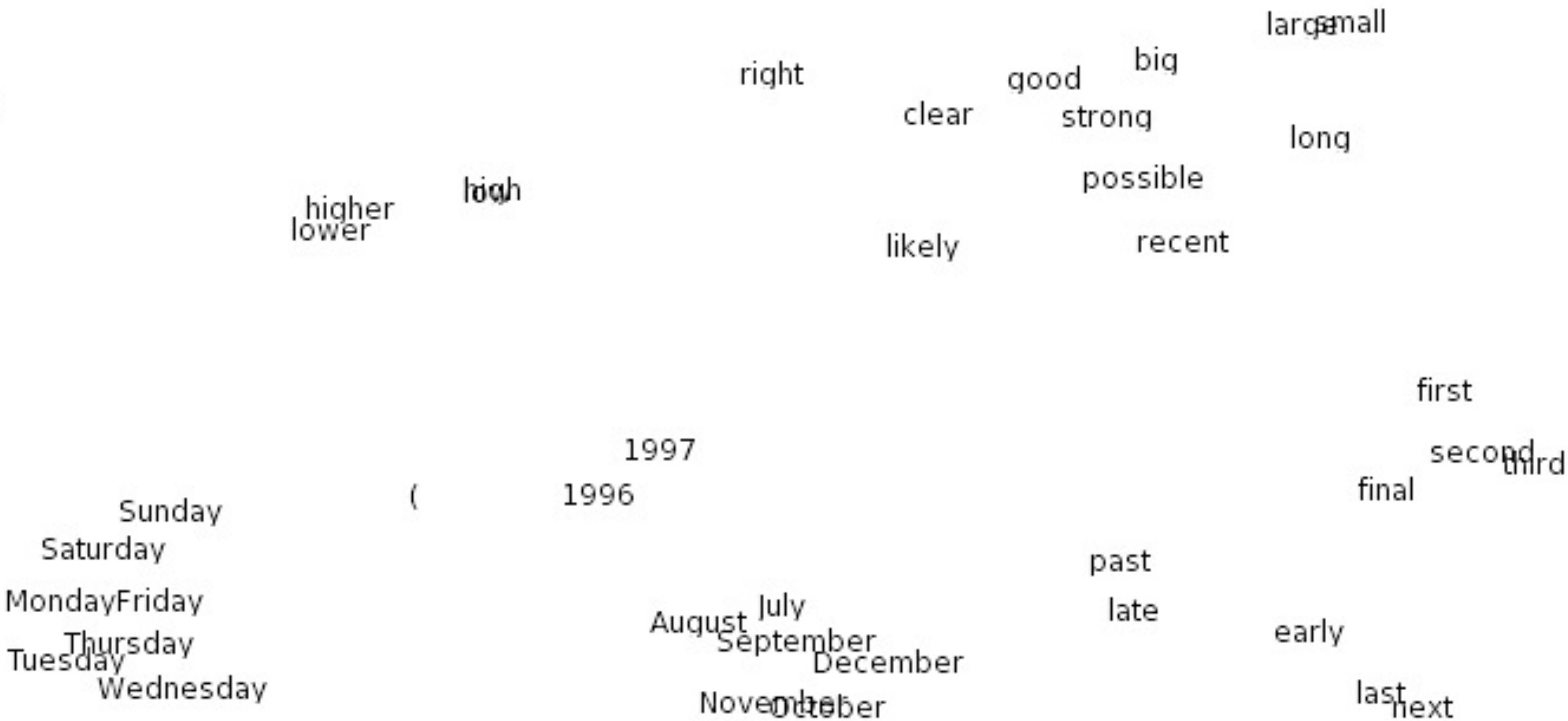
Learning Paraphrastic Representations of Natural Language Sentences

Kevin Gimpel

John Wieting, Mohit Bansal, Karen Livescu

TOYOTA TECHNOLOGICAL INSTITUTE AT CHICAGO

Word Embeddings



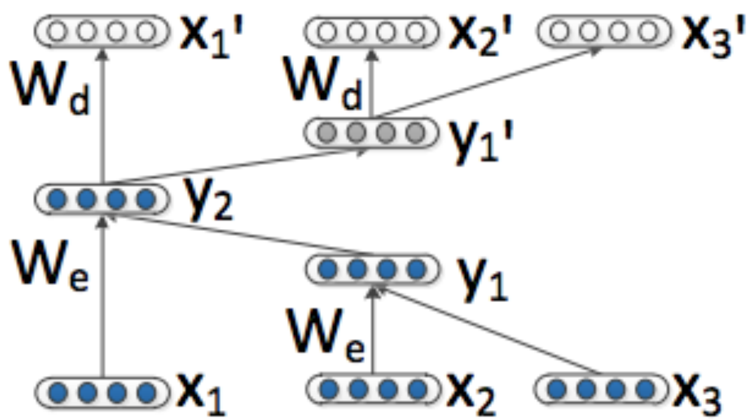
Turian et al. (2010)

- Pretrained word embeddings are really useful!

- What about pretrained embeddings for phrases and sentences?

Recursive Neural Net Autoencoders

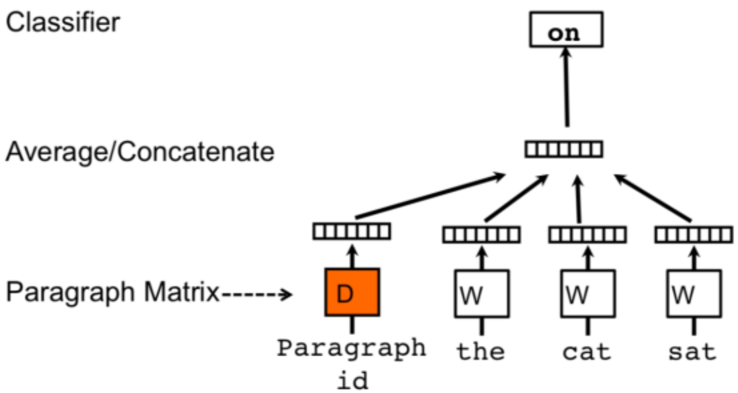
- composition based on syntactic parse

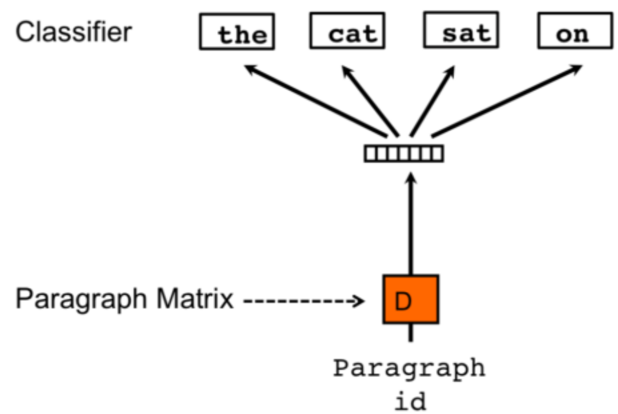


Socher, Huang, Pennington, Ng, Manning (2011)

Paragraph Vectors

- Represent sentence (or paragraph) by predicting its own words or context words

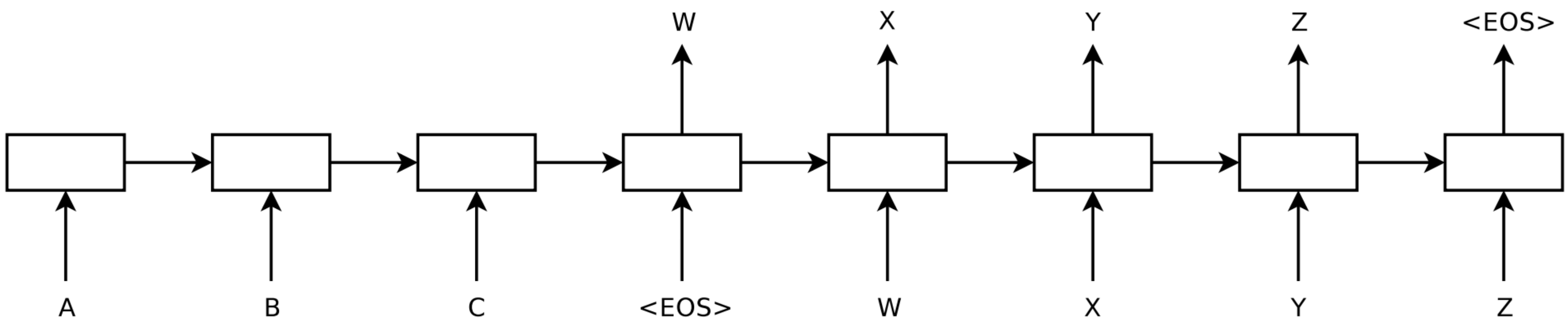




Le & Mikolov (2014)

