Text Analysis for Social Scientists and Leaders

Class 3: Embeddings & Similarity

Prof. Michael Yeomans

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

This	is	a	sentence	also	one	document	not	wolf	star	…
1	1	1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

This	is	a	sentence	also	one	document	not	wolf	star	…
1	1	1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0

Dense Representations

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

This	is	a	sentence	also	one	document	not	wolf	star	…
1	1	1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0

Dense Representations

Price	ft^2	Bedrooms	zone	Parking?
1800	440	3	3	yes
1400	610	1	1	No
1100	575	2	4	No
2200	800	2	2	Yes

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

This	is	a	sentence	also	one	document	not	wolf	star	…
1	1	1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0

Dense Representations

Lower dimensionality -> easier to estimate

Dimension Reduction

Sparse Representation

"This is a sentence"
"This is also one sentence"
"This is a document"
"This is not a document"

This	is	a	sentence	also	one	document	not	wolf	star	…
1	1	1	1	0	0	0	0	0	0	0
1	1	0	1	1	1	0	0	0	0	0
1	1	1	0	0	0	1	0	0	0	0
1	1	1	0	0	0	1	1	0	0	0

Dense Representations

Lower dimensionality -> easier to estimate
Captures “similarity”
- different words have similar meanings

Dimension Reduction

What words are similar to one another?

Dimension Reduction

What words are similar to one another?

Cat vs Lion

Dimension Reduction

What words are similar to one another?

Cat vs Lion

Cat vs Mouse

Dimension Reduction

What words are similar to one another?

Cat	vs	Lion
Cat	vs	Mouse
King	vs	Lion

Dimension Reduction

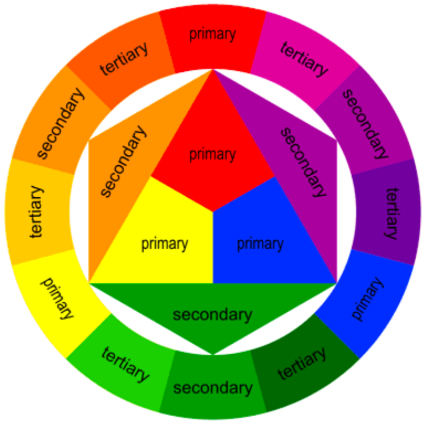
What words are similar to one another?

Cat	vs	Lion
Cat	vs	Mouse
King	vs	Lion
Red	vs	Green

Dimension Reduction

What words are similar to one another?

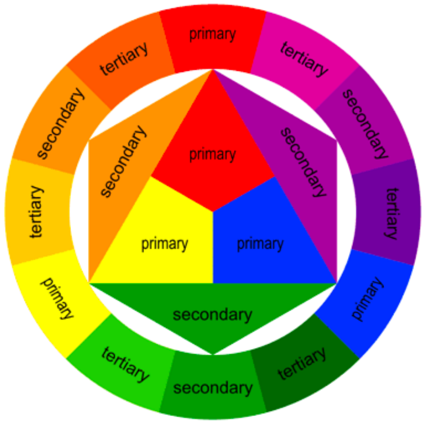
Cat	vs	Lion
Cat	vs	Mouse
King	vs	Lion
Red	vs	Green



Dimension Reduction

What words are similar to one another?

Cat	vs	Lion
Cat	vs	Mouse
King	vs	Lion
Red	vs	Green
Mouse	vs	Keyboard



Dimension Reduction

What words are similar to one another?

Cat vs Lion

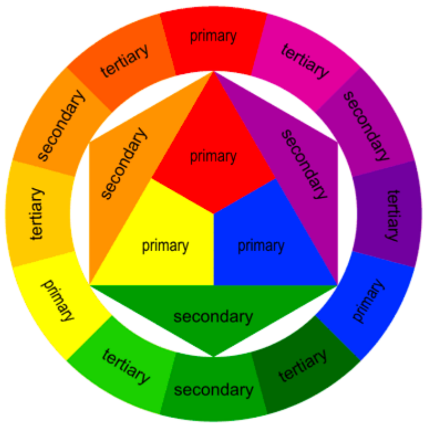
Cat vs Mouse

King vs Lion

Red vs Green

Mouse vs Keyboard

Cat vs Keyboard



Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Synonyms: same meaning in all senses
Probably no true examples….

Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Synonyms: same meaning in all senses
- Probably no true examples….
  - Big/large

Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Synonyms: same meaning in all senses

Probably no true examples….

Big/large - my large sister != my big sister

Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Synonyms: same meaning in all senses

Probably no true examples….

Big/large - my large sister != my big sister

Antonyms: opposite along one dimension, otherwise similar

Hot/cold, in/out, tall/short, long/short

Dimension Reduction

What words are similar to one another?

Similarity is high-dimensional!

Synonyms: same meaning in all senses
Probably no true examples….
Big/large - my large sister != my big sister

Antonyms: opposite along one dimension, otherwise similar
Hot/cold, in/out, tall/short, long/short

Homonyms: different meanings for same character string
River bank / National bank

Dimension Reduction

Theory-Driven - *“We’ll tell you what a word means”*

Dimension Reduction

Theory-Driven - *“We’ll tell you what a word means”*

Dictionaries

Dimension Reduction

Theory-Driven - *“We’ll tell you what a word means”*

Dictionaries

Data-Driven - *“You will know a word by the company it keeps”*

Dimension Reduction

Theory-Driven - *"We'll tell you what a word means"*

Dictionaries

Data-Driven - *"You will know a word by the company it keeps"*

Two senses of similarity:

Dimension Reduction

Theory-Driven - *"We'll tell you what a word means"*

Dictionaries

Data-Driven - *"You will know a word by the company it keeps"*

Two senses of similarity:

Syntagmatic - often found in the same documents

"GDP", "inflation", "economy", "currency"

e.g. topic models (LDA), Latent Semantic Analysis

Dimension Reduction

Theory-Driven - *"We'll tell you what a word means"*

Dictionaries

Data-Driven - *"You will know a word by the company it keeps"*

Two senses of similarity:

Syntagmatic - often found in the same documents

"GDP", "inflation", "economy", "currency"

e.g. topic models (LDA), Latent Semantic Analysis

Paradigmatic - often have the same neighbours

"wrote", "remarked", "said", "added"

e.g. word2vec, GloVe, BERT, GPT-1/2/3

Dictionaries

Dictionaries

Every word has its own score (from raters)

Dictionaries

Every word has its own score (from raters)

	Word	Score	Word	Scor
Valence	love	1.000	toxic	0.008
	happy	1.000	nightmare	0.005
Arousal	elated	0.960	mellow	0.069
	frenzy	0.965	napping	0.046
Dominance	powerful	0.991	weak	0.045
	leadership	0.983	empty	0.081

(Osgood et al., 1957; Mohammad, 2018)

Dictionaries

Every word has its own score (from raters)

	Word	Score	Word	Scor
Valence	love	1.000	toxic	0.008
	happy	1.000	nightmare	0.005
Arousal	elated	0.960	mellow	0.069
	frenzy	0.965	napping	0.046
Dominance	powerful	0.991	weak	0.045
	leadership	0.983	empty	0.081

(Osgood et al., 1957; Mohammad, 2018)

Every word in a category has the same score

Dictionaries

Every word has its own score (from raters)

	Word	Score	Word	Scor
Valence	love	1.000	toxic	0.008
	happy	1.000	nightmare	0.005
Arousal	elated	0.960	mellow	0.069
	frenzy	0.965	napping	0.046
Dominance	powerful	0.991	weak	0.045
	leadership	0.983	empty	0.081

(Osgood et al., 1957; Mohammad, 2018)

Every word in a category has the same score

LIWC

(Tausczik & Pennebaker, 2010)

“a+”	"abound"	"abounds"	"achievement"
"abundance"	"abundant"	"accessable"	“acumen"
"accessible"	"acclaim"	"acclaimed"	"adequate"
"acclamation"	"accolade"	"accolades"	"achievements
"accommodative"	"accomodative'	"accomplish"	"adaptable"
"accomplished"	“accomplishment	ccomplishment	"adjustable"

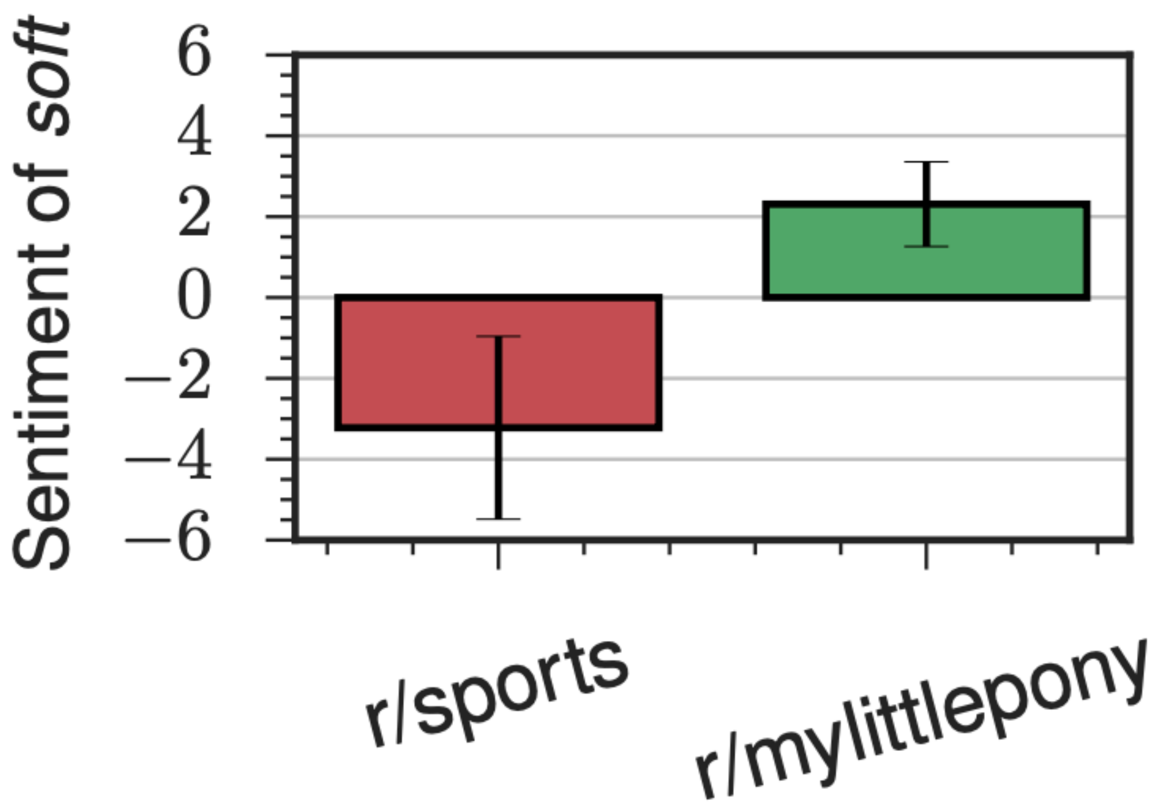
Sentiment

Context Specificity

"Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora"
(Hamilton et al., 2016)

