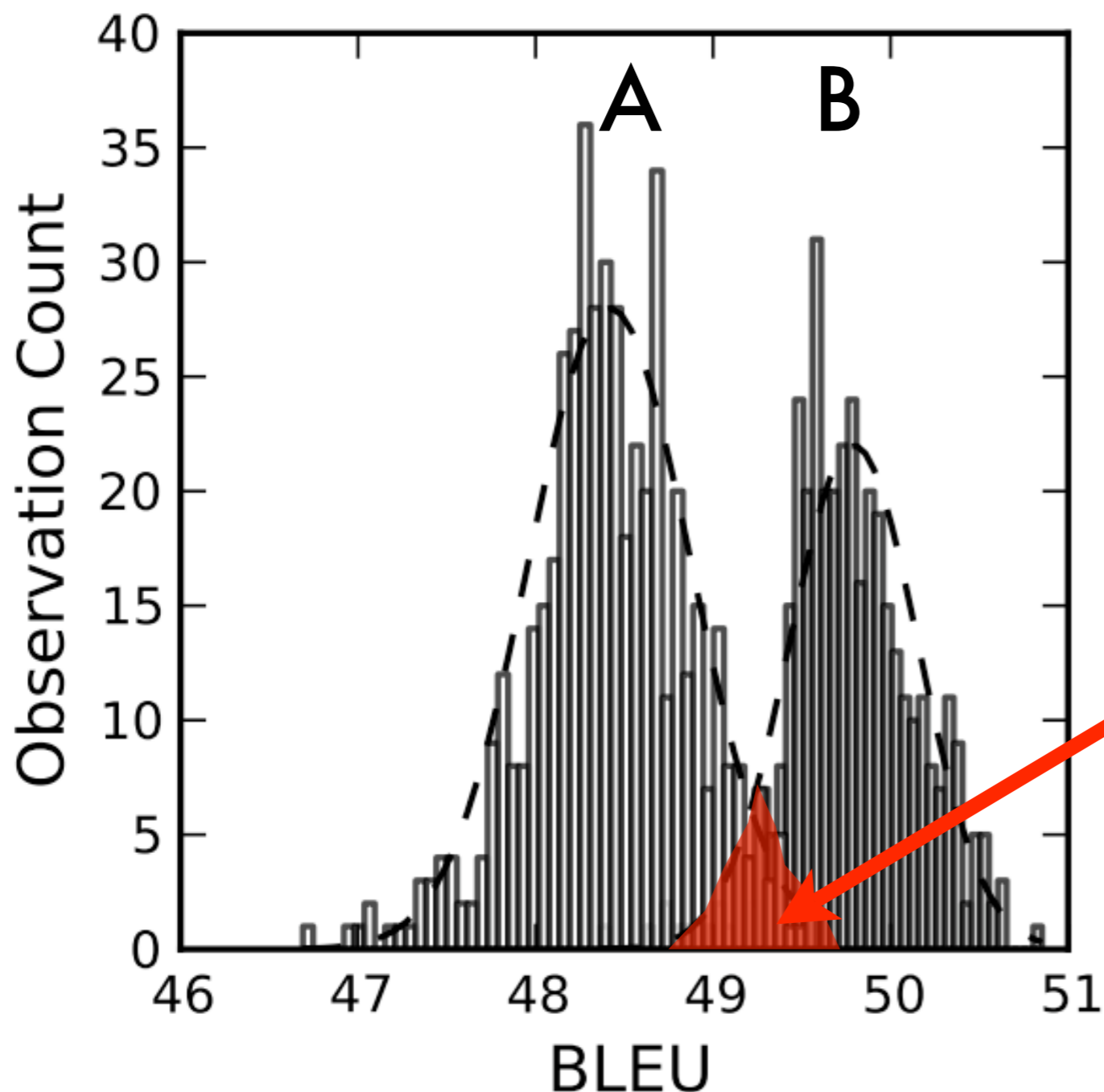
Single System Variation

System	BLEU Avg	S_{sel}	S_{dev}	S_{test}
BTEC Chinese-English				
A	48.4	1.6	0.2	0.5
B	49.9	1.5	0.1	0.4
WMT German-English				
A	18.5	0.3	0.0	0.1
B	18.7	0.3	0.0	0.2