Neural Machine Translation

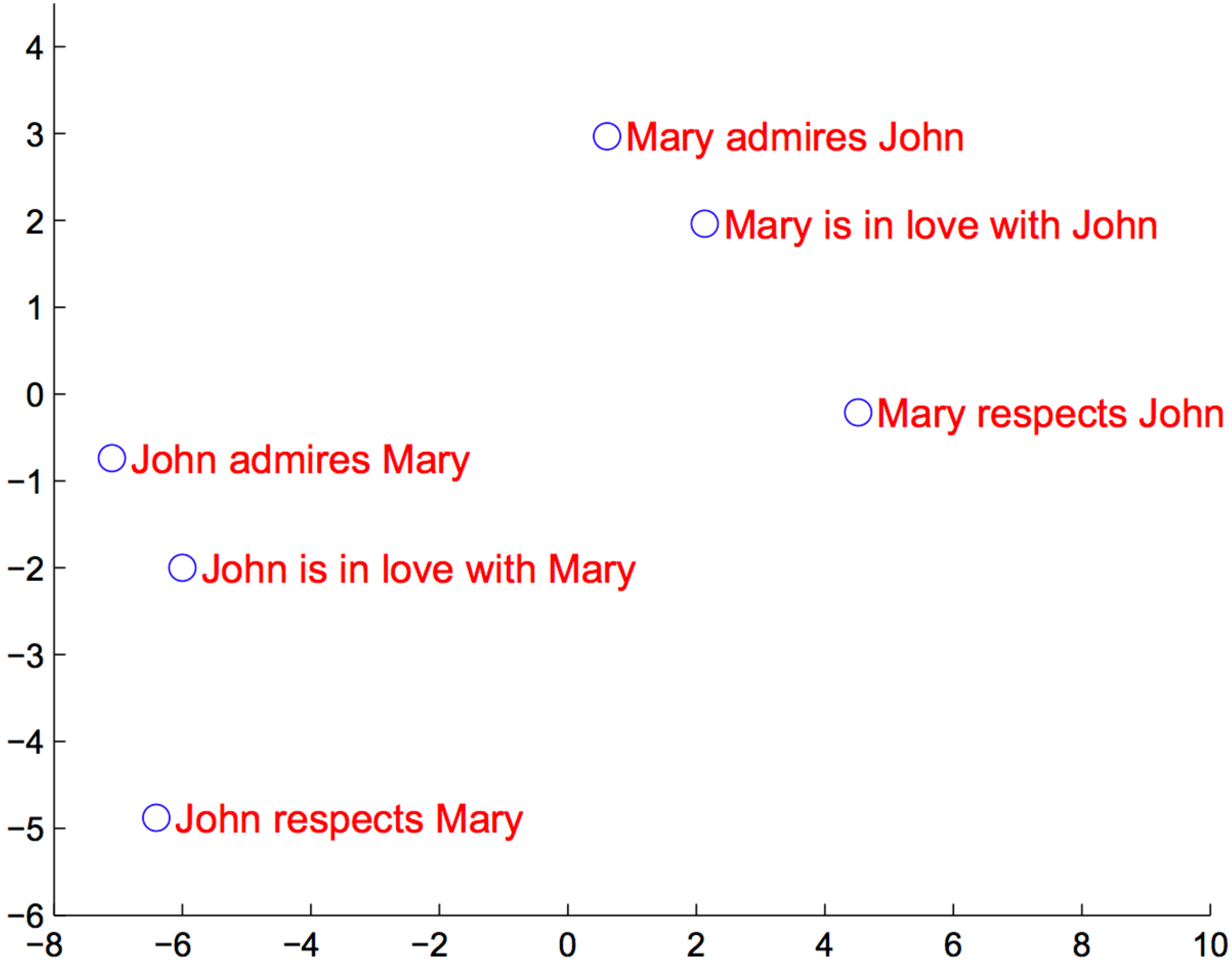
- Encode source sentence, decode translation



Sutskever, Vinyals, Le (2014)

Cho, van Merrienboer, Gulcehre, Bahdanau, Bougares, Schwenk, Bengio (2014)

Encoder as a Sentence Embedding Model?

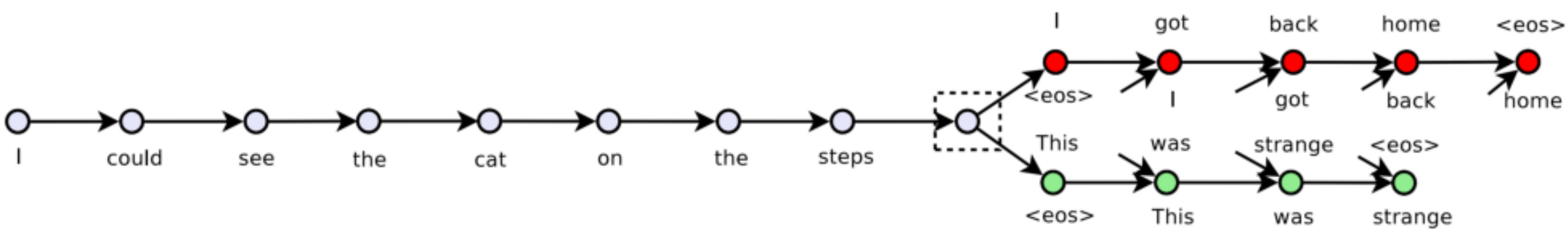


Sutskever, Vinyals, Le (2014)

Skip-Thoughts

- Encode sentence, decode neighboring sentences

...I got back home   I could see the cat on the steps   This was strange ...



Kiros, Zhu, Salakhutdinov, Zemel, Torralba, Urtasun, Fidler (2015)

Skip-Thoughts

query sentence:

im sure youll have a glamorous evening , she said , giving an exaggerated wink .

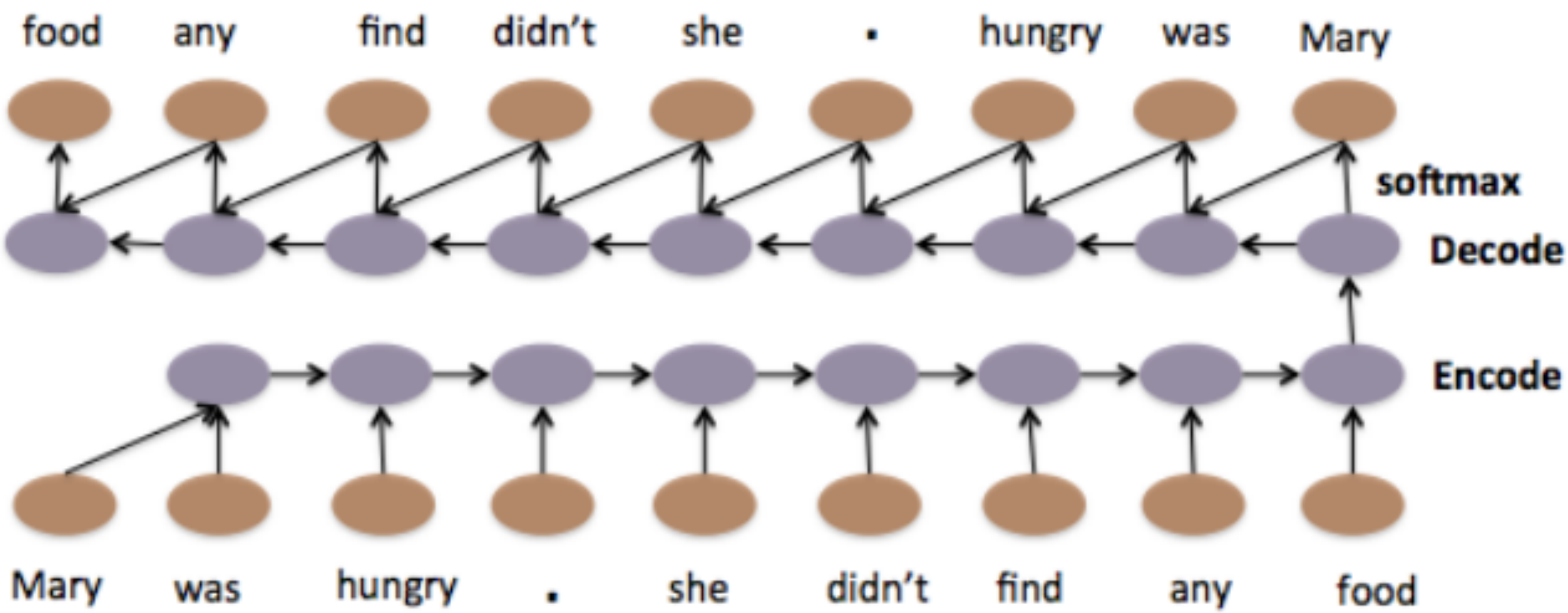
nearest neighbor:

im really glad you came to the party tonight , he said , turning to her .

Kiros, Zhu, Salakhutdinov, Zemel, Torralba, Urtasun, Fidler (2015)

LSTM Autoencoders

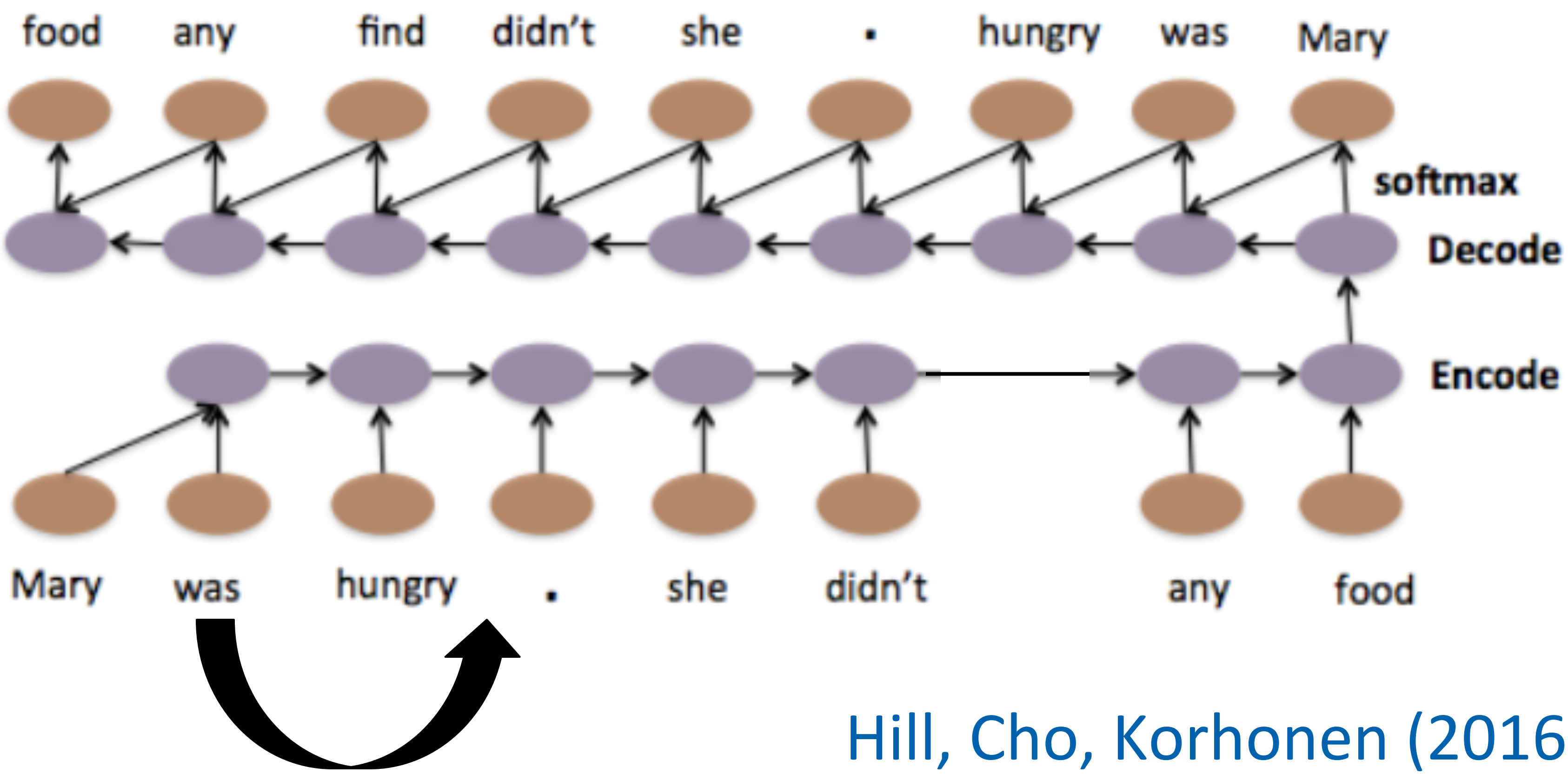
- Encode sentence, decode sentence



Li, Luong, Jurafsky (2015)