Sentiment

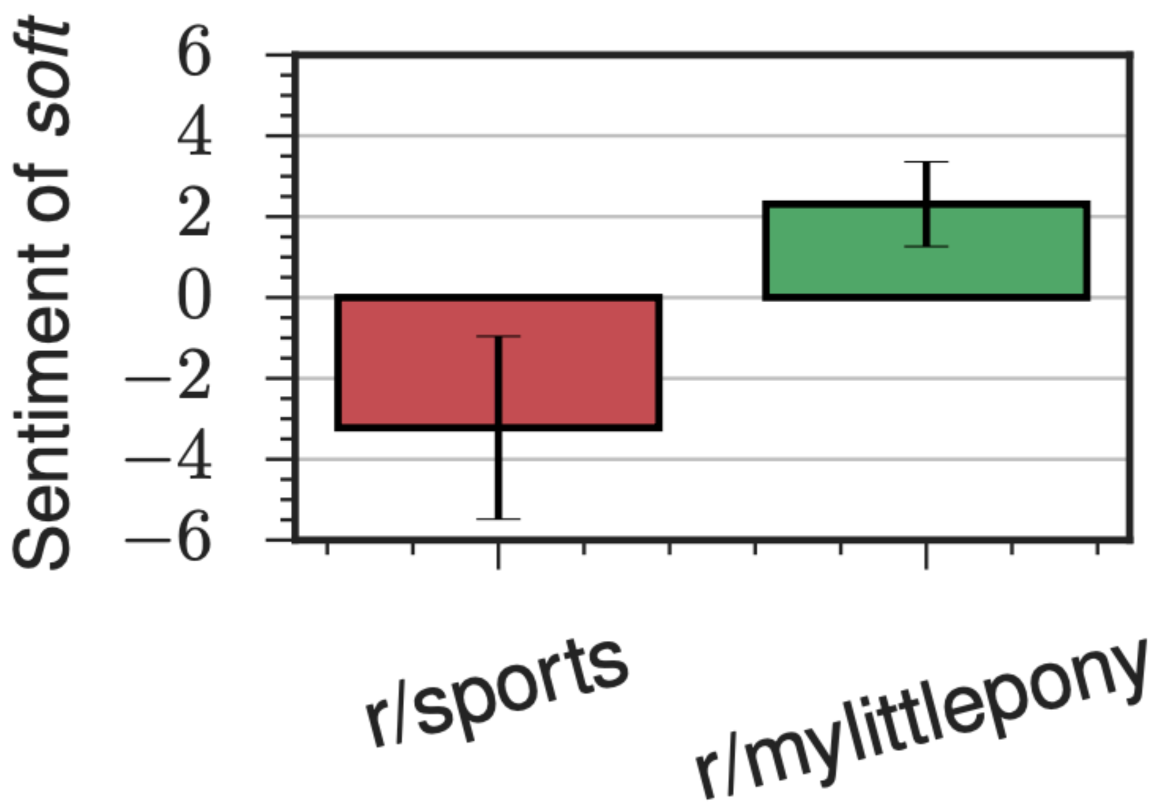
Context Specificity

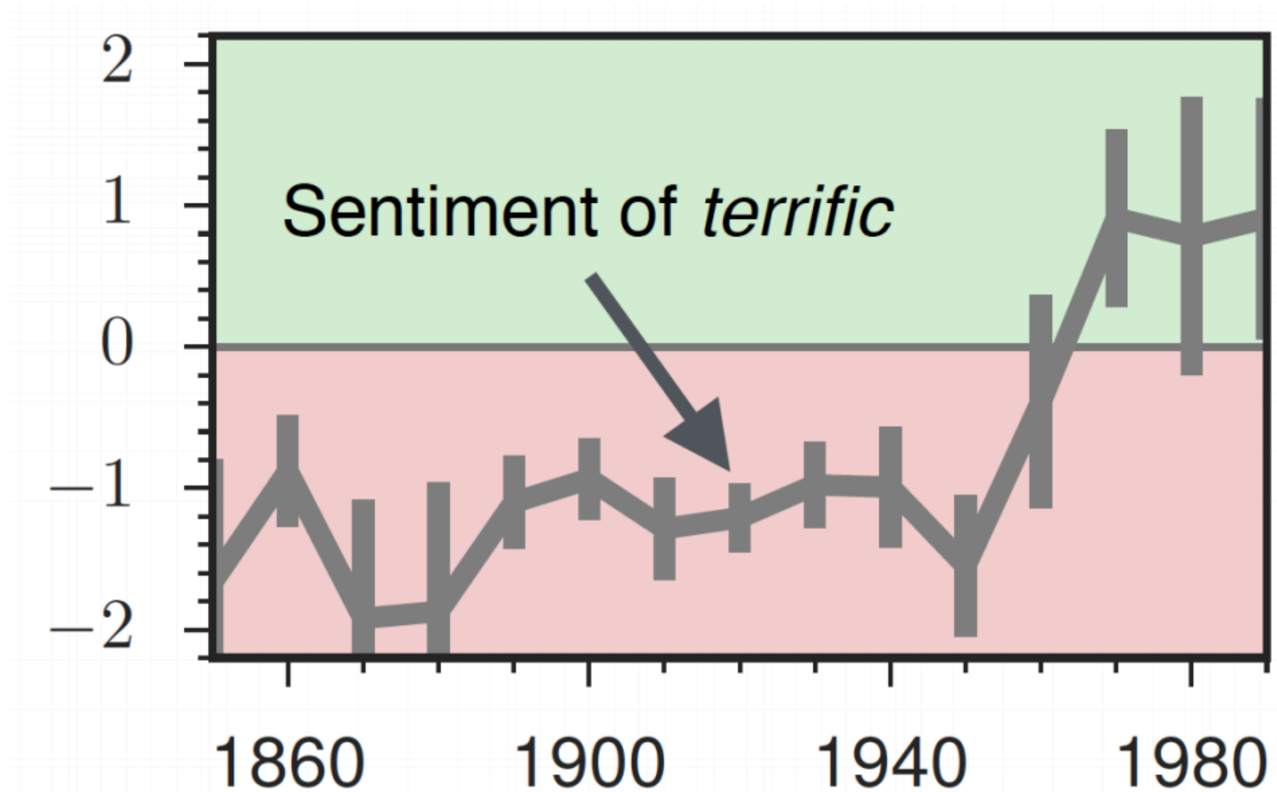


"Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora"
(Hamilton et al., 2016)

Sentiment

Context Specificity





"Inducing Domain-Specific Sentiment Lexicons from Unlabeled Corpora"
(Hamilton et al., 2016)

Sentiment

Context Specificity

Random Forest (supervised ngram model)
Loughran Macdonald (finance-specific)
Harvard PsychoSocial dictionary (domain-general)

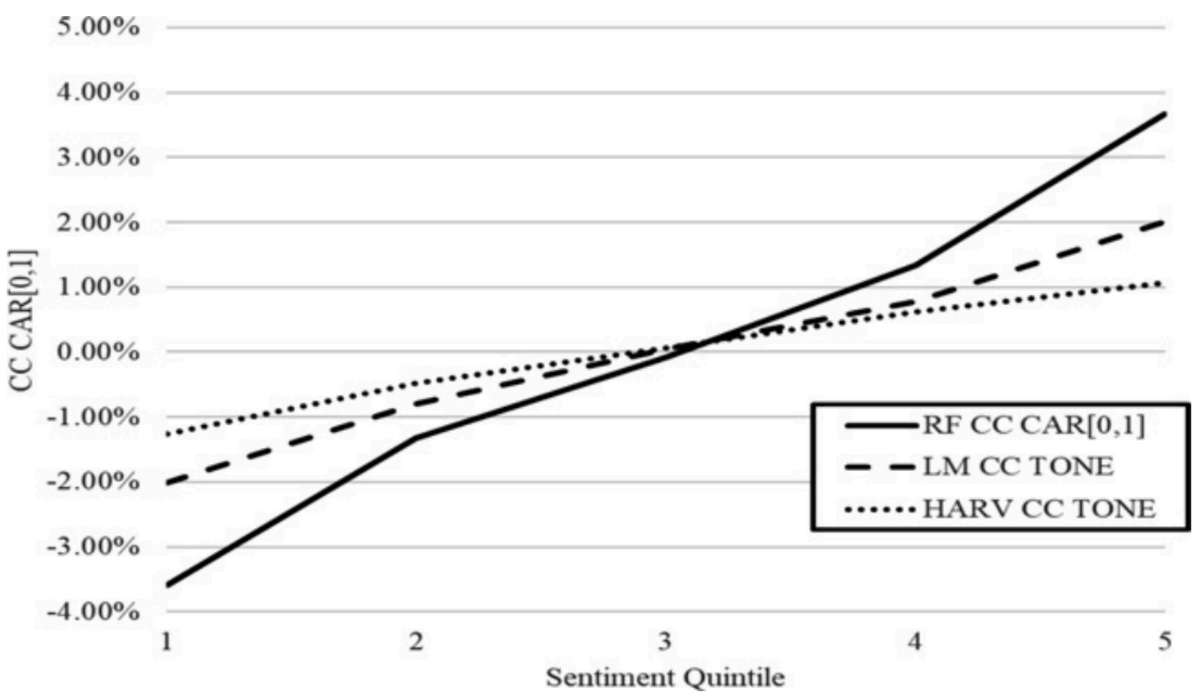
"Disclosure Sentiment: Machine Learning vs. Dictionary Methods"
(Frankel, Jennings & Lee, 2022)

Sentiment

Context Specificity

Random Forest (supervised ngram model)
Loughran Macdonald (finance-specific)
Harvard PsychoSocial dictionary (domain-general)

Figure 2. Conference Call Returns and Sentiment



"Disclosure Sentiment: Machine Learning vs. Dictionary Methods"
(Frankel, Jennings & Lee, 2022)

Sentiment

Context Specificity

Random Forest (supervised ngram model)
Loughran Macdonald (finance-specific)
Harvard PsychoSocial dictionary (domain-general)

Figure 2. Conference Call Returns and Sentiment

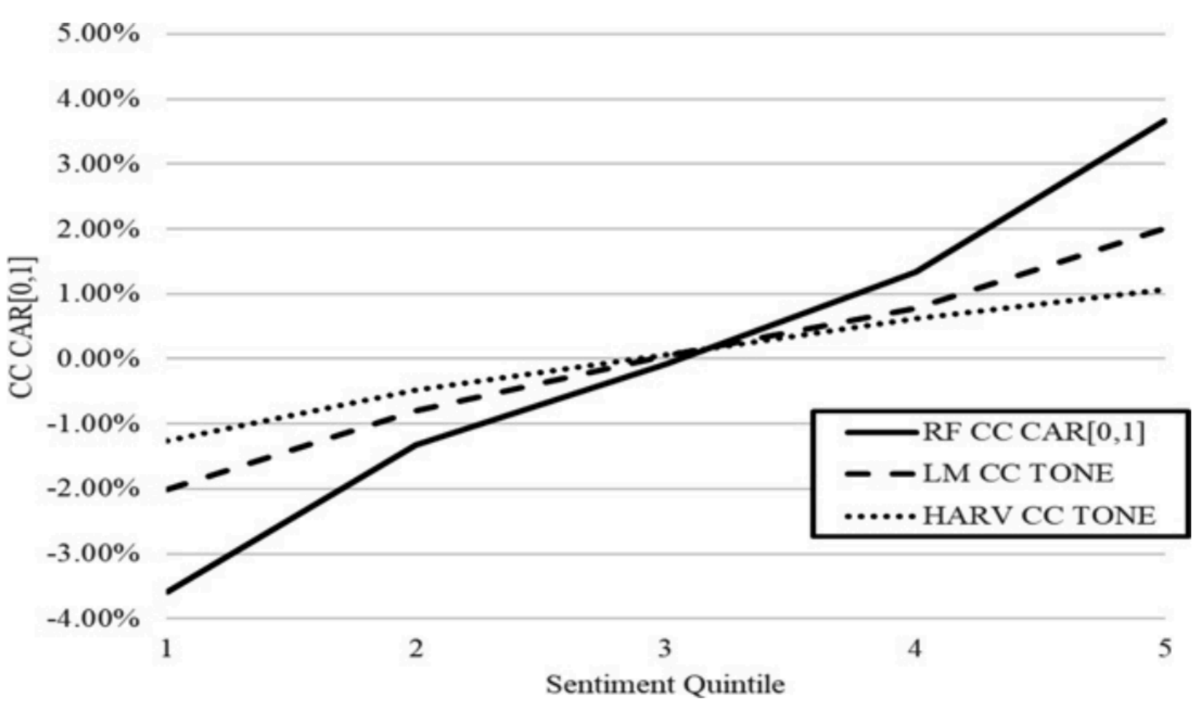
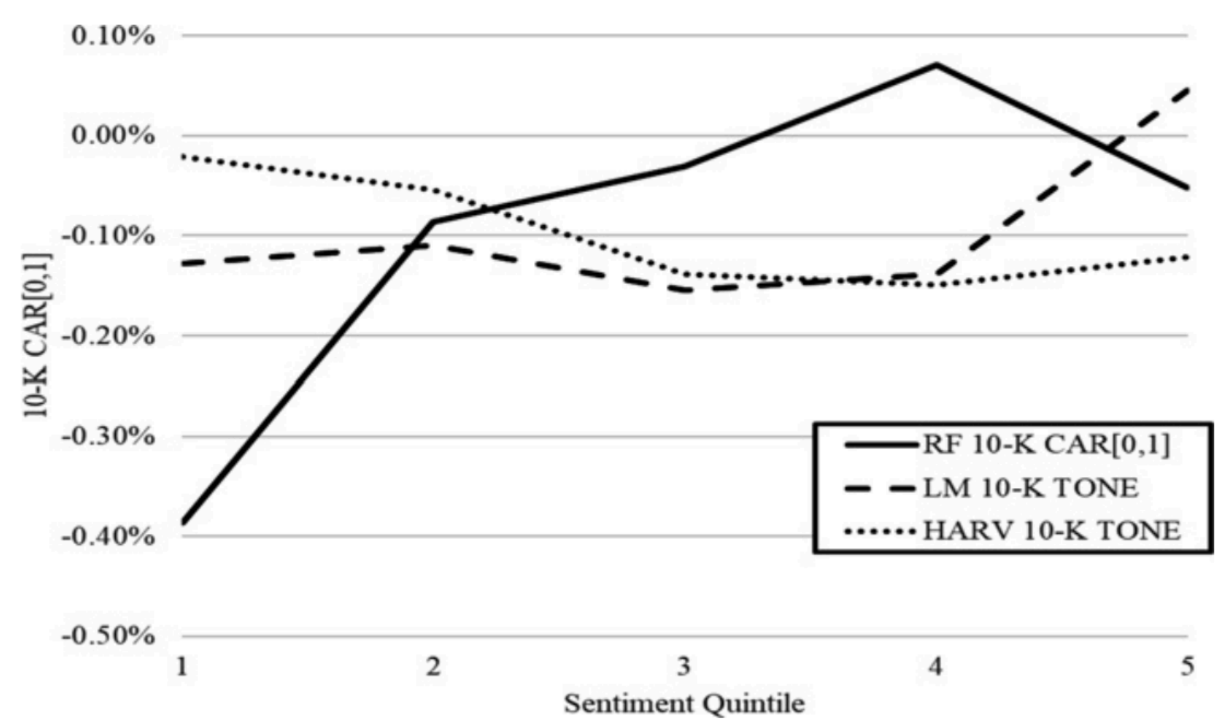


Figure 1. 10-K Filing Returns and Sentiment



"Disclosure Sentiment: Machine Learning vs. Dictionary Methods"
(Frankel, Jennings & Lee, 2022)