* See paper for BLEU, METEOR, & TER

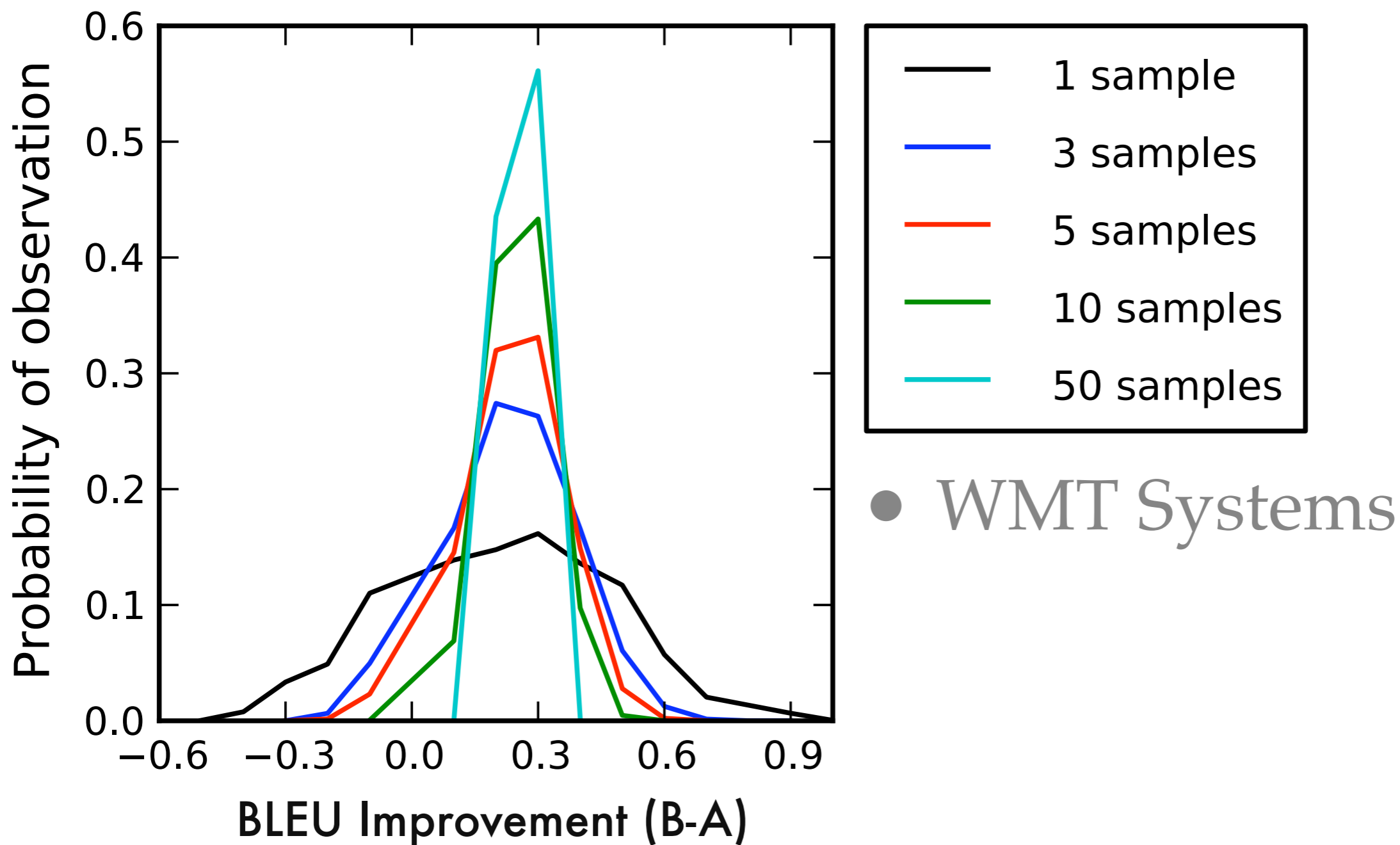
- Optimal weights for tuning may not generalize
- Many weight vectors appear equal on the tuning set, but differentiate on the test set