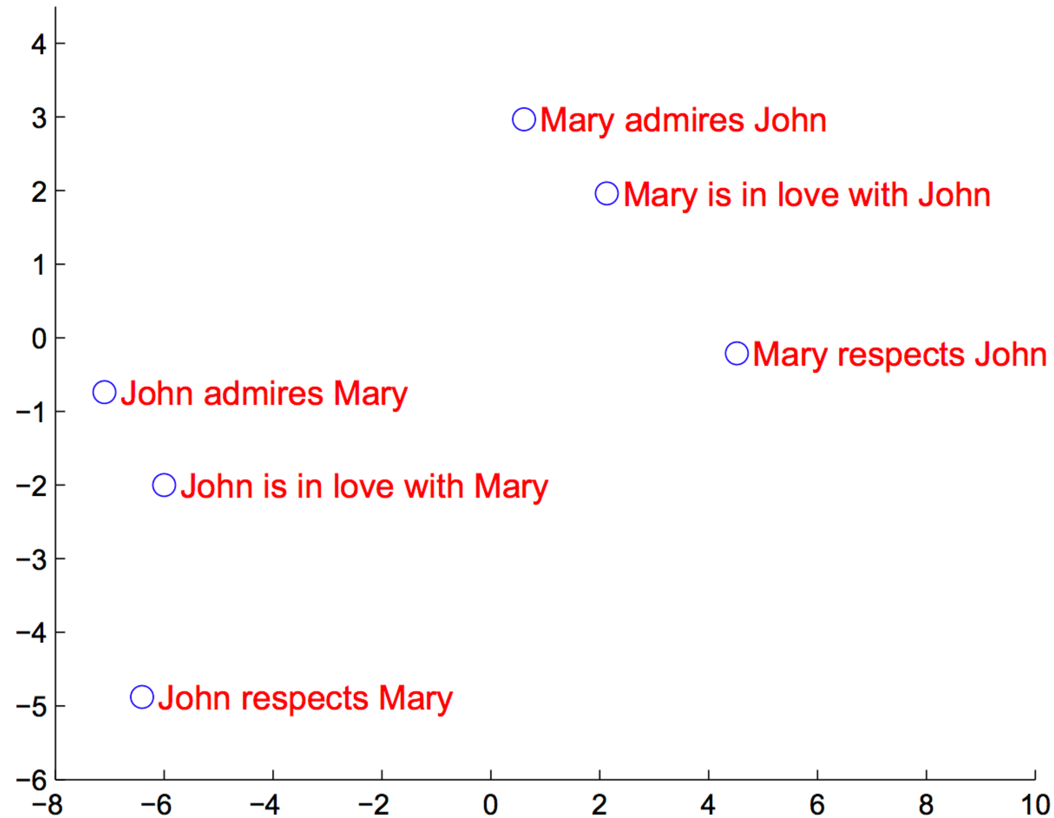
LSTM Denoising Autoencoders

- Encode "corrupted" sentence, decode sentence



Hill, Cho, Korhonen (2016)

Learning **Paraphrastic** Representations of Natural Language Sentences



- How are paraphrastic sentence embeddings useful?
  - multi-document summarization
  - automatic essay grading
  - evaluation of text generation systems
  - machine translation
  - entailment/inference

Evaluation: Semantic Textual Similarity (STS)

Other ways are needed.

We must find other ways.

4.4

I absolutely do believe there was an iceberg in those waters.

I don't believe there was any iceberg at all anywhere near the Titanic.

1.2

We evaluate on 22 datasets from many domains:

web forum posts, tweets, machine translation output, news, headlines, definition glosses, image and video captions, etc.

Evaluation

Sentence Embedding Model	**STS Pearson x 100**
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38

Hill, Cho, Korhonen (2016)
Wieting, Bansal, G, Livescu (2016)

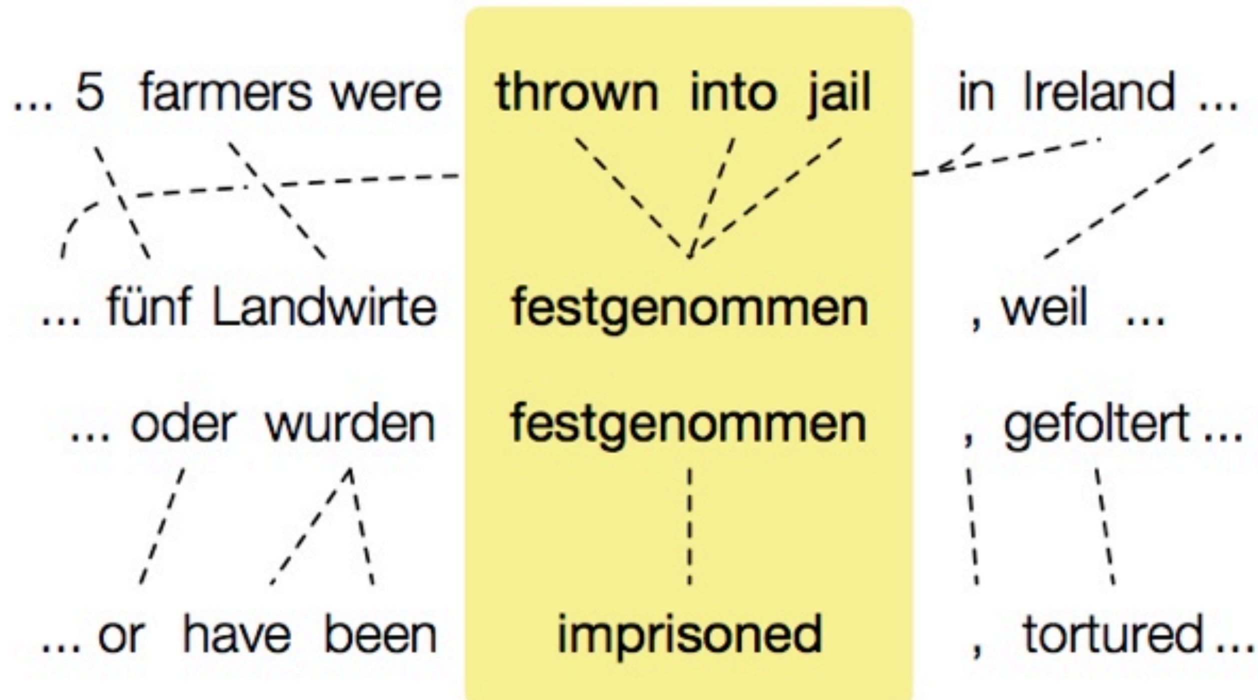
Evaluation

Sentence Embedding Model	STS Pearson x 100
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38
FastSent (bag of words)	64
Avg. pretrained word embeddings	65

Hill, Cho, Korhonen (2016)
Wieting, Bansal, G, Livescu (2016)

Paraphrase Database (PPDB)

(Ganitkevitch, Van Durme, and Callison-Burch, 2013)



credit: Chris Callison-Burch

Training Data: phrase pairs from PPDB

good	great
be given the opportunity to	have the possibility of
i can hardly hear you .	you 're breaking up .
and the establishment	as well as the development
laying the foundations	pave the way
making every effort	to do its utmost
…	…

tens of millions more!

Learning

Goal: Learn sentence embedding function $g_\theta(x)$

For now, it's just word averaging:

$$g_\theta(x) = \frac{1}{|x|} \sum_i \mathrm{embedding}_\theta(x_i)$$

Learning

$$\min_{\theta} \sum_{\langle u,v \rangle \in \mathrm{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

Goal: Learn sentence embedding function $g_\theta(x)$

Learning

$$\min_{\theta} \sum_{\langle u,v\rangle \in \text{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

sum over
paraphrase pairs

Learning

$$\min_{\theta} \sum_{\langle u,v \rangle \in \text{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

sum over
paraphrase pairs

negative
example

Learning

$$\min_{\theta} \sum_{\langle u,v\rangle \in \text{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

negative
example

$$t = \operatorname*{argmax}_{s:\langle\cdot,s\rangle\in\text{batch}, s\neq v} \cos(g_\theta(u), g_\theta(s))$$

Learning

$$\min_{\theta} \sum_{\langle u,v\rangle \in \mathrm{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

negative example

$$t = \operatorname*{argmax}_{s:\langle \cdot, s\rangle \in \mathrm{batch}, s \neq v} \cos(g_\theta(u), g_\theta(s))$$

only do argmax over current mini-batch (for efficiency)

Learning

$$\min_{\theta} \sum_{\langle u,v\rangle \in \mathrm{Train}} \left[\Delta - \cos\left(g_\theta(u), g_\theta(v)\right) + \cos(g_\theta(u), g_\theta(t))\right]_+$$

we regularize by penalizing squared L_2 distance to initial (pretrained GloVe) embeddings

Evaluation

Sentence Embedding Model	STS Pearson x 100
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38
FastSent (bag of words)	64
Avg. pretrained word embeddings	65

Hill, Cho, Korhonen (2016)
Wieting, Bansal, G, Livescu (2016)

Evaluation

Sentence Embedding Model	**STS Pearson x 100**
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38
FastSent (bag of words)	64
Avg. pretrained word embeddings	65
Ours (avg. trained on PPDB)	71

Hill, Cho, Korhonen (2016)
Wieting, Bansal, G, Livescu (2016)

Word averaging throws away word order!

How about an LSTM?