Sentiment

Context Specificity

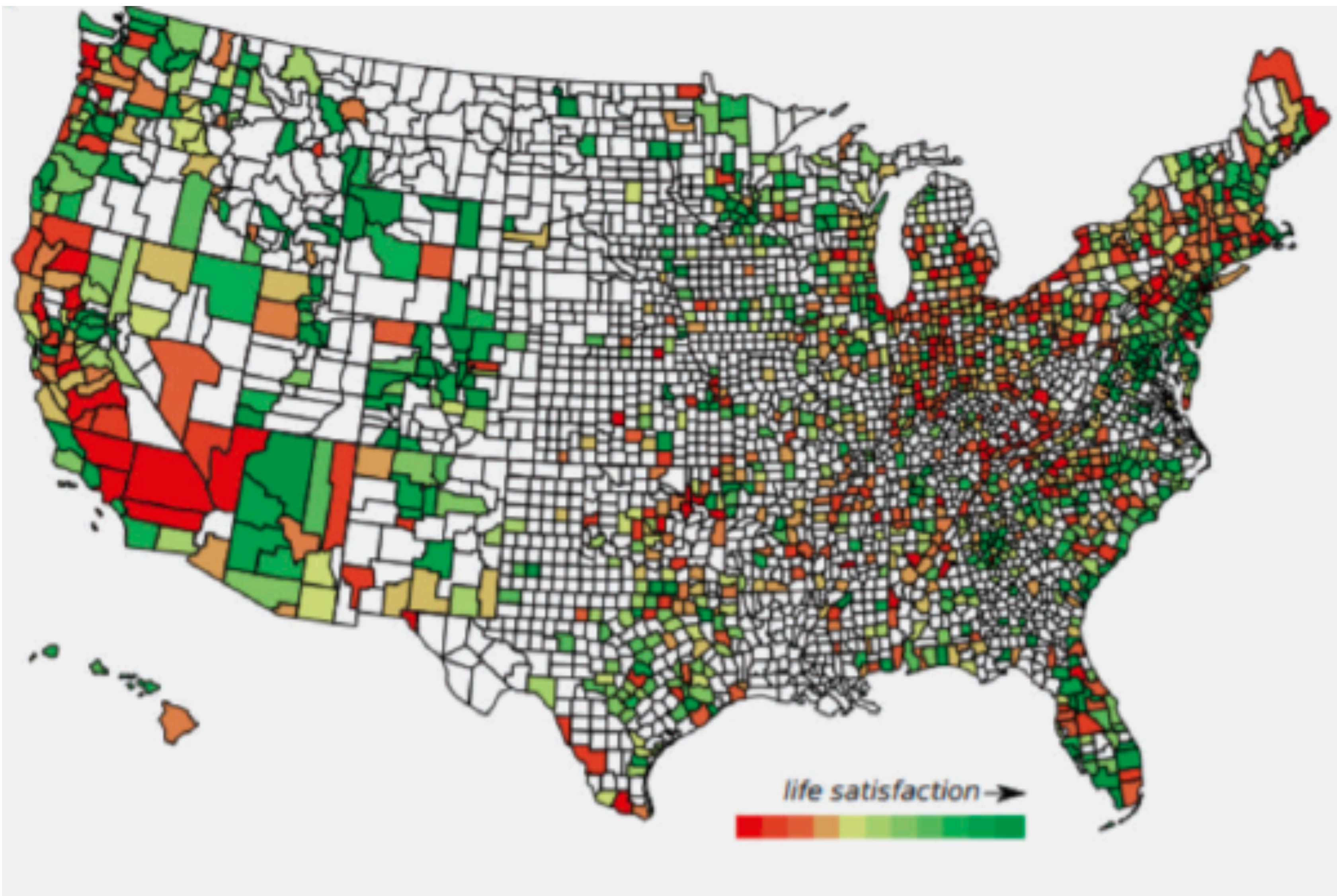
Random Forest (supervised ngram model)
Loughran Macdonald (finance-specific)
Harvard PsychoSocial dictionary (domain-general)

Positive	**Negative**
Conclusive	Evaded
Resolve	Condone
Versatile	Confess
Vibrant	Deadlocks
Invent	Costly

"Disclosure Sentiment: Machine Learning vs. Dictionary Methods"
(Frankel, Jennings & Lee, 2022)

Sentiment

Topic vs Style

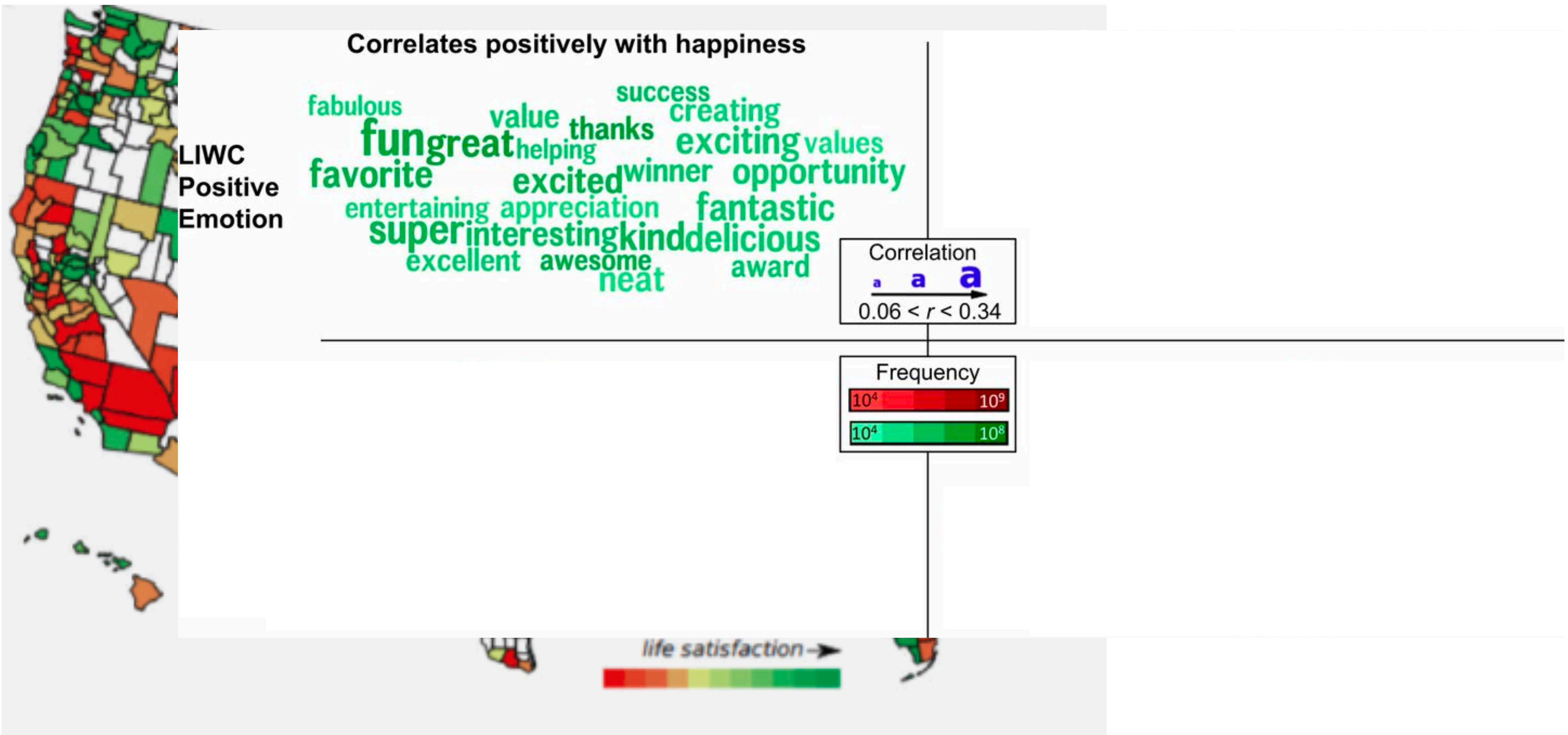


"Estimating geographic subjective well-being from Twitter"

(Jaidka et al., 2020)

Sentiment

Topic vs Style

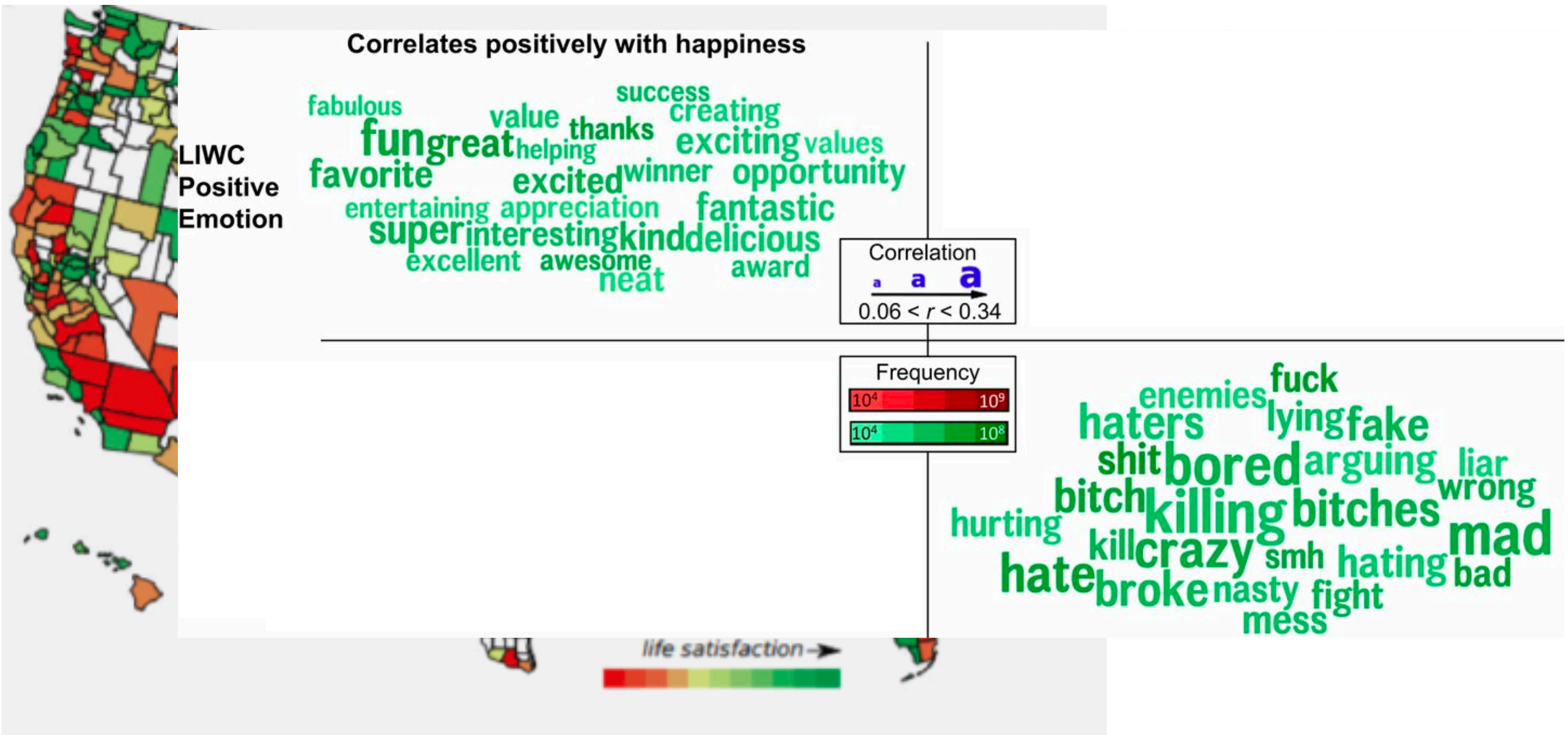


"Estimating geographic subjective well-being from Twitter"

(Jaidka et al., 2020)

Sentiment

Topic vs Style

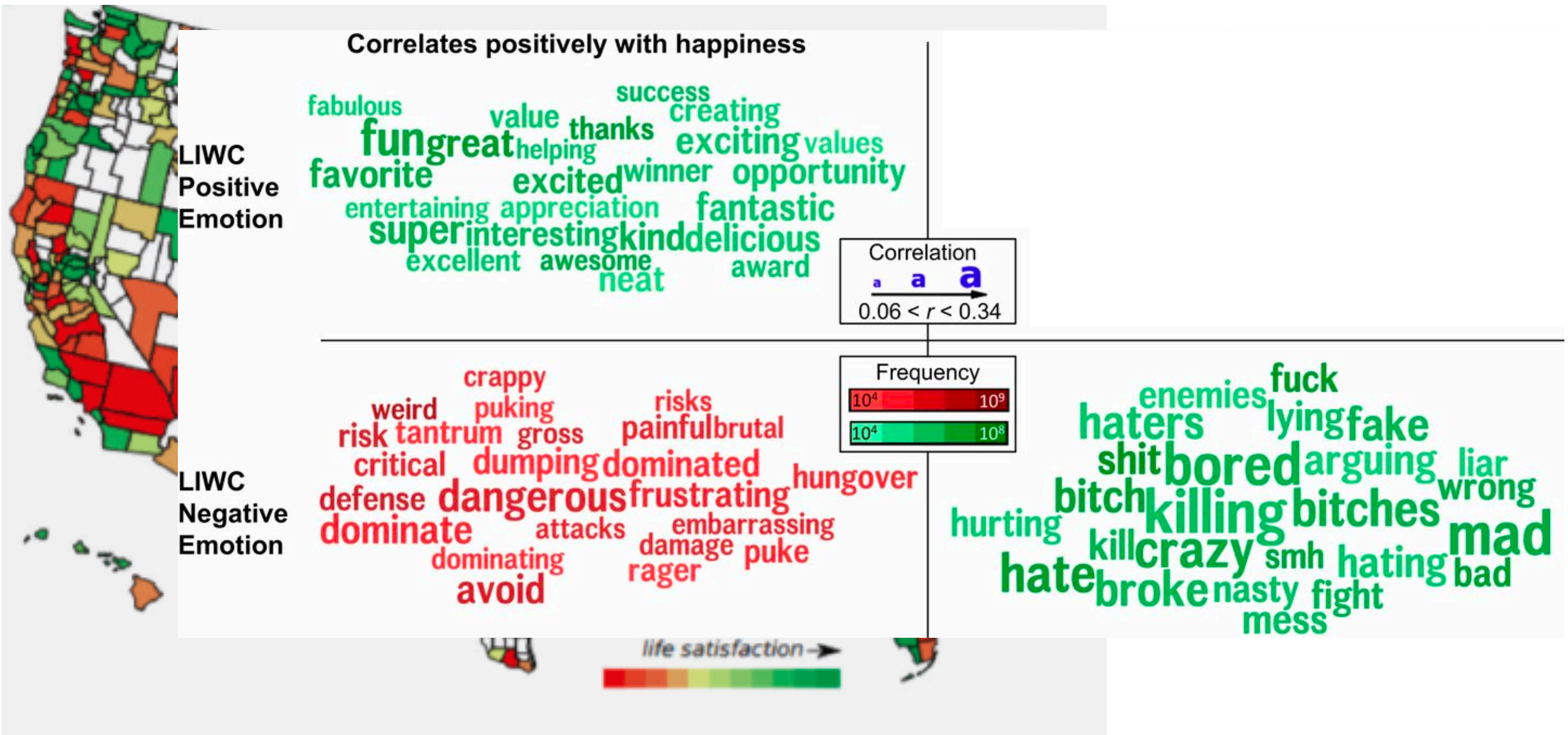


“Estimating geographic subjective well-being from Twitter”