Two System Variation



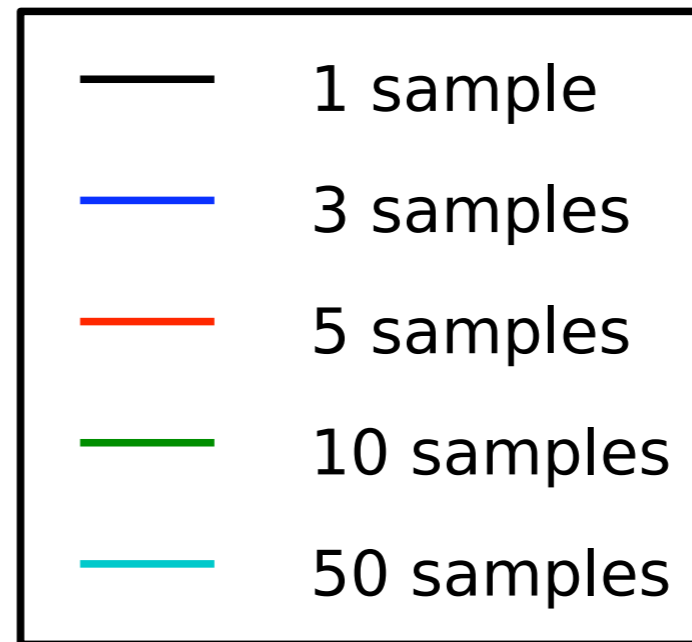
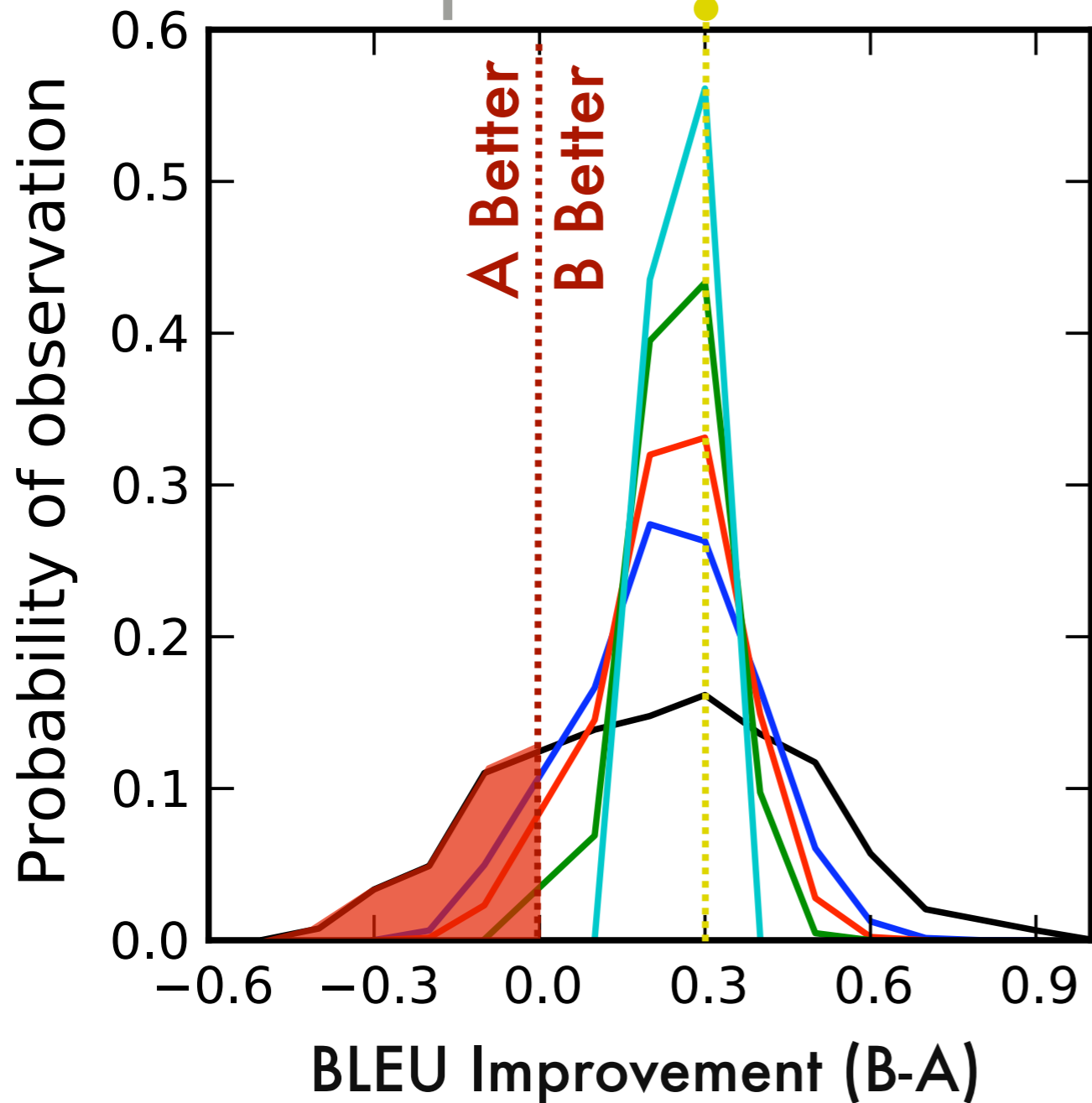
- BTEC Systems
- In the overlapping region, we could draw an opposite conclusion

How Many Runs are Needed?



How Many Runs are Needed?