Evaluation

Sentence Embedding Model	**STS Pearson x 100**
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38
FastSent (bag of words)	64
Avg. pretrained word embeddings	65
Ours (avg. trained on PPDB)	71

Evaluation

Sentence Embedding Model	STS Pearson x 100
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM Denoising Autoencoder	38
FastSent (bag of words)	64
Avg. pretrained word embeddings	65
Ours (avg. trained on PPDB)	71
Ours (LSTM trained on PPDB)	52

Evaluation

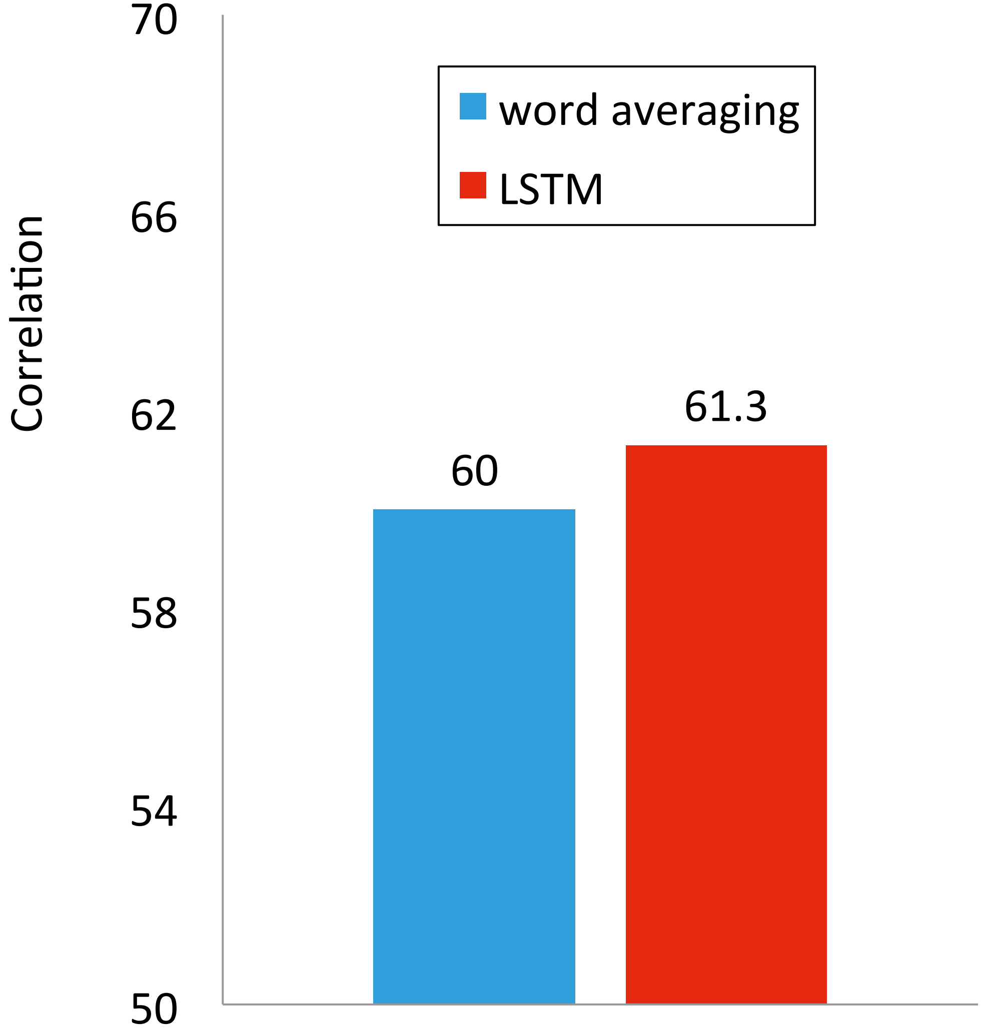
Sentence Embedding Model	**STS Pearson x 100**
Paragraph Vector	44
Neural MT Encoder	42
Skip Thought	31
LSTM Autoencoder	43
LSTM De	
FastSent	
Avg. pret	
Ours (avg. trained on PPDB)	71
Ours (LSTM trained on PPDB)	52

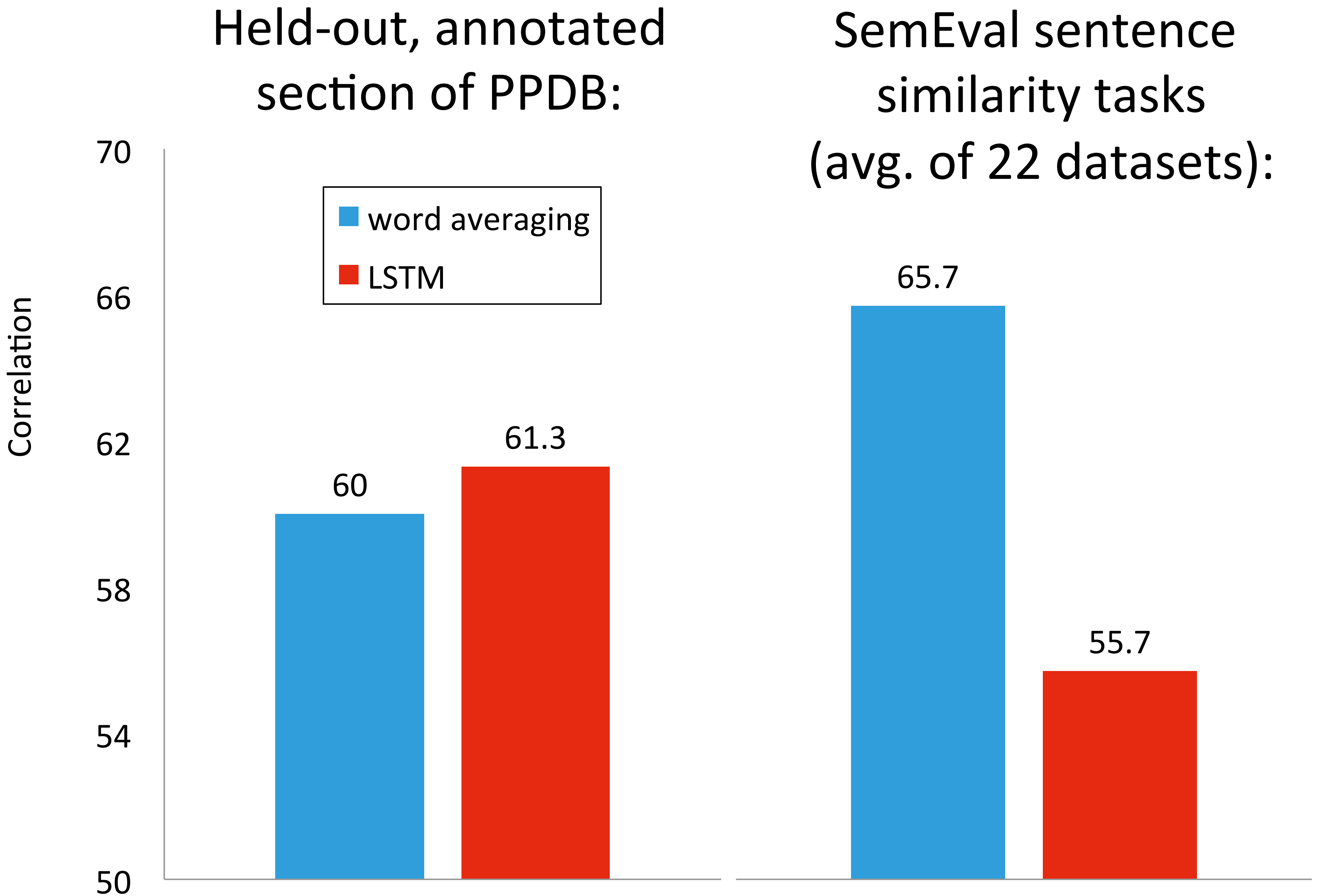
What's going on here?

In-Domain Evaluation:
held-out, annotated PPDB pairs

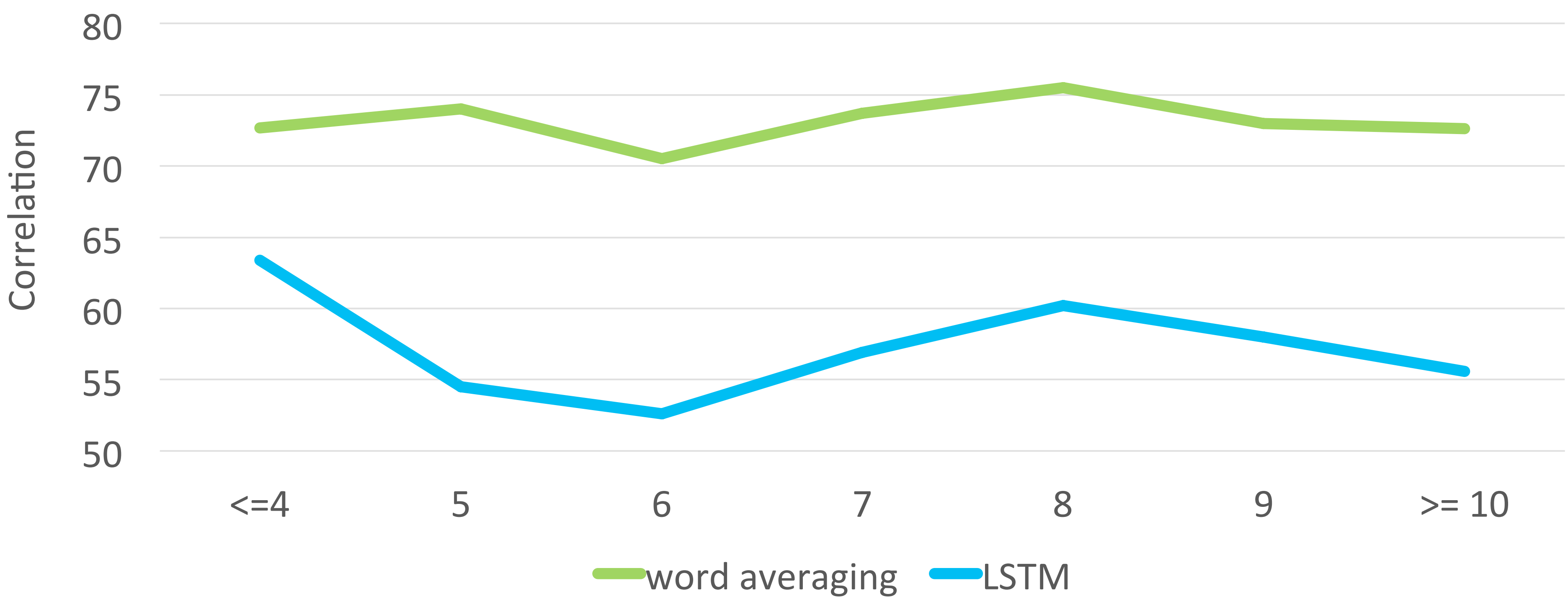
		Similarity Annotation
can not be separated from	is inseparable from	5.0
hoped to be able to	looked forward to	3.4
come on , think about it	people , please	2.2
how do you mean that	what worst feelings	1.6

Held-out, annotated section of PPDB:




Held-out, annotated section of PPDB:
SemEval sentence similarity tasks (avg. of 22 datasets):
Correlation
70
66
62
58
54
50
word averaging
LSTM
60
61.3
65.7
55.7

Sentence Length Comparison



word averaging is better at all sentence lengths in test data

- This is troubling
- Why does the LSTM struggle on out-of-domain data?

Maybe the problem is the training data…

- New data: sentence pairs automatically extracted by Coster and Kauchak (2011)

WIKIPEDIA
The Free Encyclopedia

WIKIPEDIA
Simple English

- Developed for text simplification applications; we use it as a paraphrase training set!