(Jaidka et al., 2020)

Sentiment

Topic vs Style

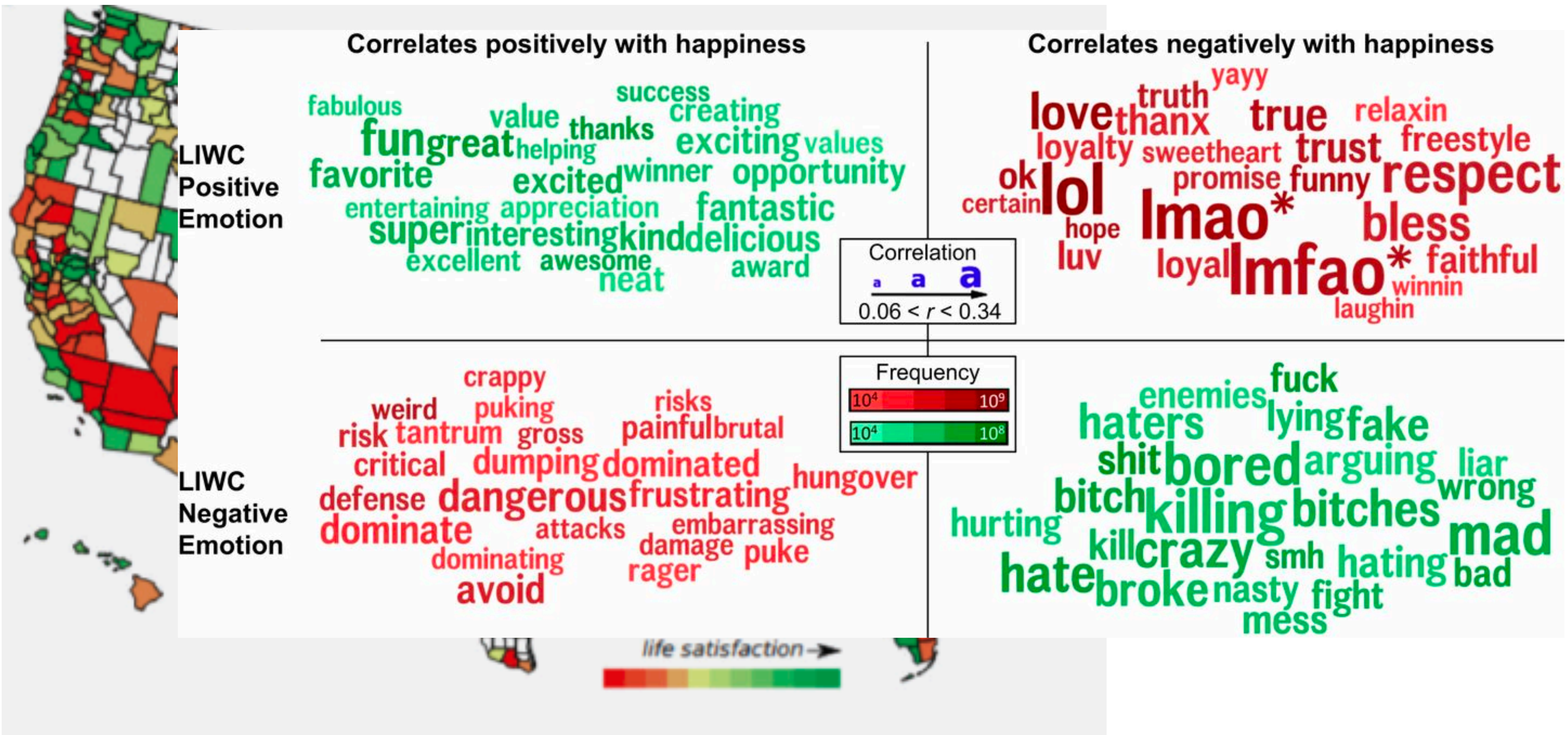


“Estimating geographic subjective well-being from Twitter”

(Jaidka et al., 2020)

Sentiment

Topic vs Style

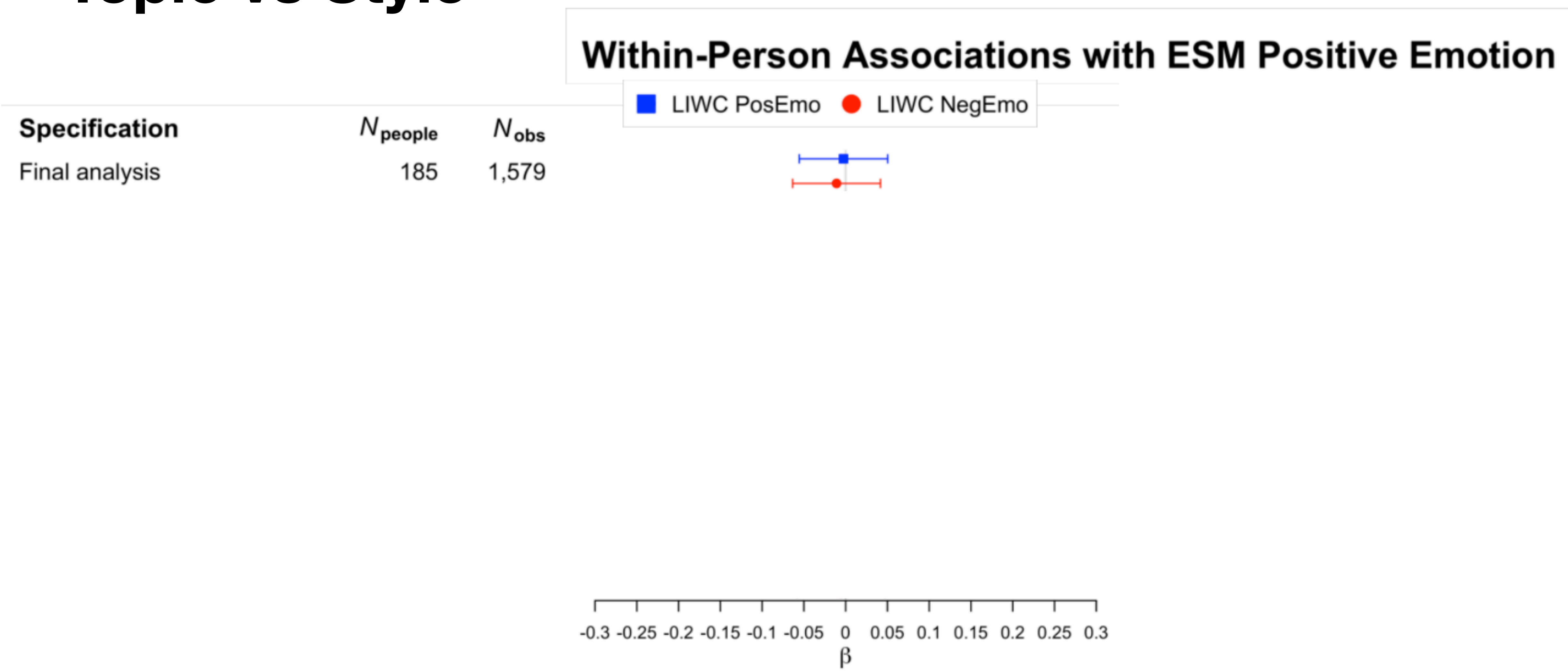


"Estimating geographic subjective well-being from Twitter"

(Jaidka et al., 2020)

Sentiment

Topic vs Style

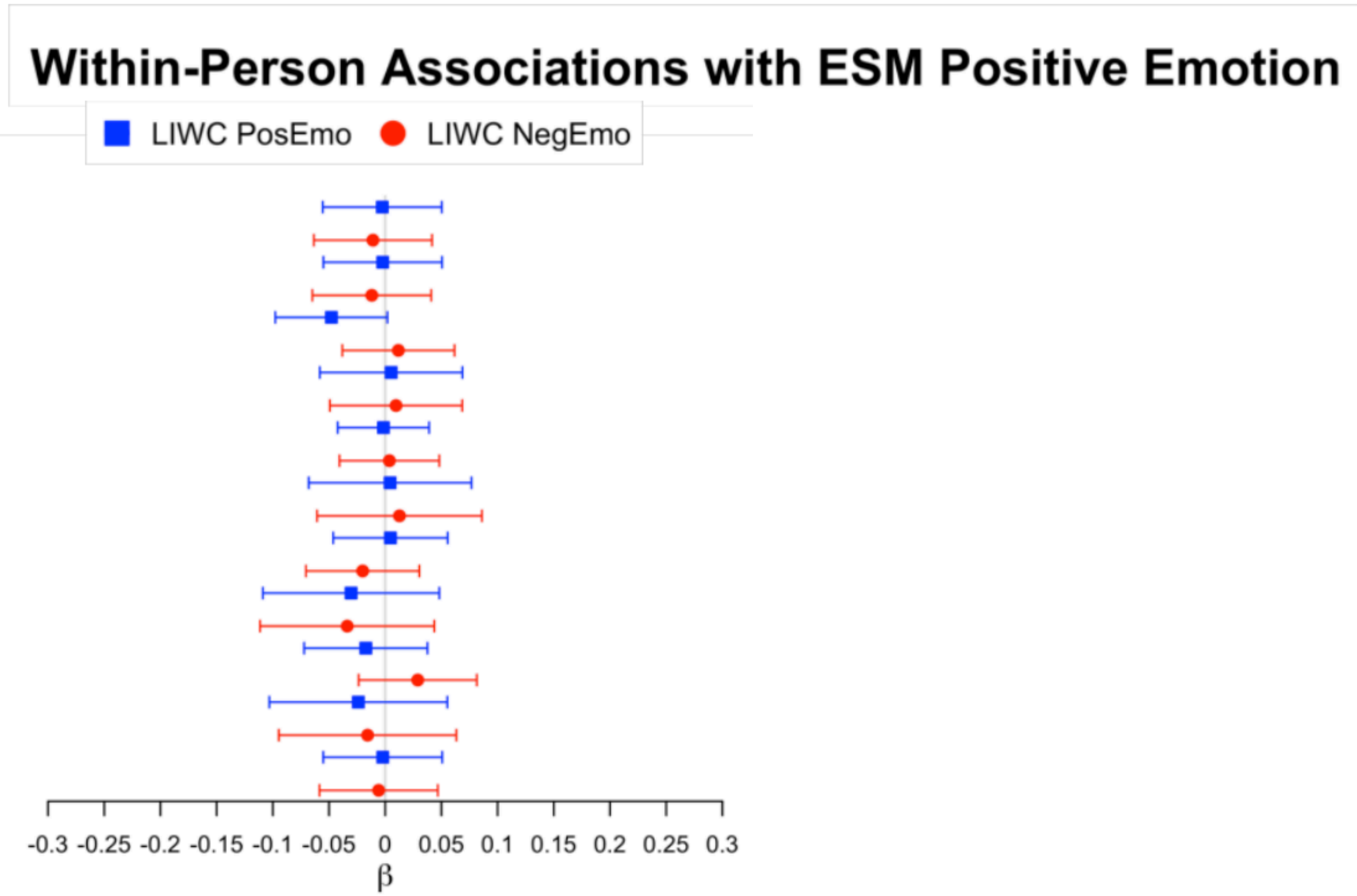


"Emotion Fluctuations and Everyday Speech"
(Sun et al., 2019)

Sentiment

Topic vs Style

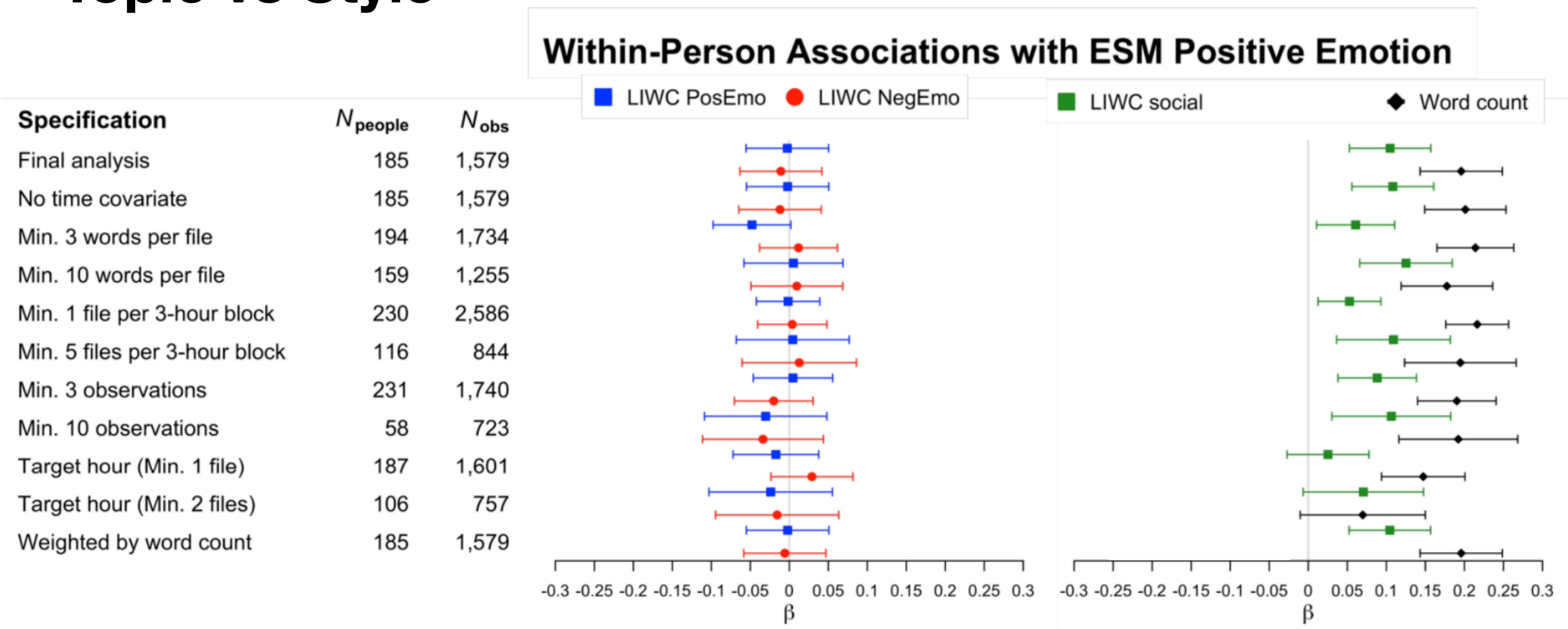
Specification	N_{people}	N_{obs}
Final analysis	185	1,579
No time covariate	185	1,579
Min. 3 words per file	194	1,734
Min. 10 words per file	159	1,255
Min. 1 file per 3-hour block	230	2,586
Min. 5 files per 3-hour block	116	844
Min. 3 observations	231	1,740
Min. 10 observations	58	723
Target hour (Min. 1 file)	187	1,601
Target hour (Min. 2 files)	106	757
Weighted by word count	185	1,579