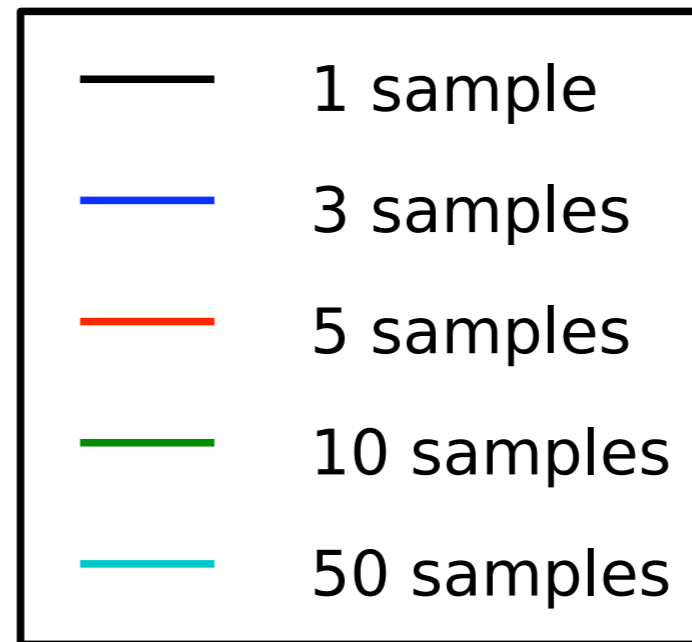
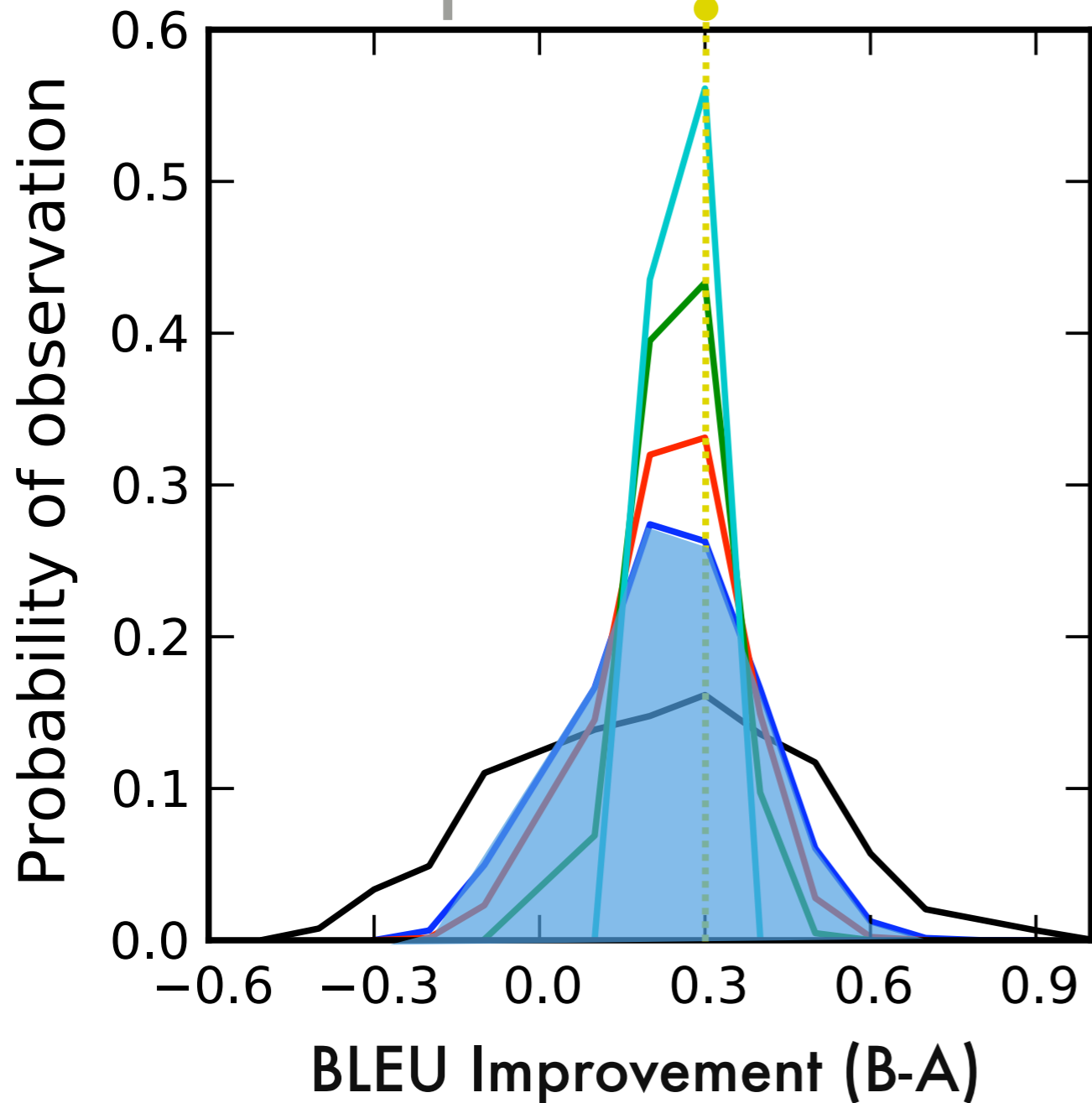
Expected Difference



- WMT Systems
- Significant chance of opposite outcome

How Many Runs are Needed?