New Data: Examples

this was also true for pompeii , where the temple of jupiter that was already there was enlarged and made more roman when the romans took over .

this held true for pompeii , where the previously existing temple of jupiter was enlarged and romanized upon conquest .

New Data: Examples

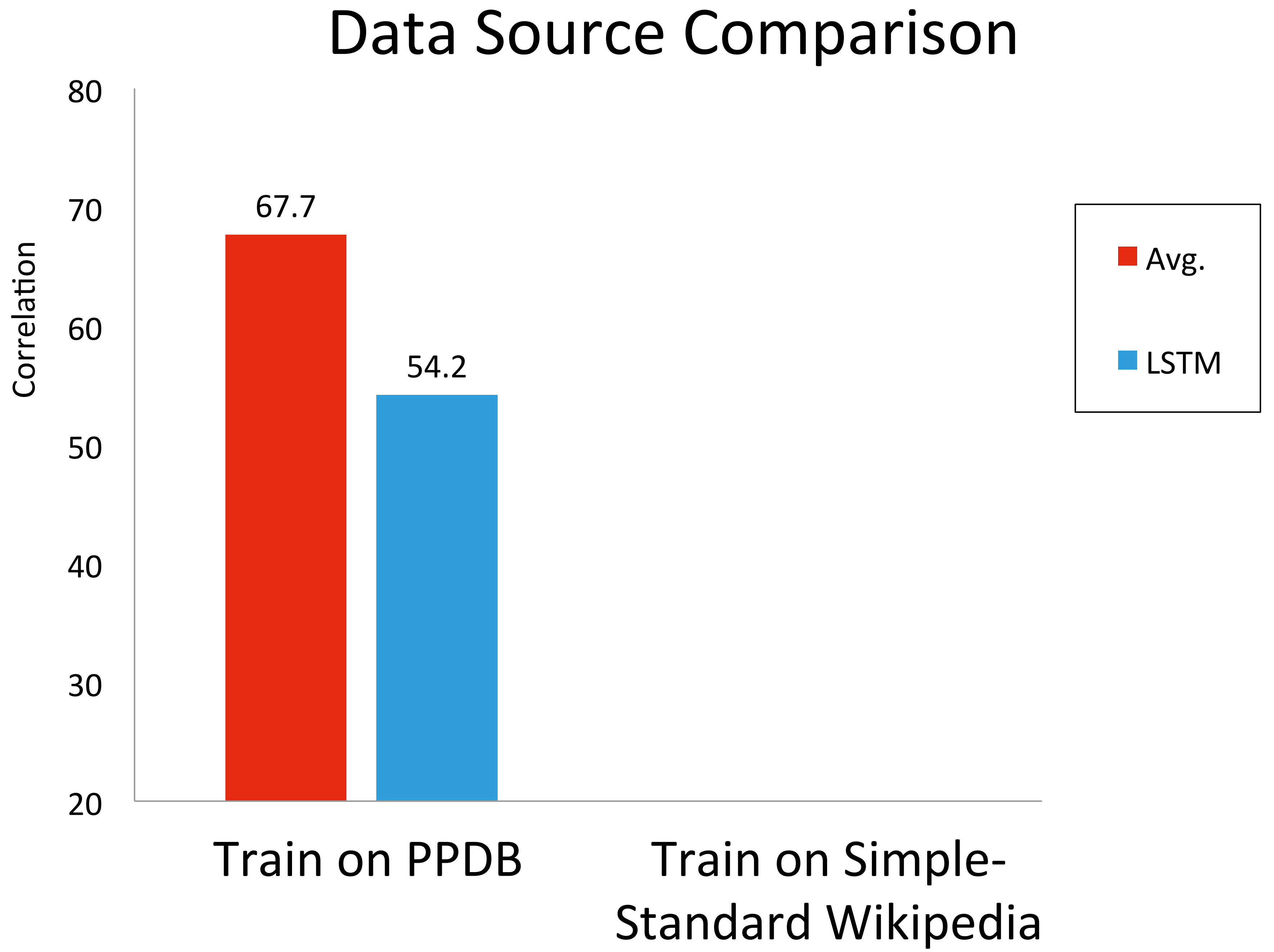
this was also true for pompeii , where the temple of jupiter that was already there was enlarged and made more roman when the romans took over .

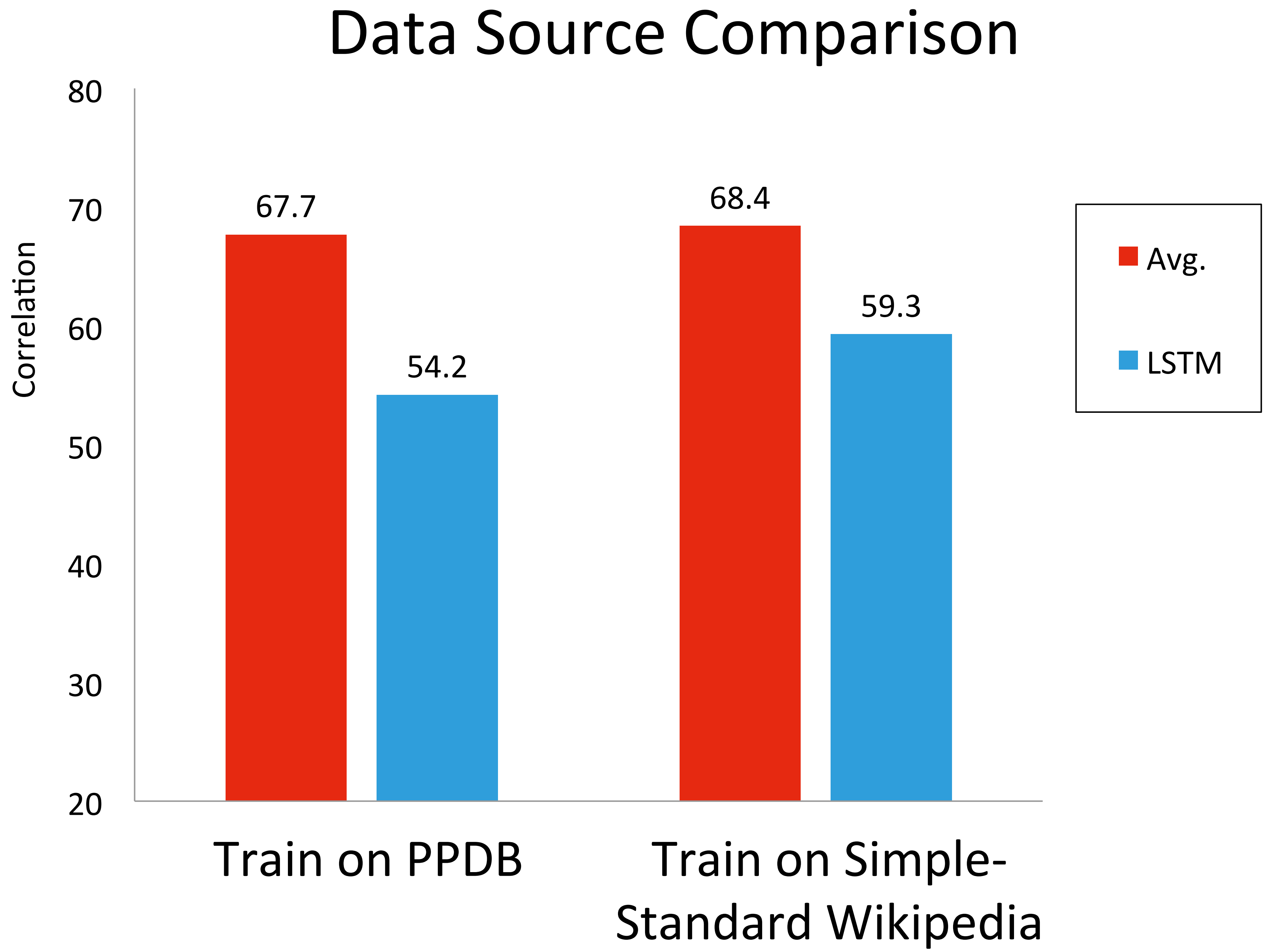
this held true for pompeii , where the previously existing temple of jupiter was enlarged and romanized upon conquest .

New Data: Examples

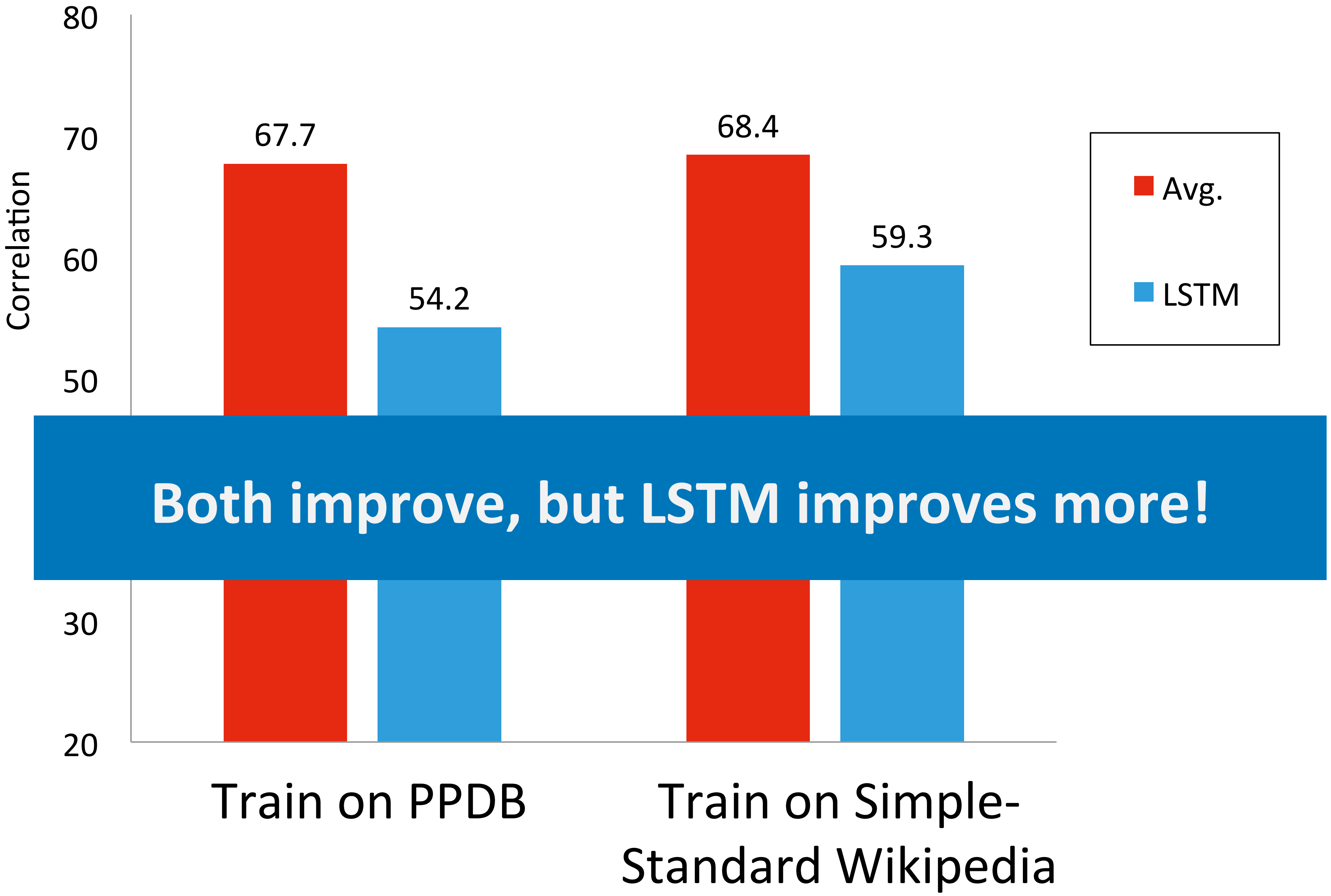
two days later leo crowned charlemagne at st. peter 's tomb .

two days later , on christmas day 800 , leo crowned charlemagne as roman emperor .