"Emotion Fluctuations and Everyday Speech"
(Sun et al., 2019)

Sentiment

Topic vs Style



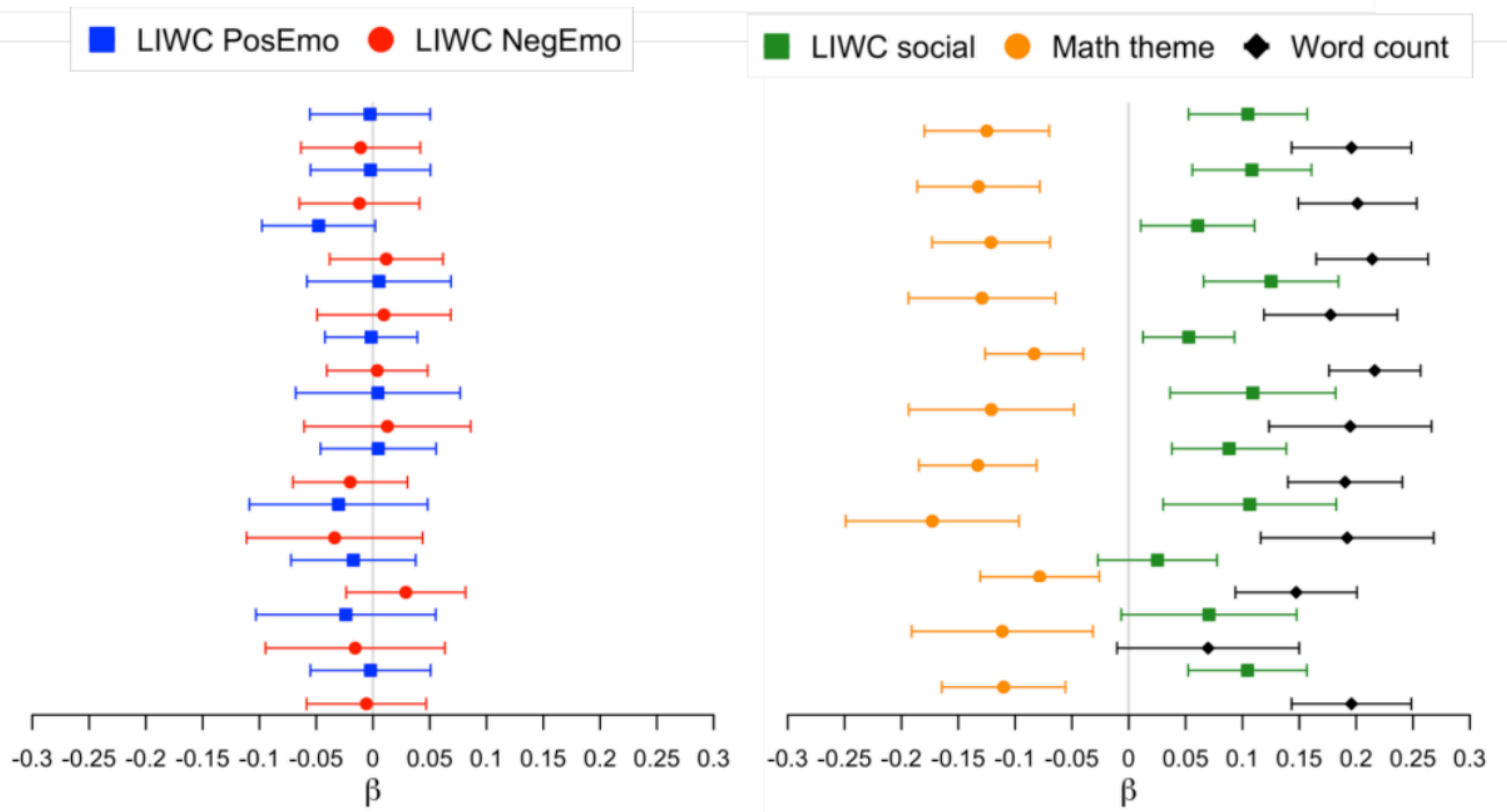
"Emotion Fluctuations and Everyday Speech"
(Sun et al., 2019)

Sentiment

Topic vs Style

Within-Person Associations with ESM Positive Emotion

Specification	N_{people}	N_{obs}
Final analysis	185	1,579
No time covariate	185	1,579
Min. 3 words per file	194	1,734
Min. 10 words per file	159	1,255
Min. 1 file per 3-hour block	230	2,586
Min. 5 files per 3-hour block	116	844
Min. 3 observations	231	1,740
Min. 10 observations	58	723
Target hour (Min. 1 file)	187	1,601
Target hour (Min. 2 files)	106	757
Weighted by word count	185	1,579



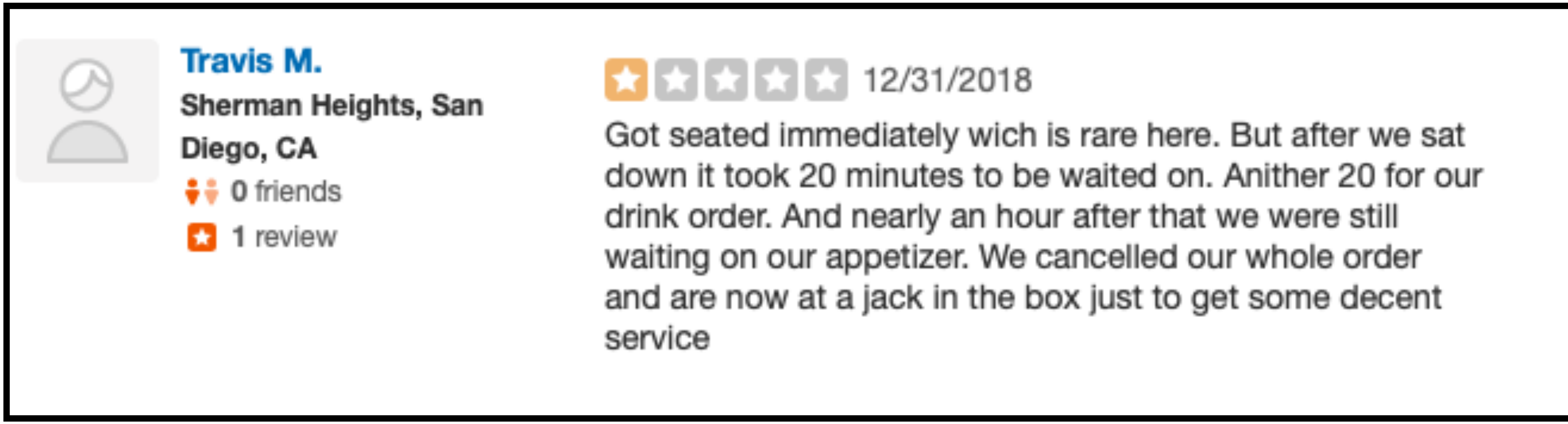
"Emotion Fluctuations and Everyday Speech"
(Sun et al., 2019)

Sentiment

Best case scenario: descriptions of things

Sentiment

Best case scenario: descriptions of things


Travis M.
Sherman Heights, San Diego, CA
0 friends
1 review

12/31/2018

Got seated immediately wich is rare here. But after we sat down it took 20 minutes to be waited on. Anither 20 for our drink order. And nearly an hour after that we were still waiting on our appetizer. We cancelled our whole order and are now at a jack in the box just to get some decent service

Sentiment

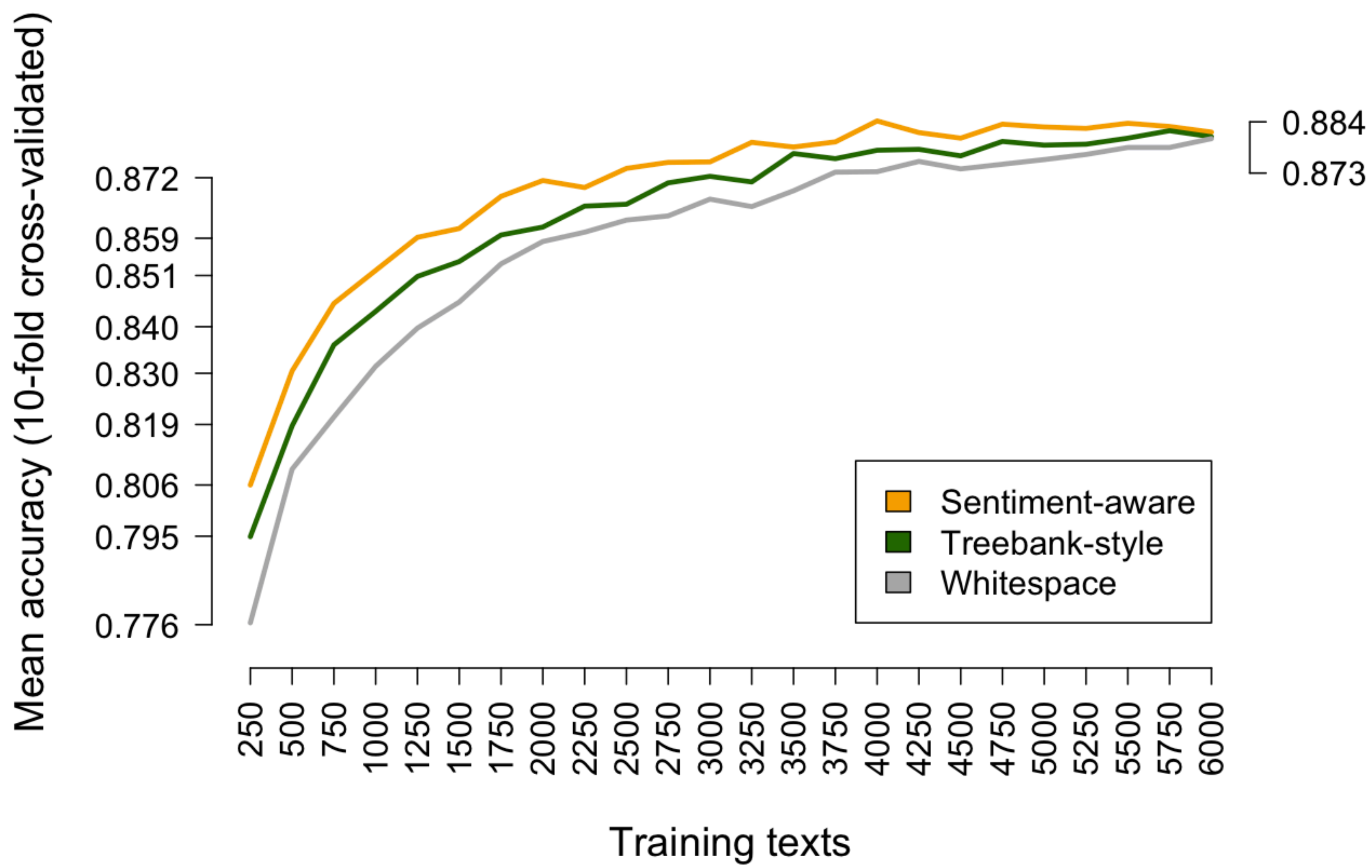
Other improvements

“Sentiment Analysis Symposium" Potts, 2011

Sentiment

Other improvements

Separate tokens for all caps [WOW], punctuation [!], emoticons [:)]