Expected Difference



- WMT Systems
- Significant chance of opposite outcome
- Clear peak by 3

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3
Optimizer Run 1	{	Sent 1	Sys A	Sys B
		Sent 2	Sys A	Sys B
		Sent 3	Sys A	Sys B

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3
Optimizer Run 1	{	Sent 1	Sys A	Sys B
		Sent 2	Sys A	Sys B
		Sent 3	Sys A	Sys B
Optimizer Run 2	{	Sent 1	Sys A	Sys B
		Sent 2	Sys A	Sys B
		Sent 3	Sys A	Sys B
Optimizer Run 3	{	Sent 1	Sys A	Sys B
		Sent 2	Sys A	Sys B
		Sent 3	Sys A	Sys B

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3	
Optimizer Run 1	{	Sent 1	Sys B	Sys A	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	<u>Simulated Trials</u> <u>> 0.3?</u>
Optimizer Run 2	{	Sent 1	Sys B	Sys A	1) X - Y = 0.2 N
		Sent 2	Sys B	Sys A	
		Sent 3	Sys B	Sys A	
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys B	Sys A	
		Sent 3	Sys A	Sys B	

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3	
Optimizer Run 1	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	
					<u>Simulated Trials</u> <u>> 0.3?</u>
Optimizer Run 2	{	Sent 1	Sys A	Sys B	1) X - Y = 0.2
		Sent 2	Sys A	Sys B	N
		Sent 3	Sys A	Sys B	
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: $A - B = 0.3$	
Optimizer Run 1	{	Sent 1	Sys B	Sys A	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	
					<u>Simulated Trials</u> <u>$\geq 0.3?$</u>
Optimizer Run 2	{	Sent 1	Sys B	Sys A	1) $X - Y = 0.2$ N
		Sent 2	Sys B	Sys A	2) $X - Y = 0.9$ Y
		Sent 3	Sys B	Sys A	
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys B	Sys A	
		Sent 3	Sys A	Sys B	

Approximate Randomization

- An approximate permutation significance test

		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3	
Optimizer Run 1	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	
					<u>Simulated Trials</u> <u>> 0.3?</u>
Optimizer Run 2	{	Sent 1	Sys A	Sys B	1) X - Y = 0.2 N
		Sent 2	Sys A	Sys B	2) X - Y = 0.9 Y
		Sent 3	Sys A	Sys B	
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	

Approximate Randomization

- An approximate permutation significance test

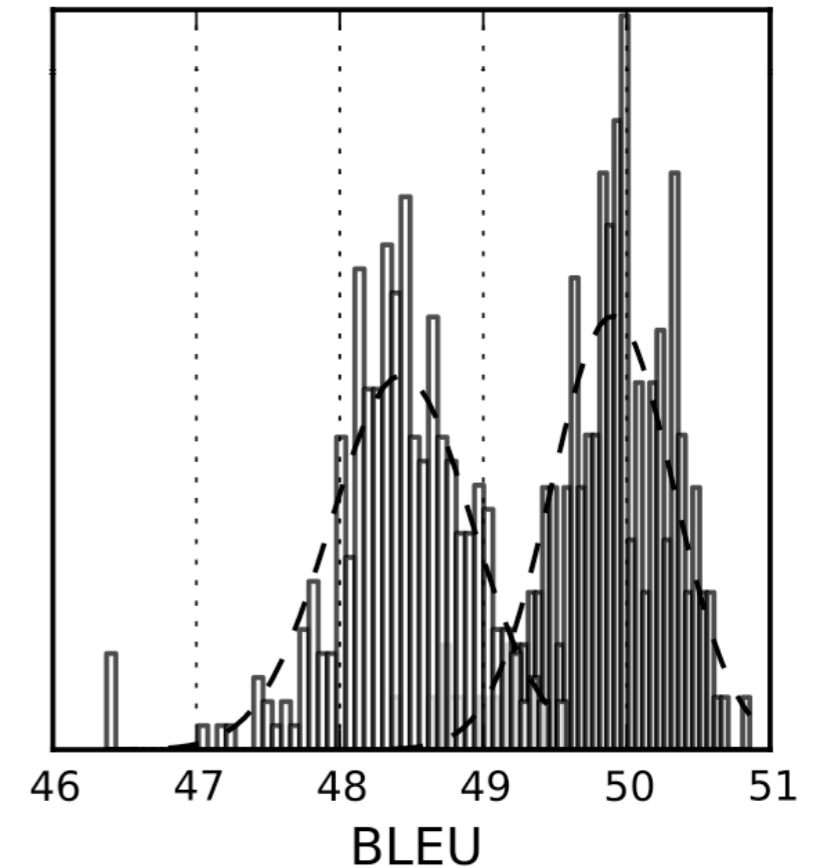
		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3	
Optimizer Run 1	{	Sent 1	Sys B	Sys A	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	
					<u>Simulated Trials</u> <u>> 0.3?</u>
Optimizer Run 2	{	Sent 1	Sys B	Sys A	1) X - Y = 0.2 N
		Sent 2	Sys B	Sys A	2) X - Y = 0.9 Y
		Sent 3	Sys B	Sys A	3) X - Y = 1.0 Y
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys B	Sys A	
		Sent 3	Sys A	Sys B	