Data Source Comparison
Correlation
80
70
60
50
40
30
20
67.7
54.2
Avg.
LSTM
Train on PPDB
Train on Simple-Standard Wikipedia


Data Source Comparison
Correlation
80
70
60
50
40
30
20
67.7
54.2
68.4
59.3
Avg.
LSTM
Train on PPDB
Train on Simple-Standard Wikipedia

Data Source Comparison



Maybe the LSTM is just memorizing the training sequences…

Scrambling

- with some probability, scramble both sentences:

originally , the college was just for boys from eton college .

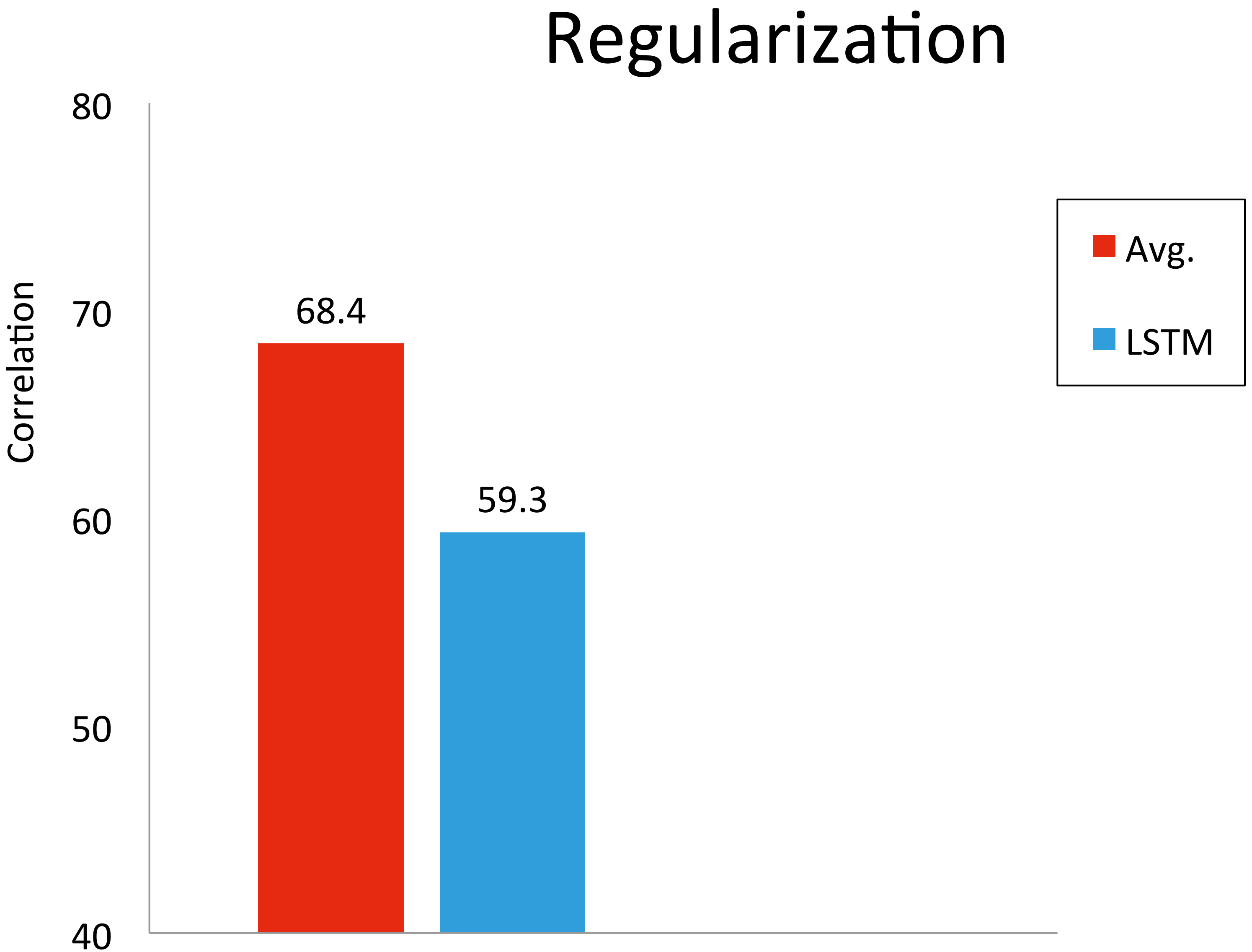
originally , the college was to be specifically for boys from eton college .

↓

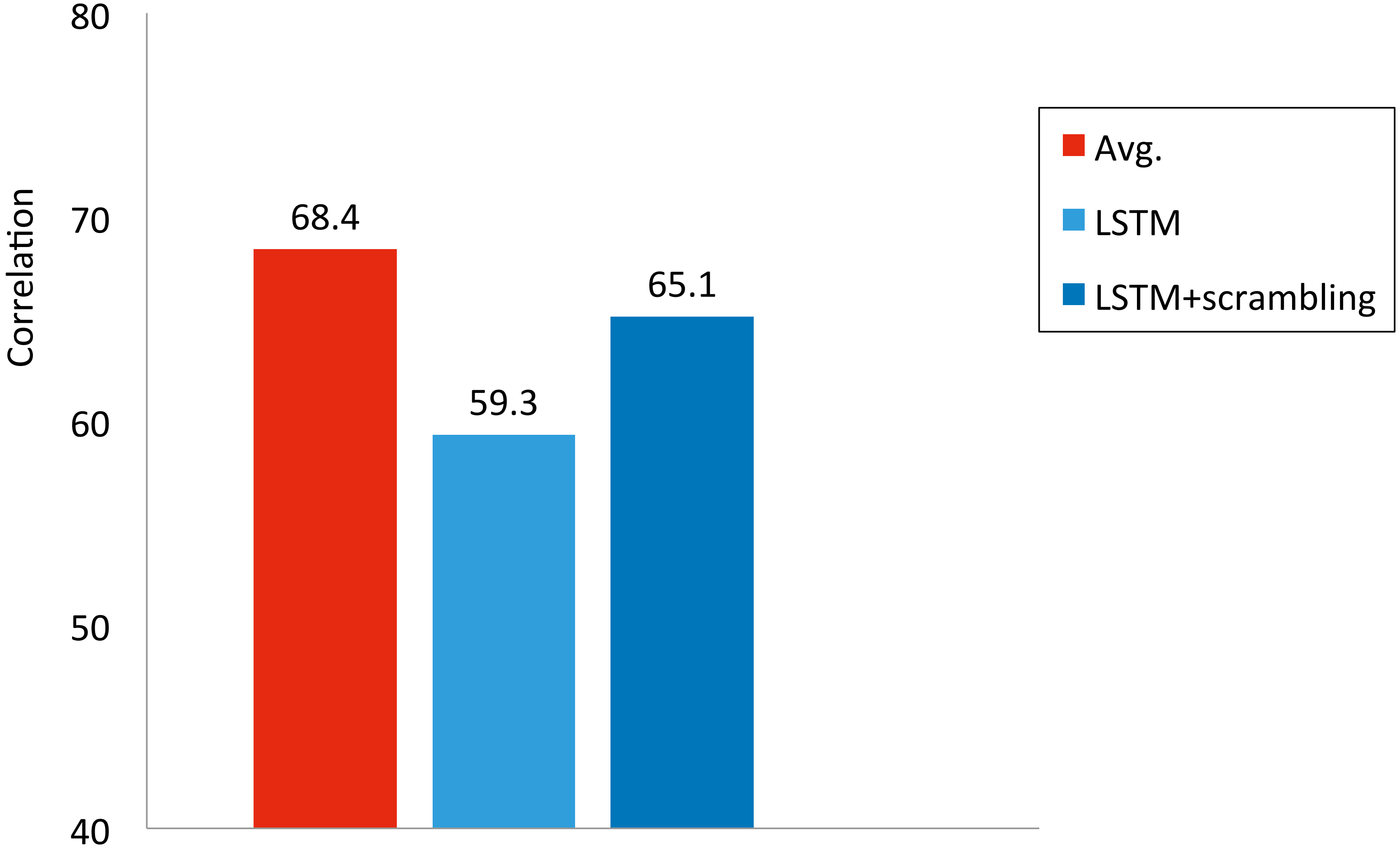
just was boys originally from , . for eton college college the