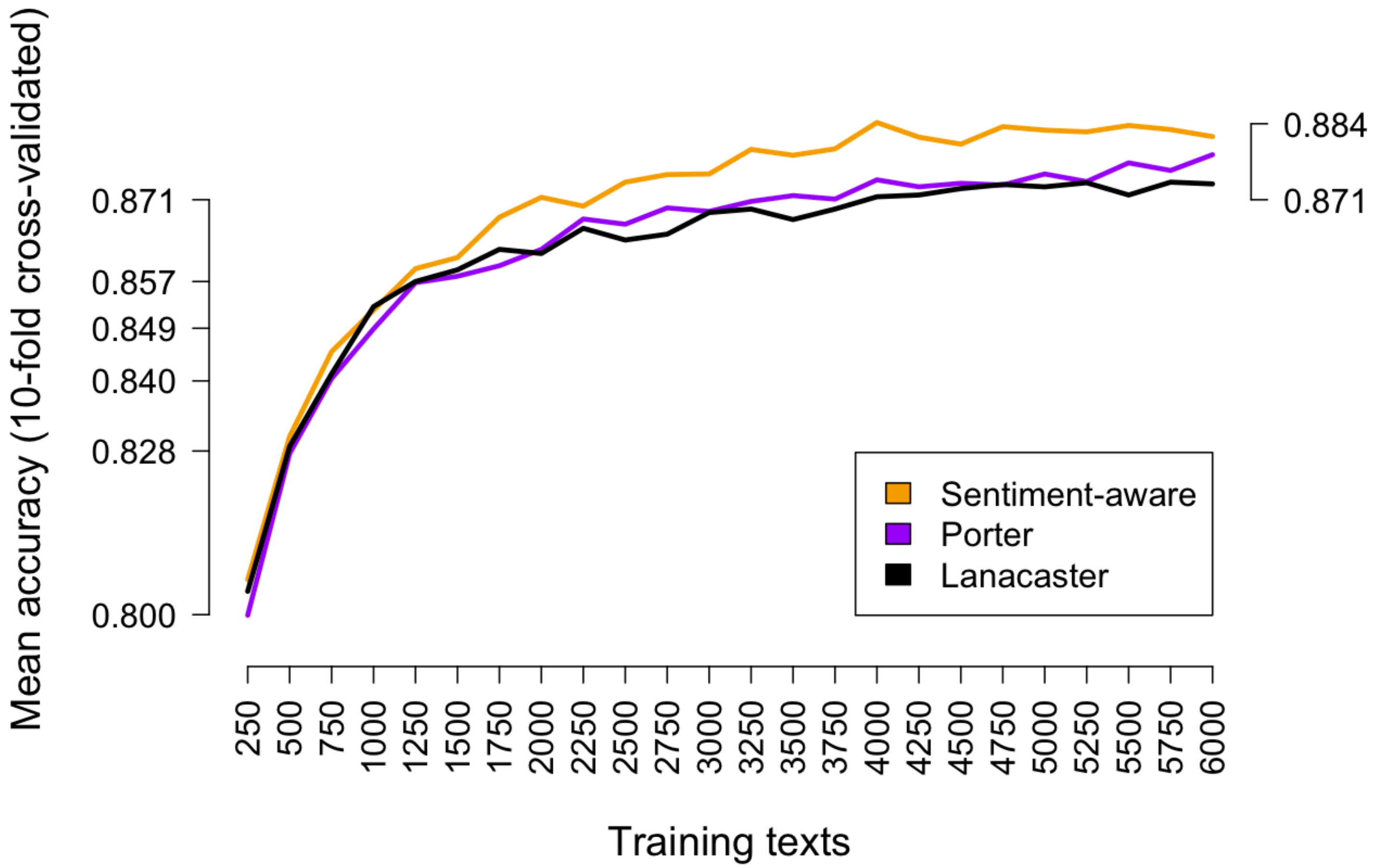
“Sentiment Analysis Symposium" Potts, 2011

Sentiment

Other improvements

Separate tokens for all caps [WOW], punctuation [!], emoticons [:)]

Stemming only tends to help at low sample sizes



“Sentiment Analysis Symposium" Potts, 2011

Sentiment

Other improvements

Negation scoping

“It was that bad.” becomes: [“it”, “was”, “that”,“bad”, “.”]

“Sentiment Analysis Symposium" Potts, 2011

Sentiment

Other improvements

Negation scoping

“It was that bad.” becomes: [“it”, “was”, “that”,“bad”, “.”]

“It was not that bad, honestly.”

“Sentiment Analysis Symposium" Potts, 2011

Sentiment

Other improvements

Negation scoping

“It was that bad.” becomes: [“it”, “was”, “that”,“bad”, “.”]

“It was not that bad, honestly.” becomes:

[“it”, “was”, “not”, “that_NEG”,“bad_NEG”, “,”, “honestly”, “.”]

“Sentiment Analysis Symposium" Potts, 2011

Sentiment

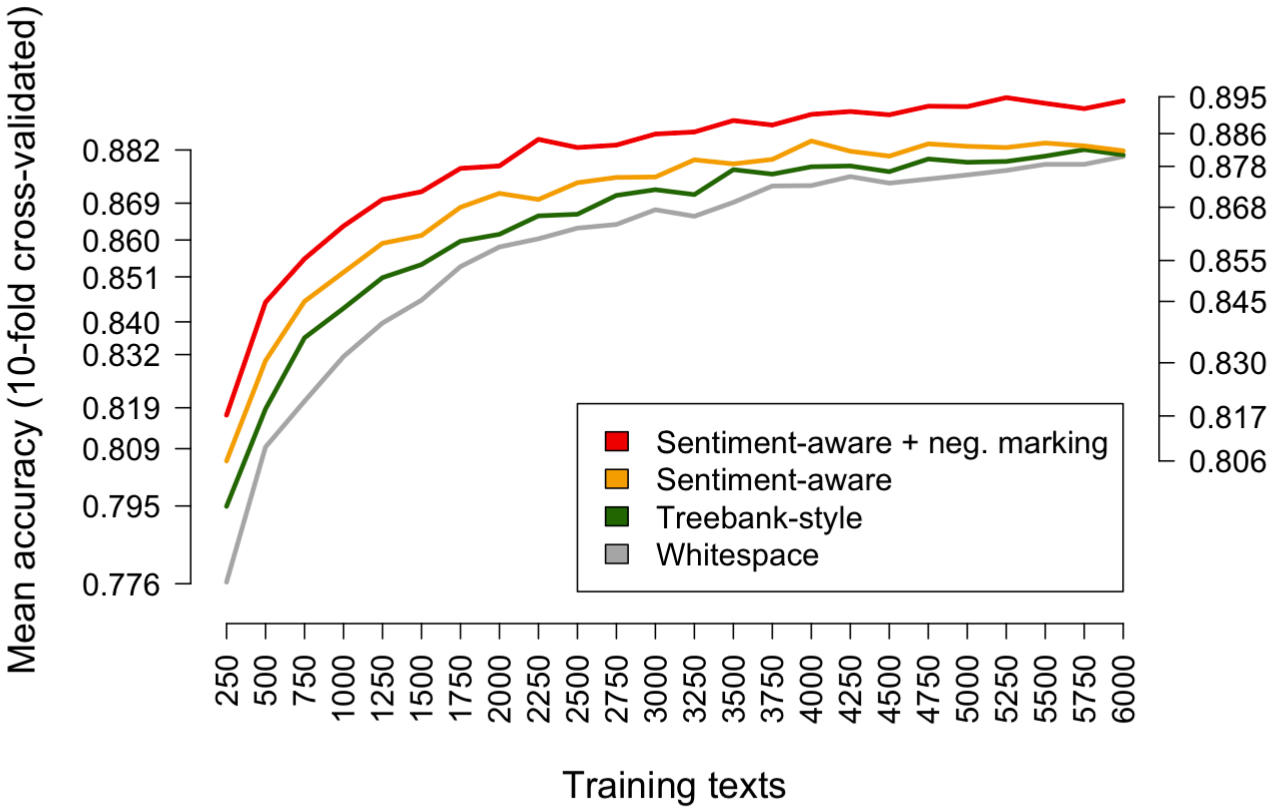
Other improvements

Negation scoping

“It was that bad.” becomes: [“it”, “was”, “that”,“bad”, “.”]

“It was not that bad, honestly.” becomes:

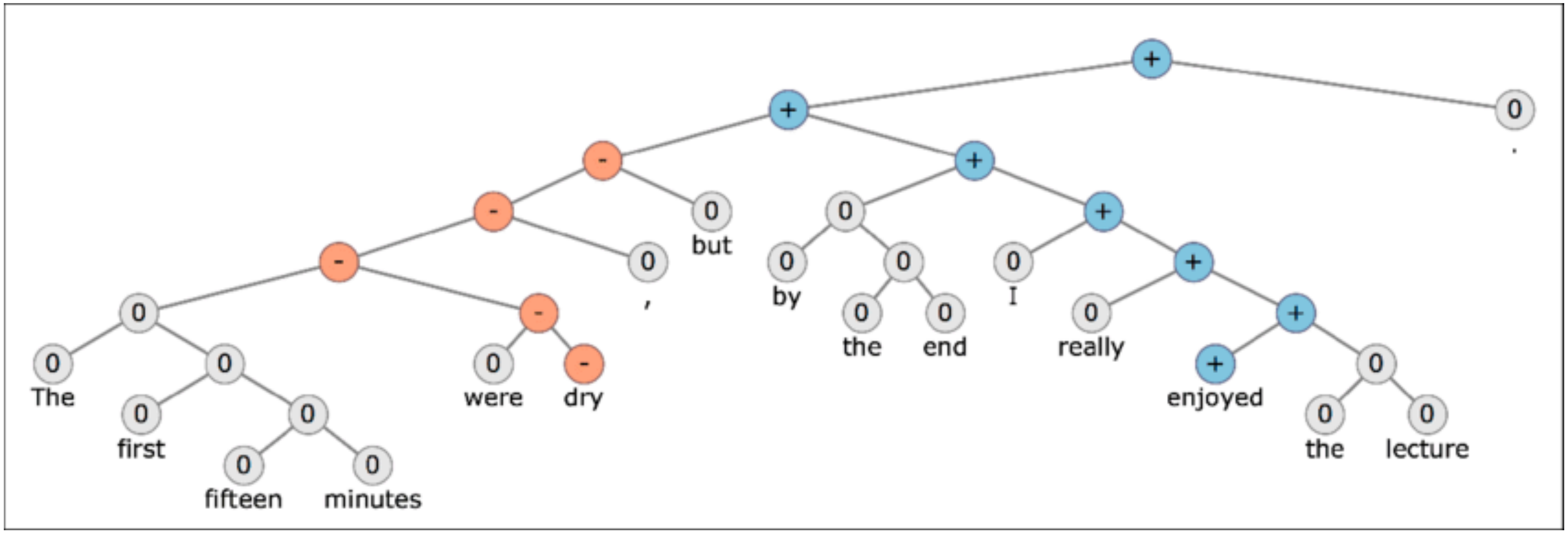
[“it”, “was”, “not”, “that_NEG”,“bad_NEG”, “,”, “honestly”, “.”]



“Sentiment Analysis Symposium" Potts, 2011

Sentiment

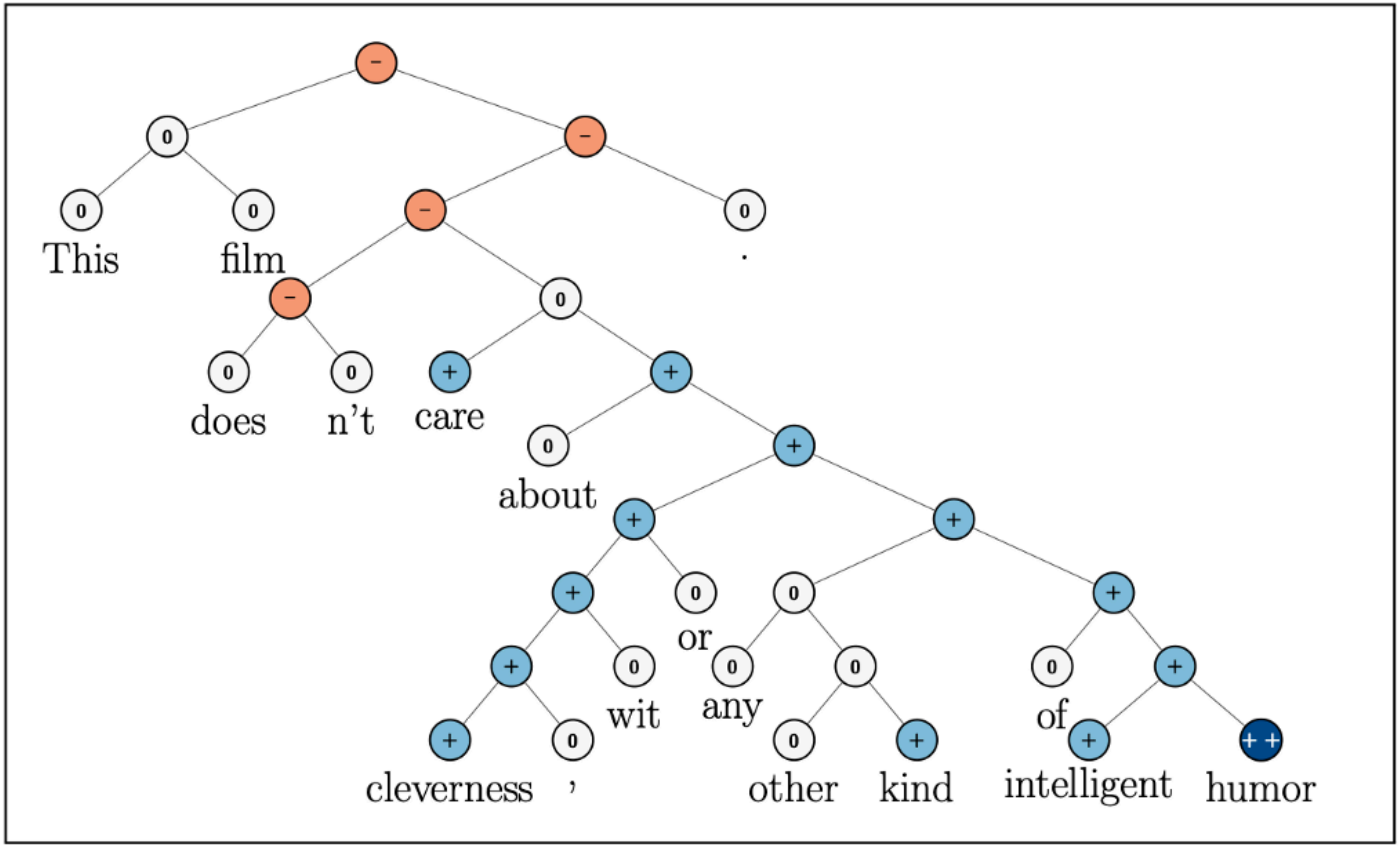
State of the art - Stanford Sentiment treebank



Socher et al., 2013

Sentiment

State of the art - Stanford Sentiment treebank



Socher et al., 2013

Sentiment

Good enough for most: VADER, sentimentr...

Word
This
is
not
good
news
,
it
is
very
bad
.

Sentiment

Good enough for most: VADER, sentimentr...