Approximate Randomization

- An approximate permutation significance test

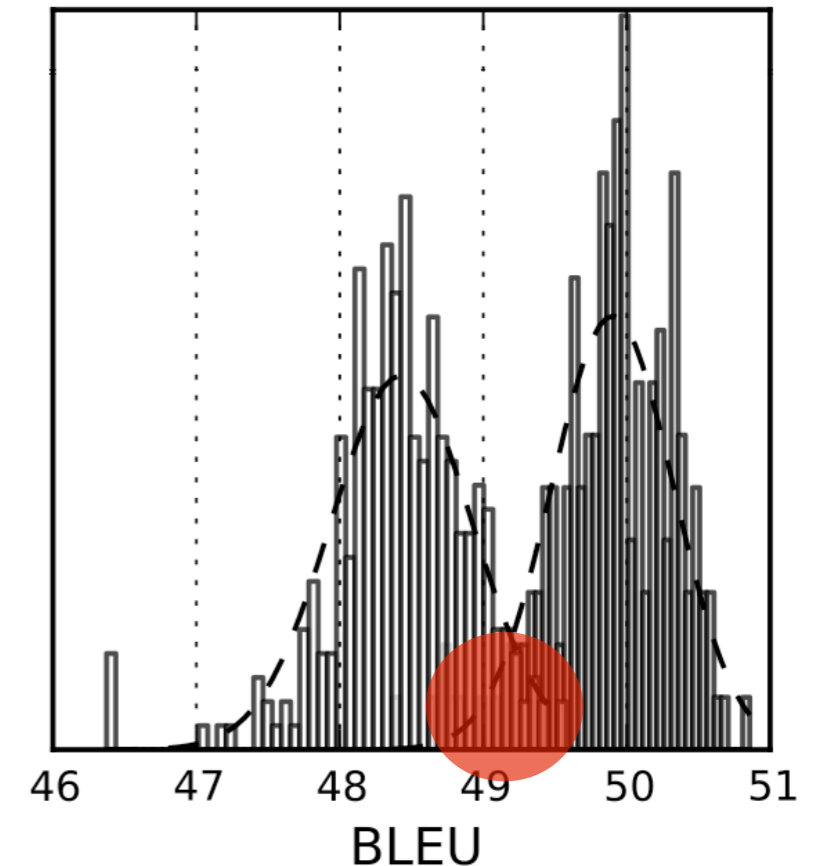
		<u>Set X</u>	<u>Set Y</u>	Observed: A - B = 0.3	
Optimizer Run 1	{	Sent 1	Sys B	Sys A	
		Sent 2	Sys A	Sys B	
		Sent 3	Sys A	Sys B	
				<u>Simulated Trials</u>	<u>> 0.3?</u>
Optimizer Run 2	{	Sent 1	Sys B	Sys A	1) X - Y = 0.2
		Sent 2	Sys B	Sys A	2) X - Y = 0.9
		Sent 3	Sys B	Sys A	3) X - Y = 1.0
Optimizer Run 3	{	Sent 1	Sys A	Sys B	
		Sent 2	Sys B	Sys A	
		Sent 3	Sys A	Sys B	
				p-value: p(by_chance)	0.67