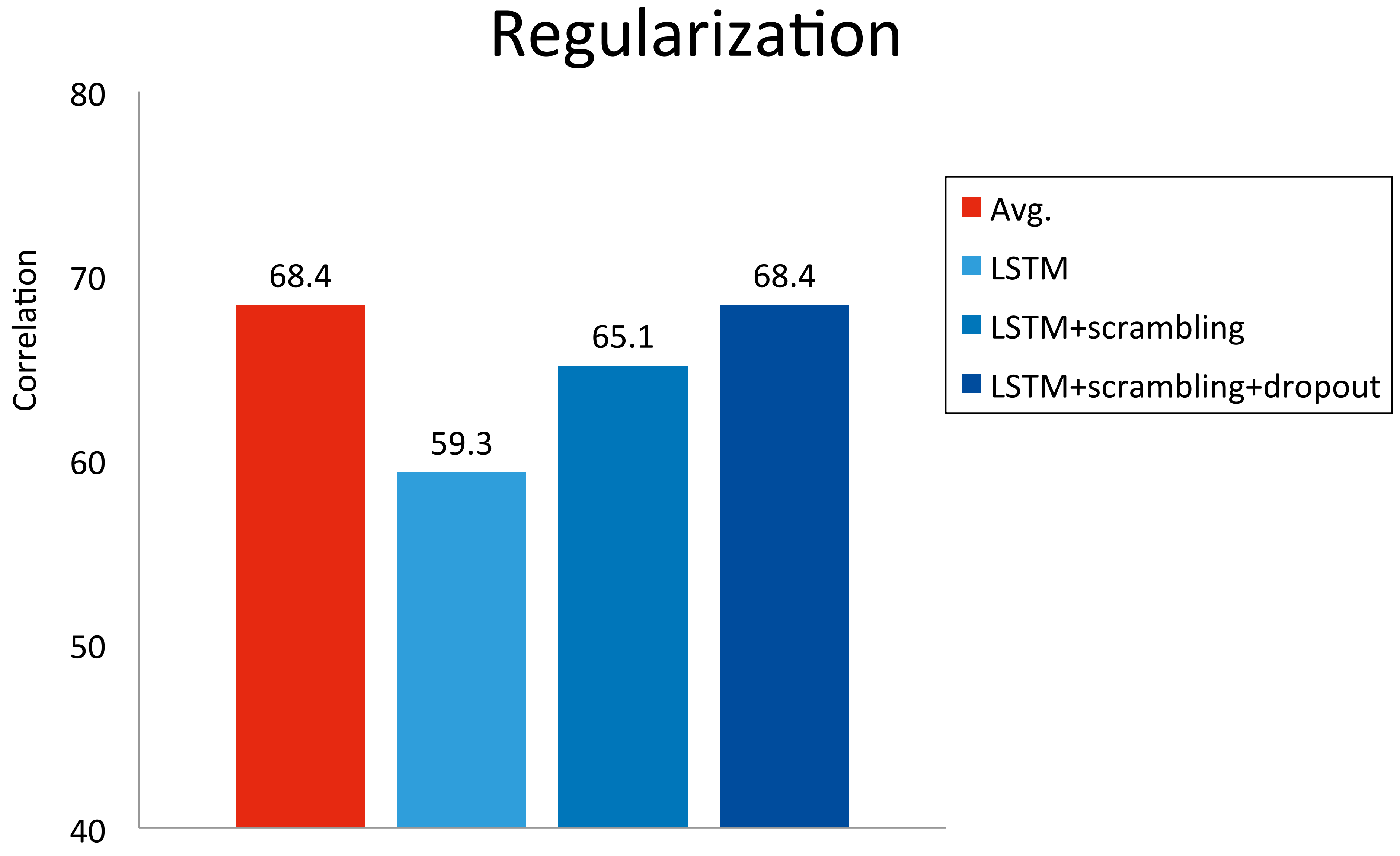
the college eton . to specifically boys was , from be originally for college

- scrambling rate tuned over {0.25, 0.5, 0.75}

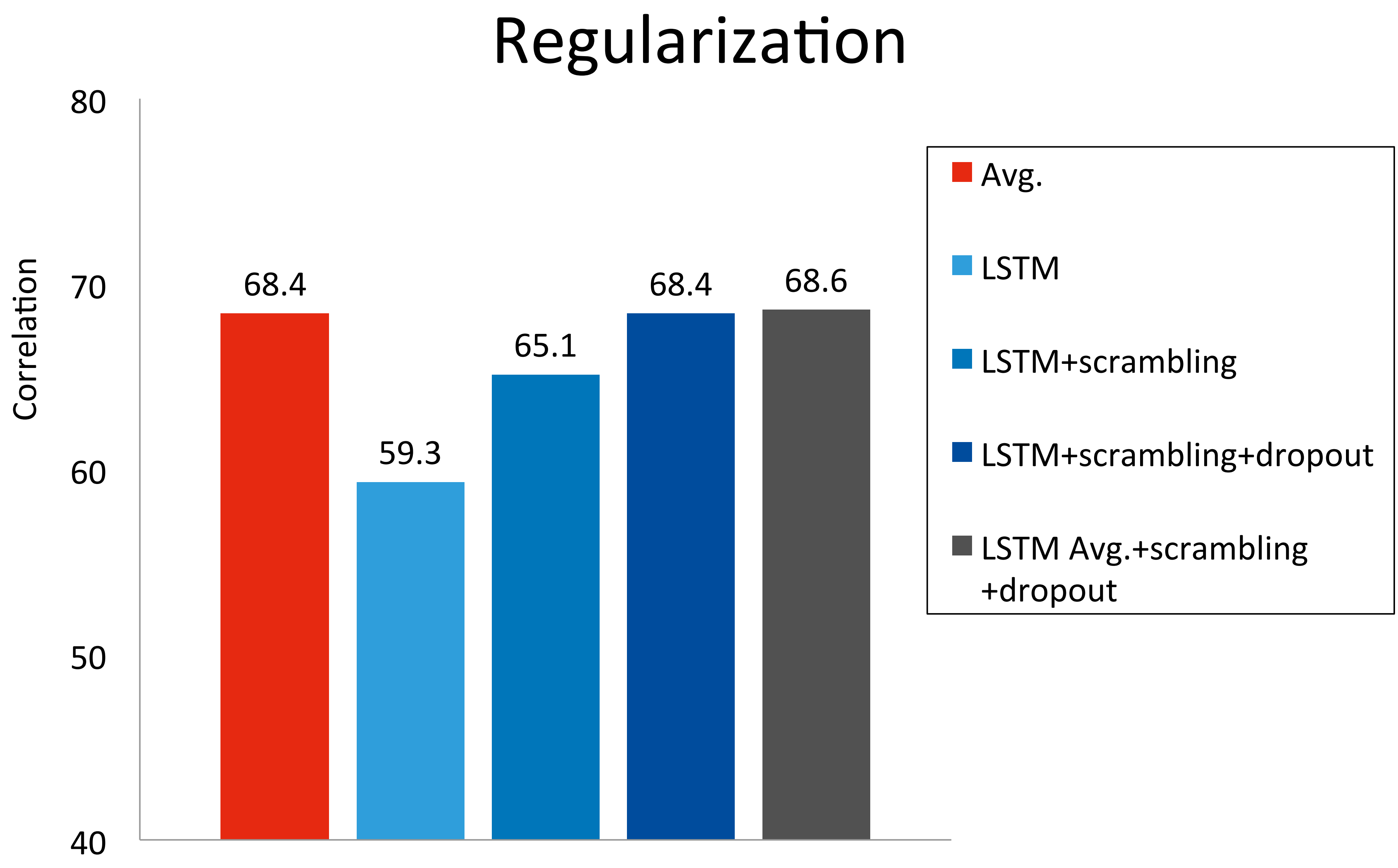

Regularization
Correlation
80
70
60
50
40
68.4
59.3
Avg.
LSTM

Regularization





Regularization
Correlation
80
70
60
50
40
68.4
59.3
65.1
68.4
Avg.
LSTM
LSTM+scrambling
LSTM+scrambling+dropout


Regularization
80
70
60
50
40
Correlation
68.4
59.3
65.1
68.4
68.6
Avg.
LSTM
LSTM+scrambling
LSTM+scrambling+dropout
LSTM Avg.+scrambling
+dropout

LSTM is better than averaging:

sentence 1	sentence 2	LSTM sim.	Avg. sim.	Gold sim.
bloomberg chips in a billion	bloomberg gives $1.1 b to university	3.99	3.04	4.0
in other regions, the sharia is imposed.	in other areas, sharia law is being introduced by force.	4.44	3.72	4.75

word averaging underestimates similarity when there are multiword paraphrases:

"chips in" = "gives"
"a billion" = "$1.1 b"
"the sharia" = "sharia law"
"imposed" = "being introduced by force"

LSTM overestimates similarity:

sentence 1	**sentence 2**	**LSTM sim.**	**Avg. sim.**	**Gold sim.**
three men in suits sitting at a table.	two women in the kitchen looking at a object.	3.33	2.79	0.0
we never got out of it in the first place!	where does the money come from in the first place?	4.00	3.33	0.8
two birds interacting in the grass.	two dogs play with each other outdoors.	3.44	2.81	0.2

LSTM overestimates similarity with similar sequences of syntactic categories, but different meanings

Gated Recurrent Averaging Network (GRAN)

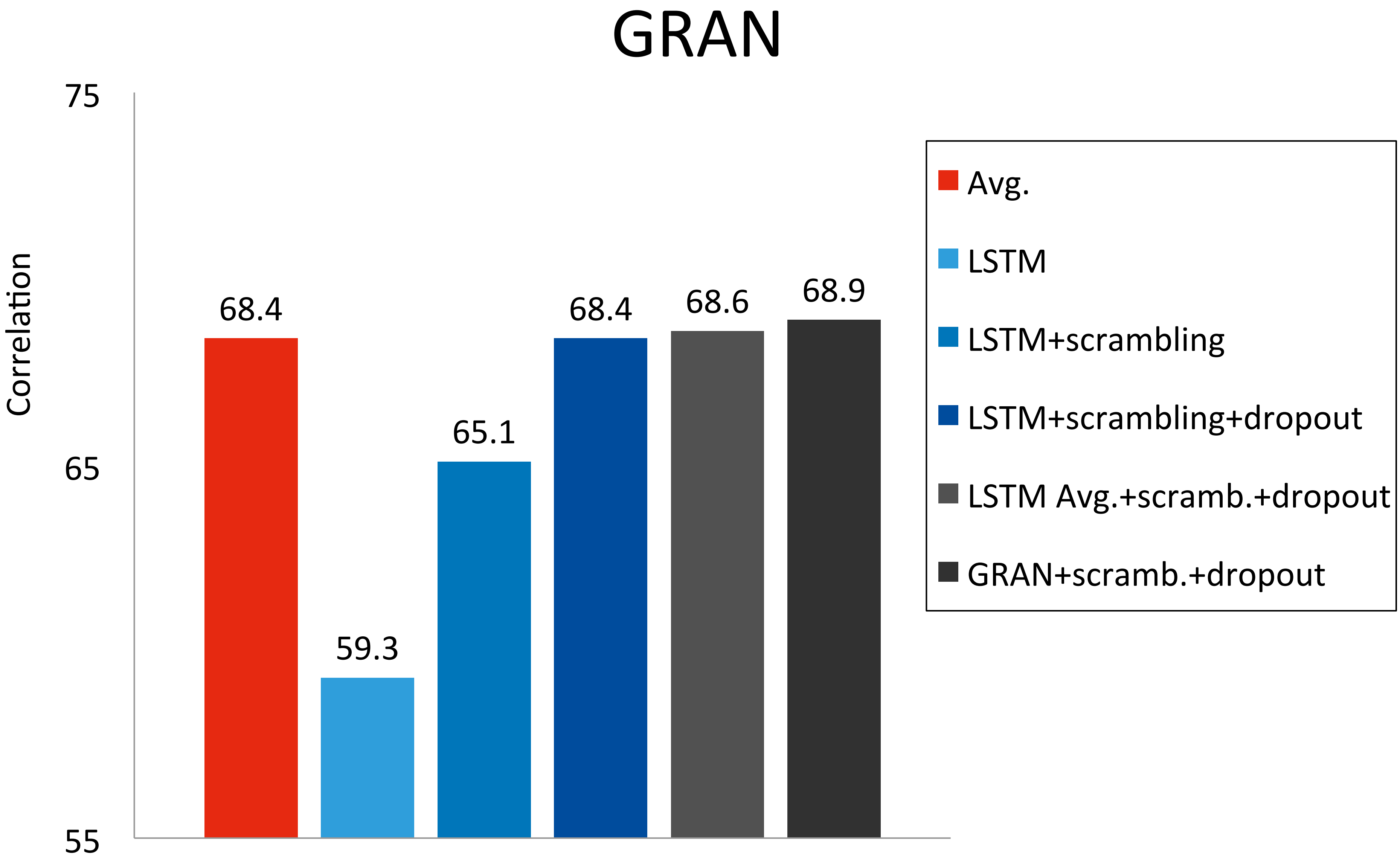
- Inspired by the success of averaging and the LSTM, we propose a new model:

$$a_t = x_t \odot \sigma\left(W_x x_t + W_h h_t + b\right)$$

embedding of word at position t

LSTM hidden vector at position t

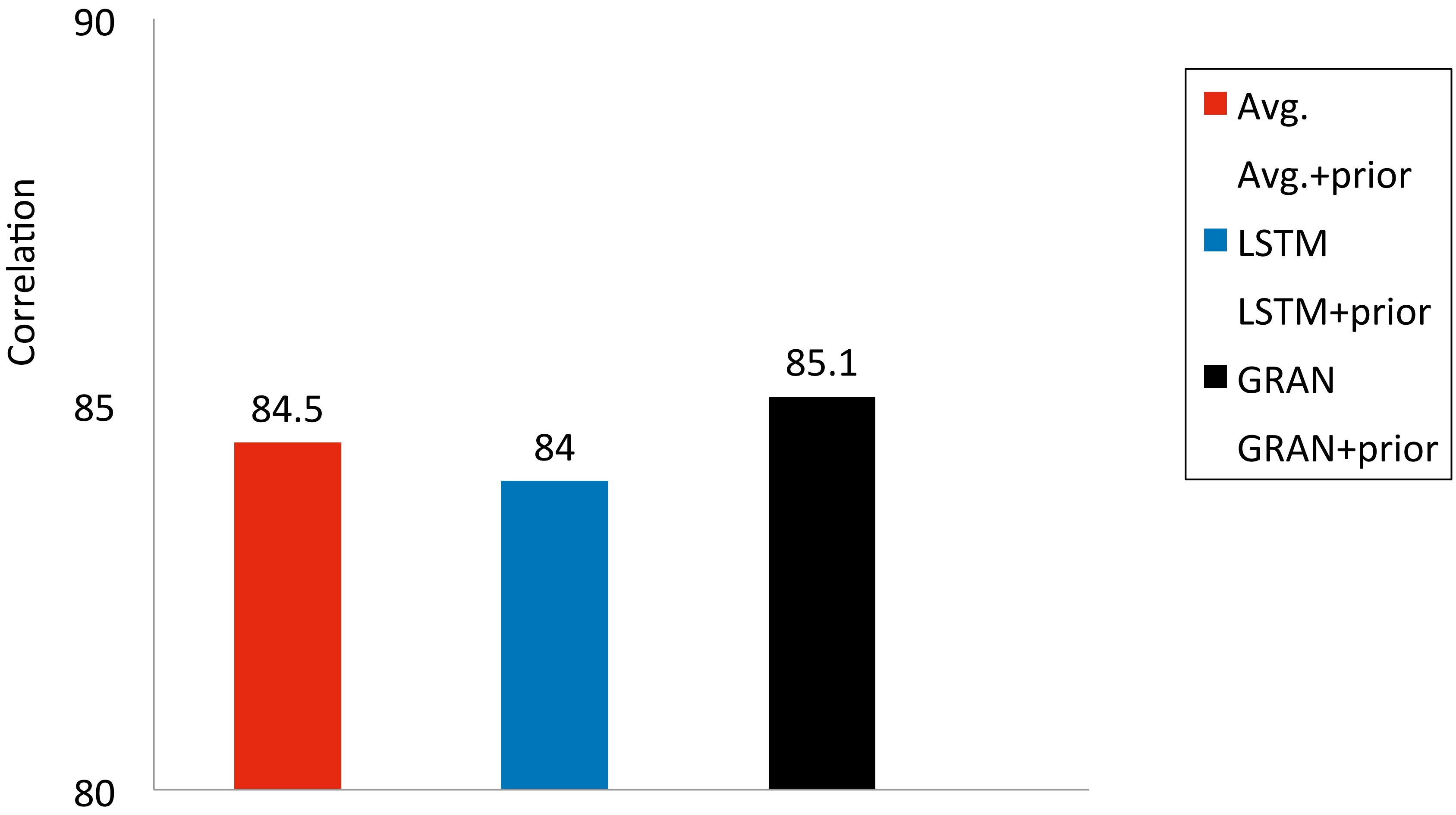
$$g(x) = \frac{1}{|x|} \sum_t a_t$$


GRAN
75
65
55
Correlation
68.4
59.3
65.1
68.4
68.6
68.9
Avg.
LSTM
LSTM+scrambling
LSTM+scrambling+dropout
LSTM Avg.+scramb.+dropout
GRAN+scramb.+dropout

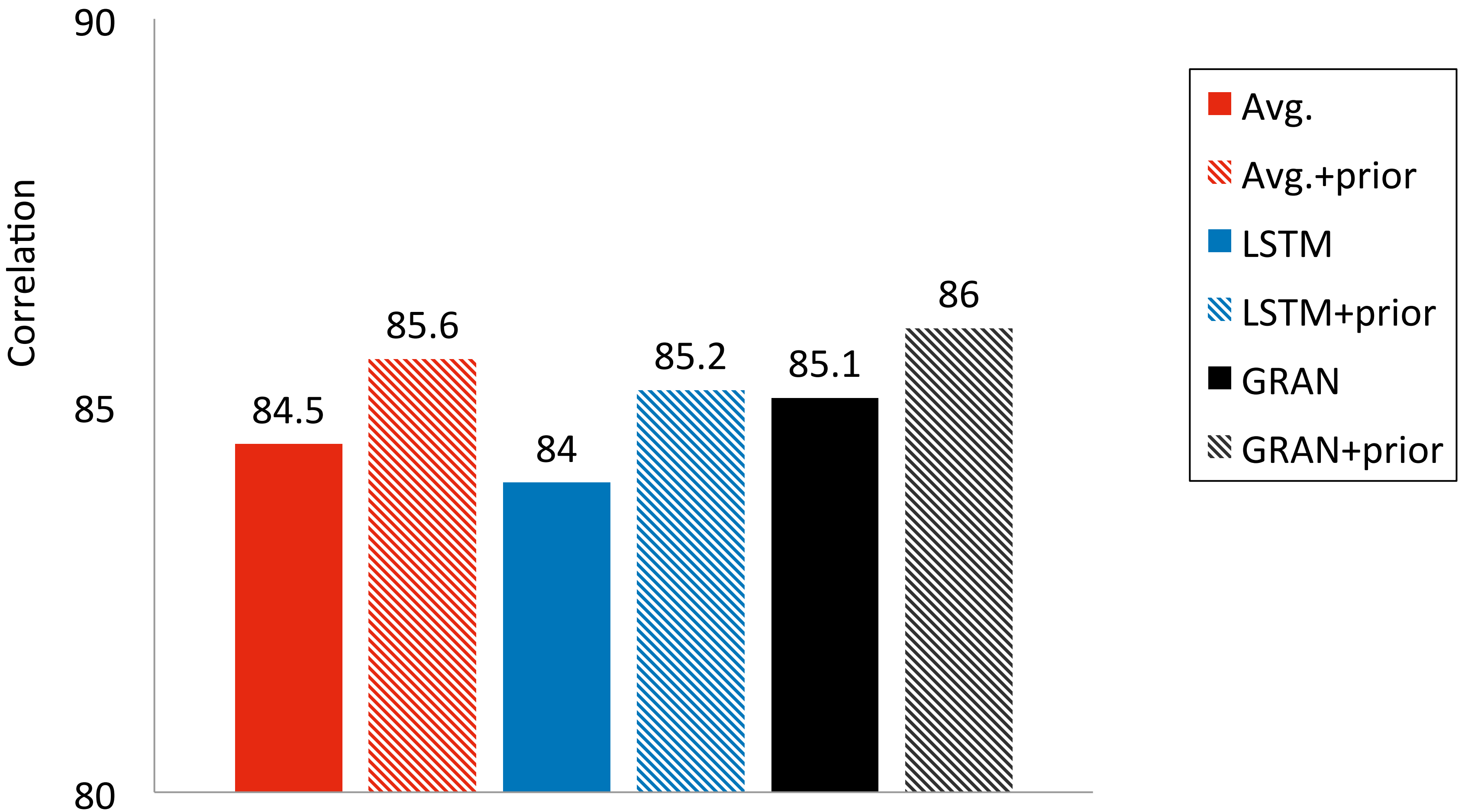
Analyzing GRAN Gates

POS		Dep. Label	
top 10	bot. 10	top 10	bot. 10
NNP	TO	number	possessive
NNPS	WDT	nn	cop
CD	POS	num	det
NNS	DT	acomp	auxpass
VBG	WP	appos	prep
NN	IN	pobj	cc
JJ	CC	vmod	mark
UH	PRP	dobj	aux
VBN	EX	amod	expl
JJS	WRB	conj	neg

Supervised Learning + Regularize to Unsupervised Representations



Supervised Learning + Regularize to Unsupervised Representations



All models benefit from regularizing toward unsupervised representations

Ongoing Work

- New data:

automatically-translated bilingual sentence pairs

the room was very pleasant and the hotel 's location next to the park and teh maritime museum was suberb .

excellent location - right next door to the maritime museum and greenwich park with the observatory and time museum .

Thank you!