Word	Sentiment
This	0
is	0
not	0
good	+1
news	0
,	0
it	0
is	0
very	0
bad	-1
.	0

Sentiment

Good enough for most: VADER, sentimentr…

Word	Sentiment	Modifier
This	0	
is	0	
not	0	
good	+1	
news	0	
,	0	
it	0	
is	0	
very	0	
bad	-1	
.	0	

Sentiment

Good enough for most: VADER, sentimentr…

Word	Sentiment	Modifier
This	0	
is	0	
not	0	Negation
good	+1	
news	0	
,	0	
it	0	
is	0	
very	0	Intensifier
bad	-1	
.	0	

Sentiment

Good enough for most: VADER, sentimentr...

Word	Sentiment	Modifier
This	0	
is	0	
not	0	Negation
good	+1	*(-1)
news	0	*(-1)
,	0	
it	0	
is	0	
very	0	Intensifier
bad	-1	*(2)
.	0	

Sentiment

Good enough for most: VADER, sentimentr…

Word	Sentiment	Modifier	Score
This	0		0
is	0		0
not	0	Negation	0
good	+1	*(-1)	-1
news	0	*(-1)	0
,	0		0
it	0		0
is	0		0
very	0	Intensifier	0
bad	-1	*(2)	-2
.	0		0

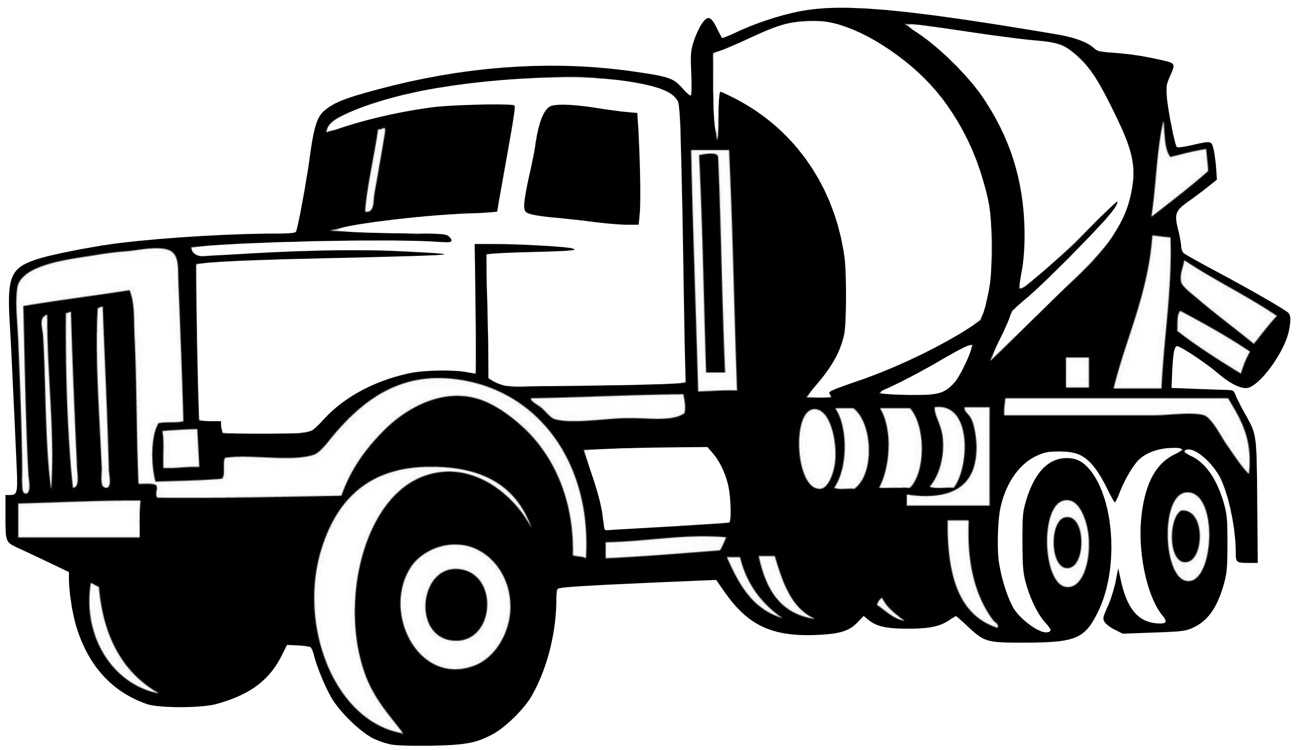
Sentiment

Good enough for most: VADER, sentimentr…

Word	Sentiment	Modifier	Score
This	0		0
is	0		0
not	0	Negation	0
good	+1	*(-1)	-1
news	0	*(-1)	0
,	0		0
it	0		0
is	0		0
very	0	Intensifier	0
bad	-1	*(2)	-2
.	0		0

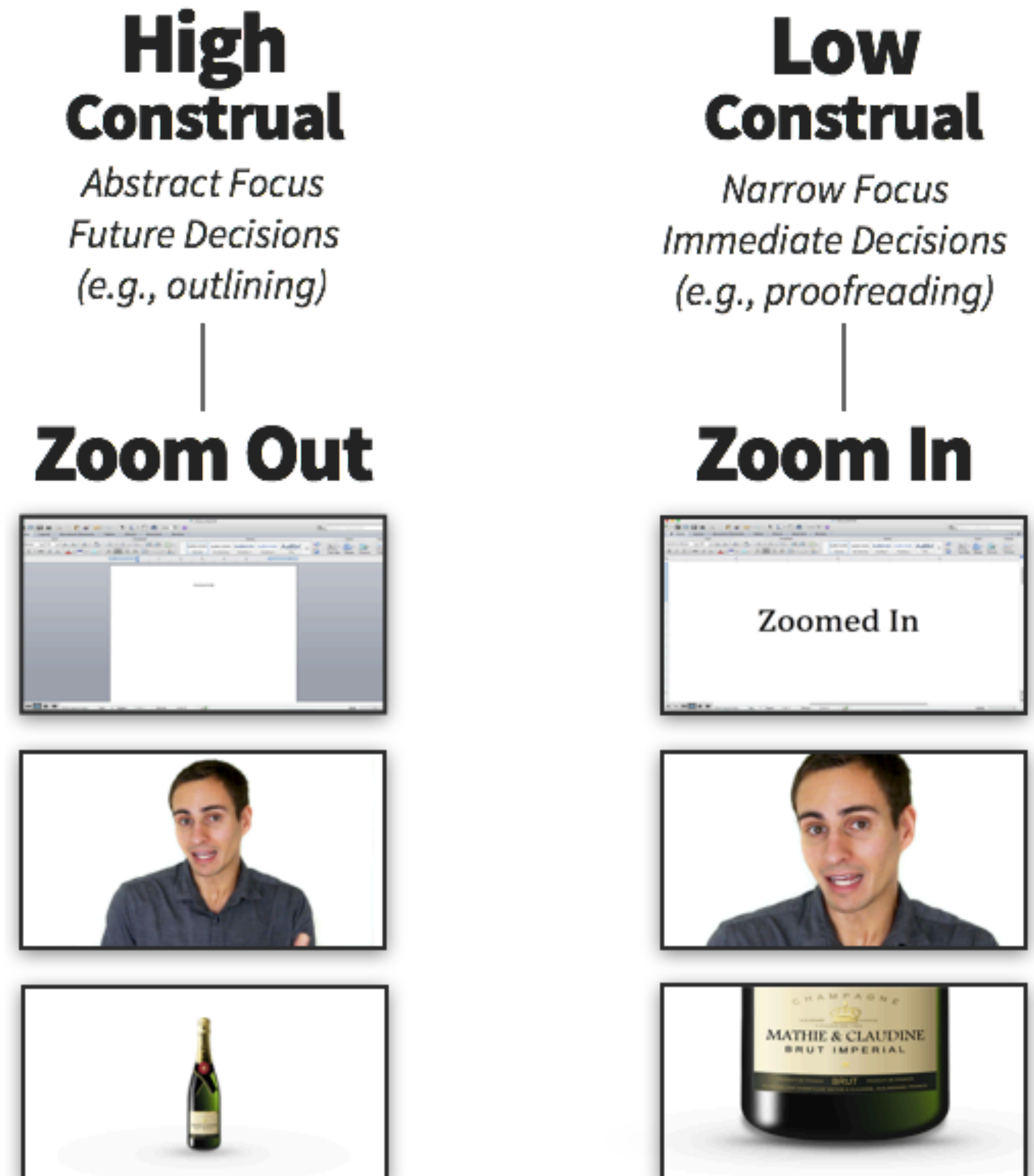
$\boxed{-4}$

Concreteness



(Yeomans, OBDHP, 2021)

Concreteness



(Yeomans, OBDHP, 2021)

The Data

Giving Advice

Domain: People writing advice for others

6 contexts, 3,289 observations

Concreteness: human-annotated specificity

The Data

Giving Advice

Domain: People writing advice for others

6 contexts, 3,289 observations

Concreteness: human-annotated specificity

The Data

Giving Advice

Domain: People writing advice for others

6 contexts, 3,289 observations

Concreteness: human-annotated specificity

Making Plans

Domain: Pre-course surveys at HarvardX

7 classes, 5,172 observations

Concreteness: Manipulated distance (short/long)

Concreteness: Annotated specificity (α = .68)

The Data

Giving Advice

Domain: People writing advice for others

6 contexts, 3,289 observations

Concreteness: human-annotated specificity

Making Plans

Domain: Pre-course surveys at HarvardX

7 classes, 5,172 observations

Concreteness: Manipulated distance (short/long)

Concreteness: Annotated specificity (α = .68)

Descriptions

Domain: Lab studies where participants describe things

3 studies, 1,319 observations

Concreteness: Manipulated distance (close/far, why/how)

The Measures

Word-Level

- Empirical word ratings by humans
- Many words, individual scores

The Measures

Word-Level

	Original MRC	mTurk Ratings	Bootstrapped MRC
This	240	2.14	212.36
example	--	3.03	335.35
sentence	--	3.57	397.16
has	267	2.18	272.31
both	322	2.97	256.11
concrete	562	4.59	506.81
and	220	1.52	277.14
abstract	--	1.45	373.73
words.	--	3.56	389.48

Size	~1,000	~40,000	~80,000
Source	Colthart, 1981	Brysbaert et al, 2014	Paetzold & Specia, 2017

The Measures

Word-Level

- Empirical word ratings by humans
- Many words, individual scores

Categorical

- Theoretically-driven word categories
- Few categories, simple weights

The Measures

Word-Level

- Empirical word ratings by humans
- Many words, individual scores

Categorical

"Linguistic Category Model"

Part of Speech	Score
Descriptive Action Verbs	5
Interpretive Action Verbs	4
State Verbs	3
Adjectives	2
Nouns	1

“Immediacy”

LIWC Category	Valence
1st Person Singular	1
Present-Tense Verbs	1
Discrepancies	1
Words >6 Letters	-1
Articles	-1

A Concrete Mega-Analysis

12 measures of concreteness
17 datasets across four domains
9,780 documents

Zero = < .03	Medium = .2 - .4
Very Low = .03 - .1	High = .4 - .6
Low = .1 - .2	Very High = >.6

	Name of Measure	Measurement Validity			
		Advice	Plan Distance	Plan Specificity	Describing
Word-Level	mTurk Ratings				
	Original MRC				
	Bootstrap MRC				
Categorical	Immediacy				
	Larrimore- LIWC				
	Pan-LIWC				
	Part-of- Speech LCM				
	Syntax LCM				
	DICTION				

A Concrete Mega-Analysis

12 measures of concreteness
17 datasets across four domains
9,780 documents

Zero = < .03	Medium = .2 - .4
Very Low = .03 - .1	High = .4 - .6
Low = .1 - .2	Very High = >.6

	Name of Measure	Measurement Validity			
		Advice	Plan Distance	Plan Specificity	Describing
Word-Level	mTurk Ratings	Low	Low	Low	Low
	Original MRC	Low	Low	Very Low	Medium
	Bootstrap MRC	Low	Low	Low	Low
Categorical	Immediacy	Zero	Very Low	Zero	Medium
	Larrimore- LIWC	Very Low	Very Low	Very Low	Zero
	Pan-LIWC	Zero	Very Low	Very Low	Zero
	Part-of- Speech LCM	Zero	Very Low	Zero	Medium
	Syntax LCM	Zero	Zero	Very Low	Low
	DICTION	Very Low	Zero	Zero	Very Low