Sig. Test for Multiple Runs



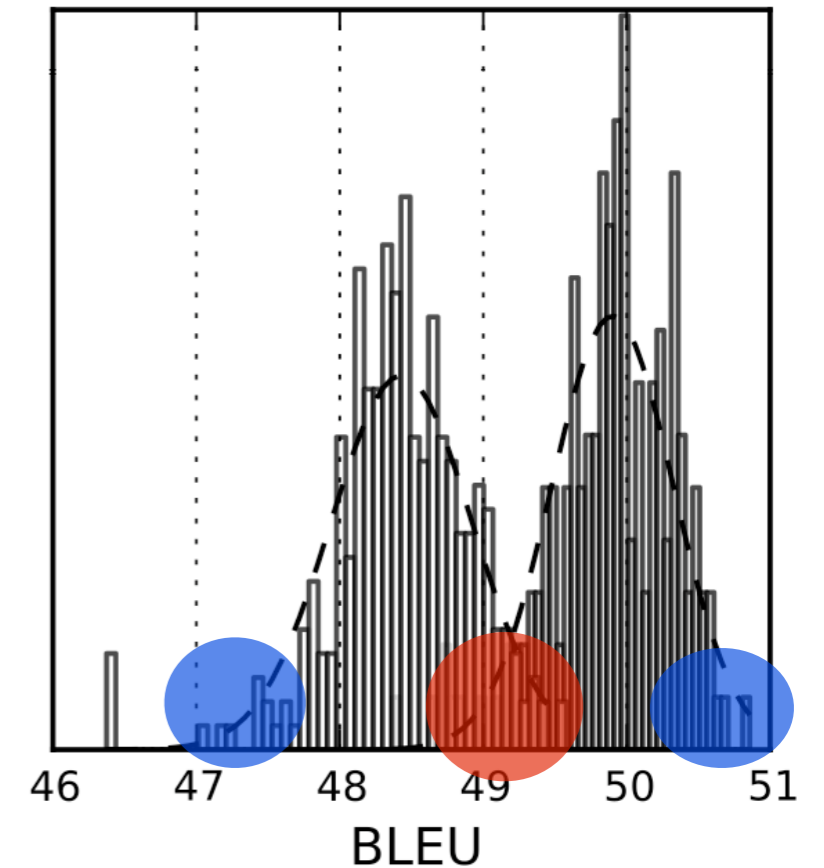
Sig. Test for Multiple Runs

Samples	System A	System B	BTEC <i>p</i> -value
1	high	low	0.25



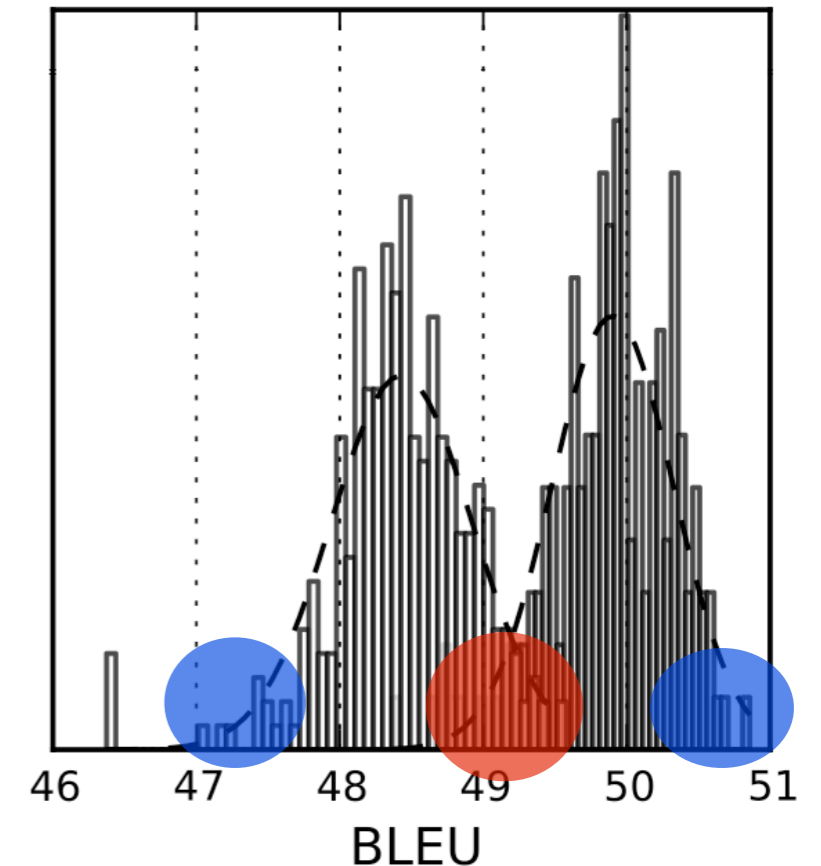
Sig. Test for Multiple Runs

Samples	System A	System B	BTEC <i>p</i> -value
1	high	low	0.25
1	low	high	0.0003



Sig. Test for Multiple Runs

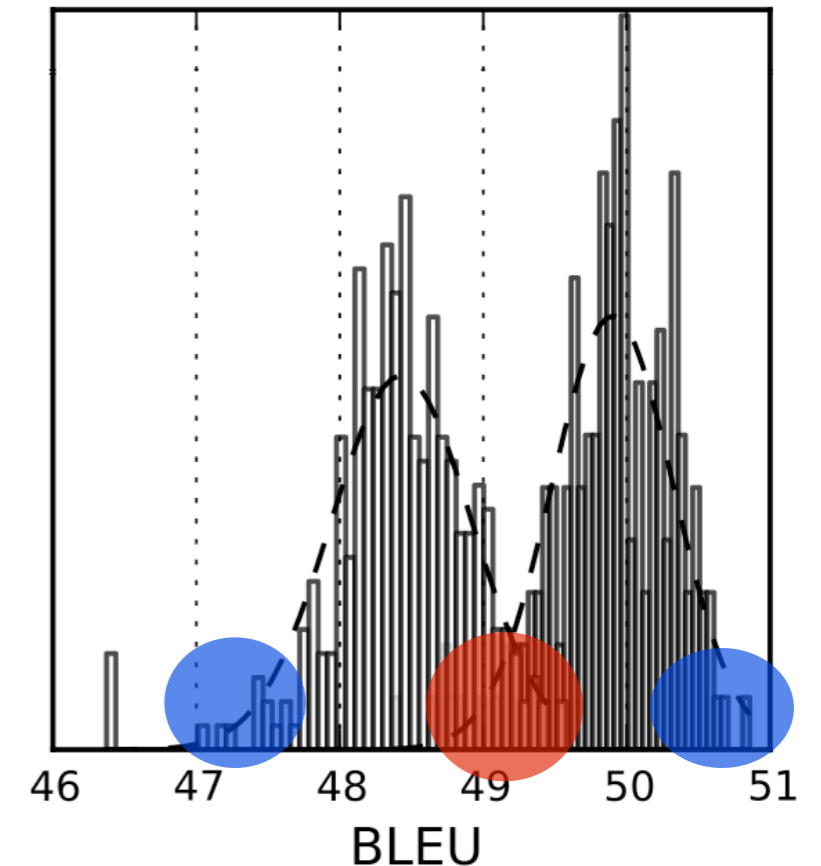
Samples	System A	System B	BTEC p -value
1	high	low	0.25
1	low	high	0.0003



- For small sample sizes, p -values vary wildly, even when the true difference is reasonably large

Sig. Test for Multiple Runs

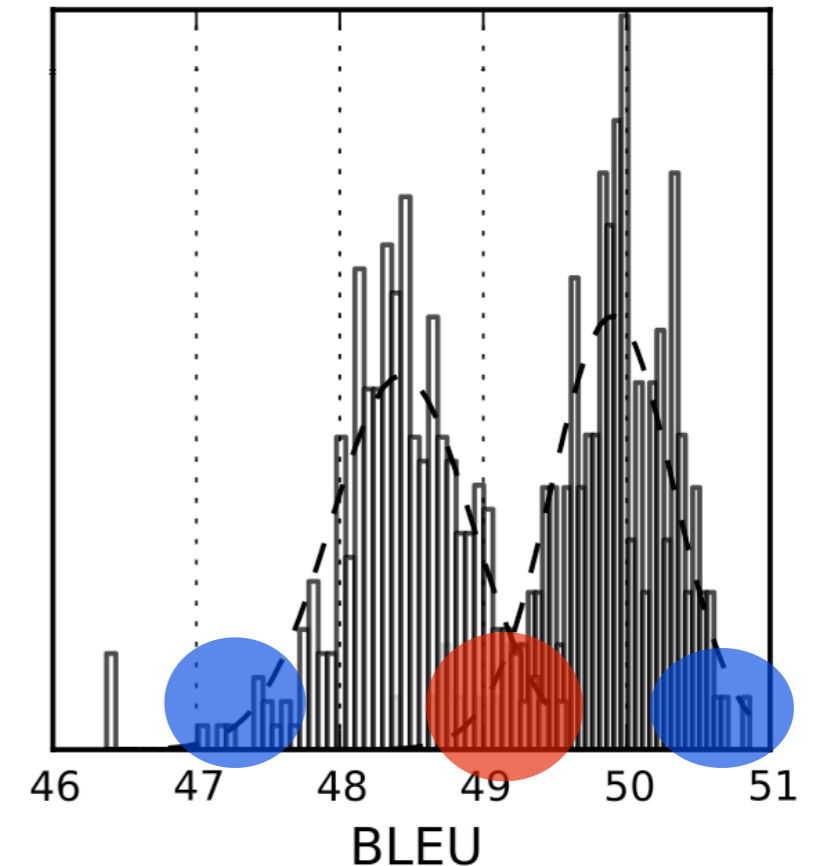
Samples	System A	System B	BTEC <i>p</i> -value
1	high	low	0.25
1	low	high	0.0003
			<i>p</i> -value 95% CI
5	random	random	0.001 - 0.034



- For small sample sizes, *p*-values vary wildly, even when the true difference is reasonably large

Sig. Test for Multiple Runs

Samples	System A	System B	BTEC <i>p</i> -value
1	high	low	0.25
1	low	high	0.0003
			<i>p</i> -value 95% CI
5	random	random	0.001 - 0.034
50	random	random	0.001 - 0.001



- For small sample sizes, *p*-values vary wildly, even when the true difference is reasonably large

Closing Thoughts

- We can't escape all confounding variables, but reporting bad results is a big gamble
- But single samples are bad, so:
 - Run your optimizer 3+ times, decode with each of these weight vectors, and score the results
 - Report average and variance of these scores
 - Report significance w.r.t. multiple runs

Questions?

- Software at www.github.com/jhclark/multeval
- BLEU, METEOR, and TER included
- Variance and p -values
- Runs within seconds
- Plaintext in. LaTeX out.