A Concrete Mega-Analysis

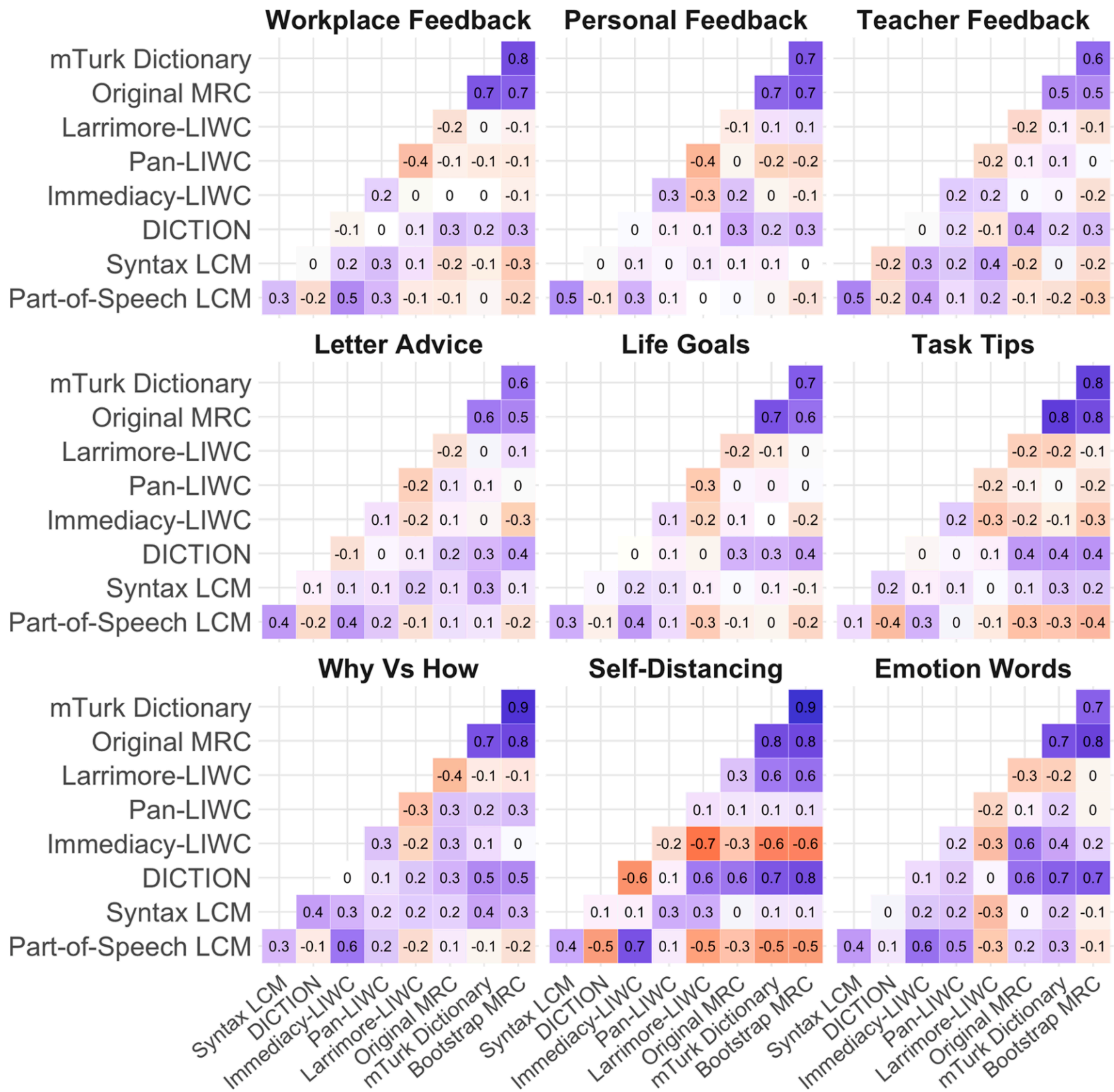
12 measures of concreteness
17 datasets across four domains
9,780 documents

Zero = < .03	Medium = .2 - .4
Very Low = .03 - .1	High = .4 - .6
Low = .1 - .2	Very High = >.6

	Name of Measure	Measurement Validity			
		Advice	Plan Distance	Plan Specificity	Describing
Word-Level	mTurk Ratings	Low	Low	Low	Low
	Original MRC	Low	Low	Very Low	Medium
	Bootstrap MRC	Low	Low	Low	Low
$95 for LIWC	Immediacy	Zero	Very Low	Zero	Medium
	Larrimore- LIWC	Very Low	Very Low	Very Low	Zero
Categorical	Pan-LIWC	Zero	Very Low	Very Low	Zero
	Part-of- Speech LCM	Zero	Very Low	Zero	Medium
	Syntax LCM	Zero	Zero	Very Low	Low
$219 for DICTION	DICTION	Very Low	Zero	Zero	Very Low

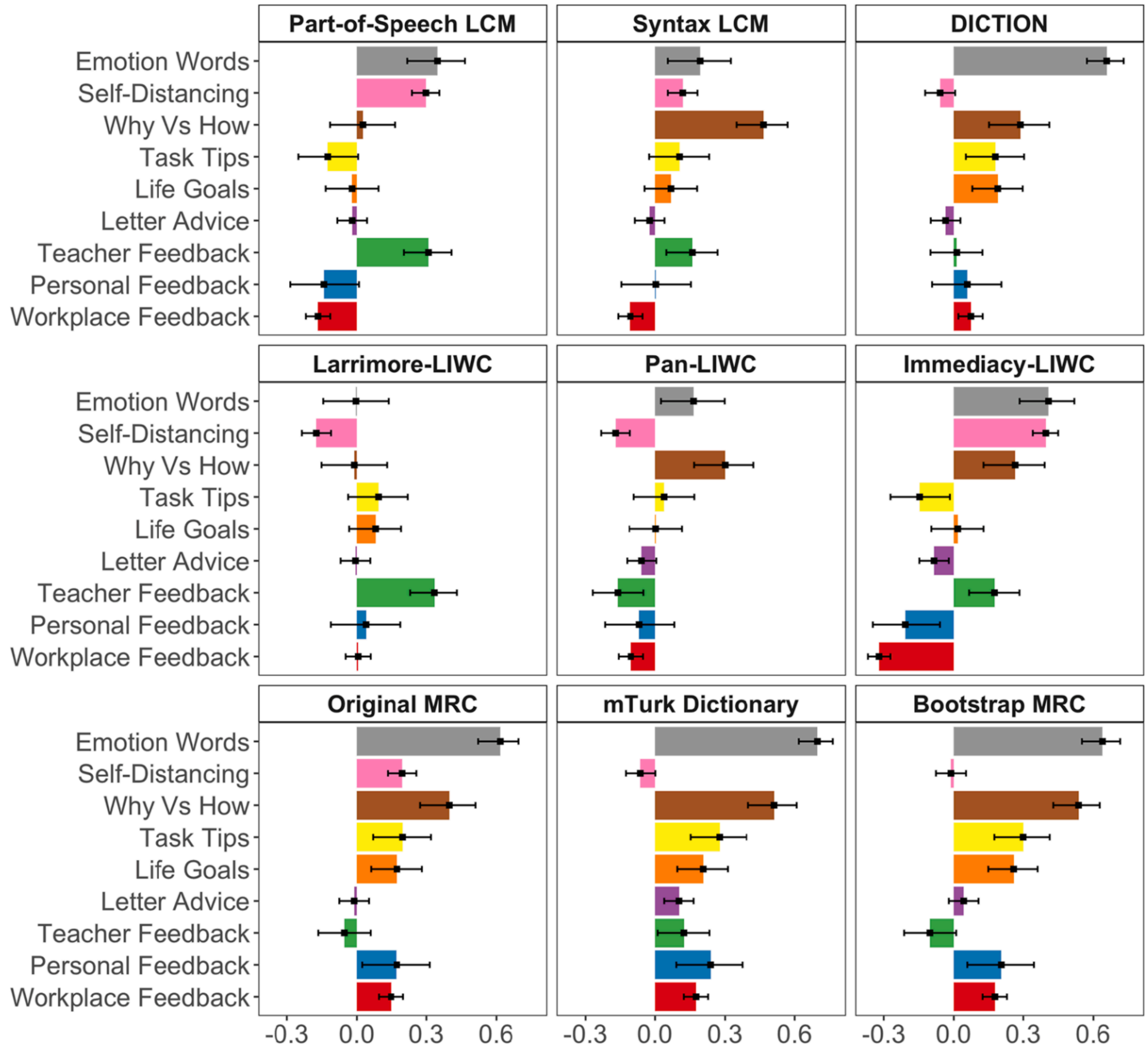
A Concrete Mega-Analysis

Concreteness dictionaries do not correlate with each other



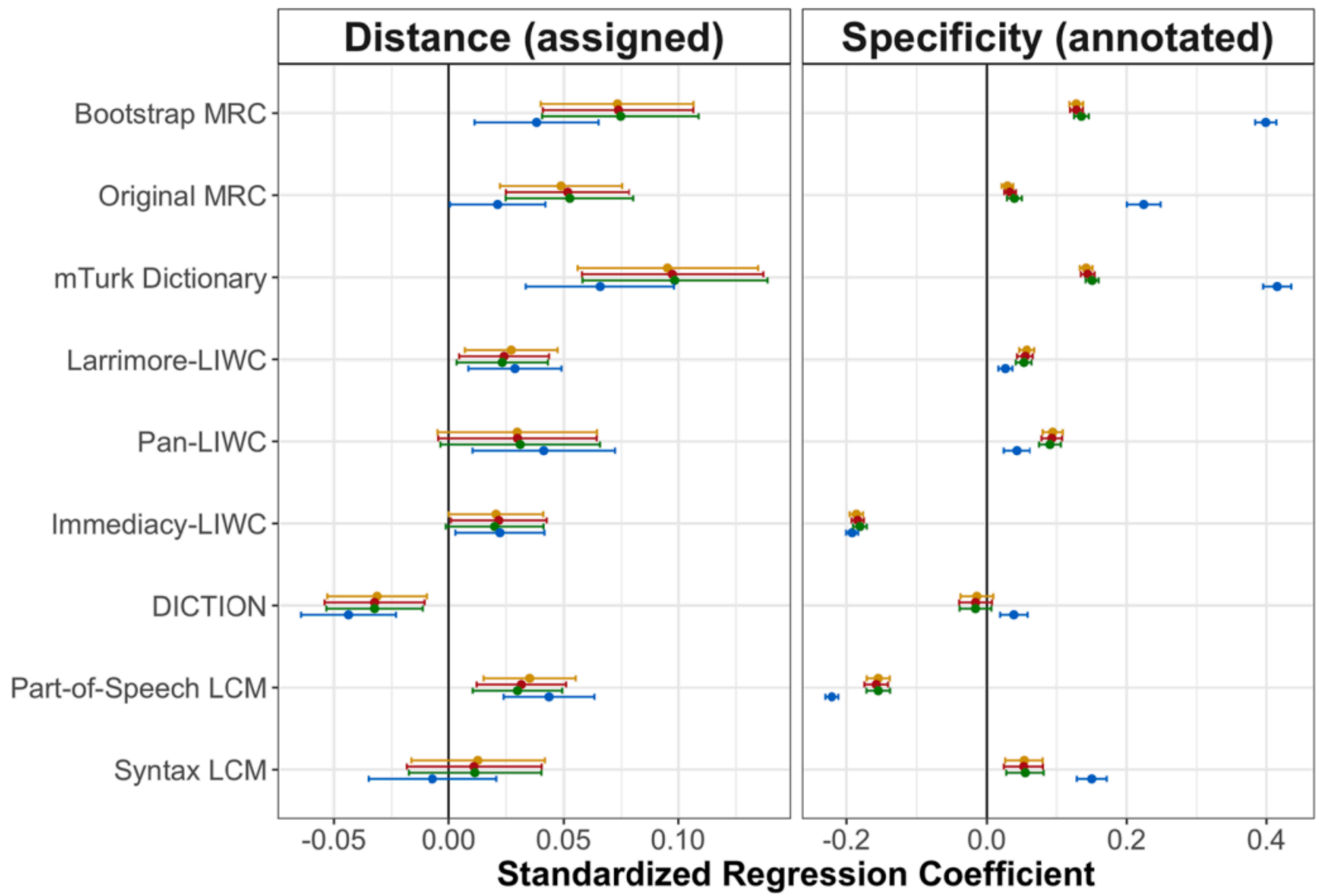
A Concrete Mega-Analysis

Concreteness dictionaries don't correlate with concreteness



A Concrete Mega-Analysis

Concreteness dictionaries don't correlate with concreteness



Supervised NLP

Supervised NLP

	Test Dataset
Training Dataset	Advice
Advice	.228 [.195, .260]
Plan Distance	
Plan Specificity	
Describing	

Supervised NLP

	Test Dataset
Training Dataset	**Advice**
Advice	**.228** **[.195, .260]**
Plan Distance	.022 [-.012, .056]
Plan Specificity	.191 [.158, .224]
Describing	.119 [.085, .152]

Supervised NLP

	Test Dataset		
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**
Advice	**.228** [.195, .260]		
Plan Distance	.022 [-.012, .056]	**.339** [.315, .363]	
Plan Specificity	.191 [.158, .224]		**.733** [.720, .745]
Describing	.119 [.085, .152]		

Supervised NLP

	Test Dataset		
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**
Advice	**.228** **[.195, .260]**	.004 [-.024, .031]	.258 [.232, .283]
Plan Distance	.022 [-.012, .056]	**.339** **[.315, .363]**	.026 [-.001, .053]
Plan Specificity	.191 [.158, .224]	.038 [.011, .065]	**.733** **[.720, .745]**
Describing	.119 [.085, .152]	.012 [-.015, .039]	.417 [.394, .439]

Supervised NLP

	Test Dataset		
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**
Advice	**.228** **[.195, .260]**	.004 [-.024, .031]	.258 [.232, .283]
Plan Distance	.022 [-.012, .056]	**.339** **[.315, .363]**	.026 [-.001, .053]
Plan Specificity	.191 [.158, .224]	.038 [.011, .065]	**.733** **[.720, .745]**
Describing	.119 [.085, .152]	.012 [-.015, .039]	.417 [.394, .439]

Best Previous	.155 [.122, .188]	.047 [.020, .075]	.488 [.466, .510]

Supervised NLP

	Test Dataset			
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**	**Describing**
Advice	**.228** **[.195, .260]**	.004 [-.024, .031]	.258 [.232, .283]	
Plan Distance	.022 [-.012, .056]	**.339** **[.315, .363]**	.026 [-.001, .053]	
Plan Specificity	.191 [.158, .224]	.038 [.011, .065]	**.733** **[.720, .745]**	
Describing	.119 [.085, .152]	.012 [-.015, .039]	.417 [.394, .439]	**.092** **[.038, .145]**

Best Previous	.155 [.122, .188]	.047 [.020, .075]	.488 [.466, .510]

Supervised NLP

	Test Dataset			
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**	**Describing**
Advice	**.228** **[.195, .260]**	.004 [-.024, .031]	.258 [.232, .283]	
Plan Distance	.022 [-.012, .056]	**.339** **[.315, .363]**	.026 [-.001, .053]	
Plan Specificity	.191 [.158, .224]	.038 [.011, .065]	**.733** **[.720, .745]**	
Describing	.119 [.085, .152]	.012 [-.015, .039]	.417 [.394, .439]	**.092** **[.038, .145]**

Best Previous	.155 [.122, .188]	.047 [.020, .075]	.488 [.466, .510]	.363 [.315, .409]

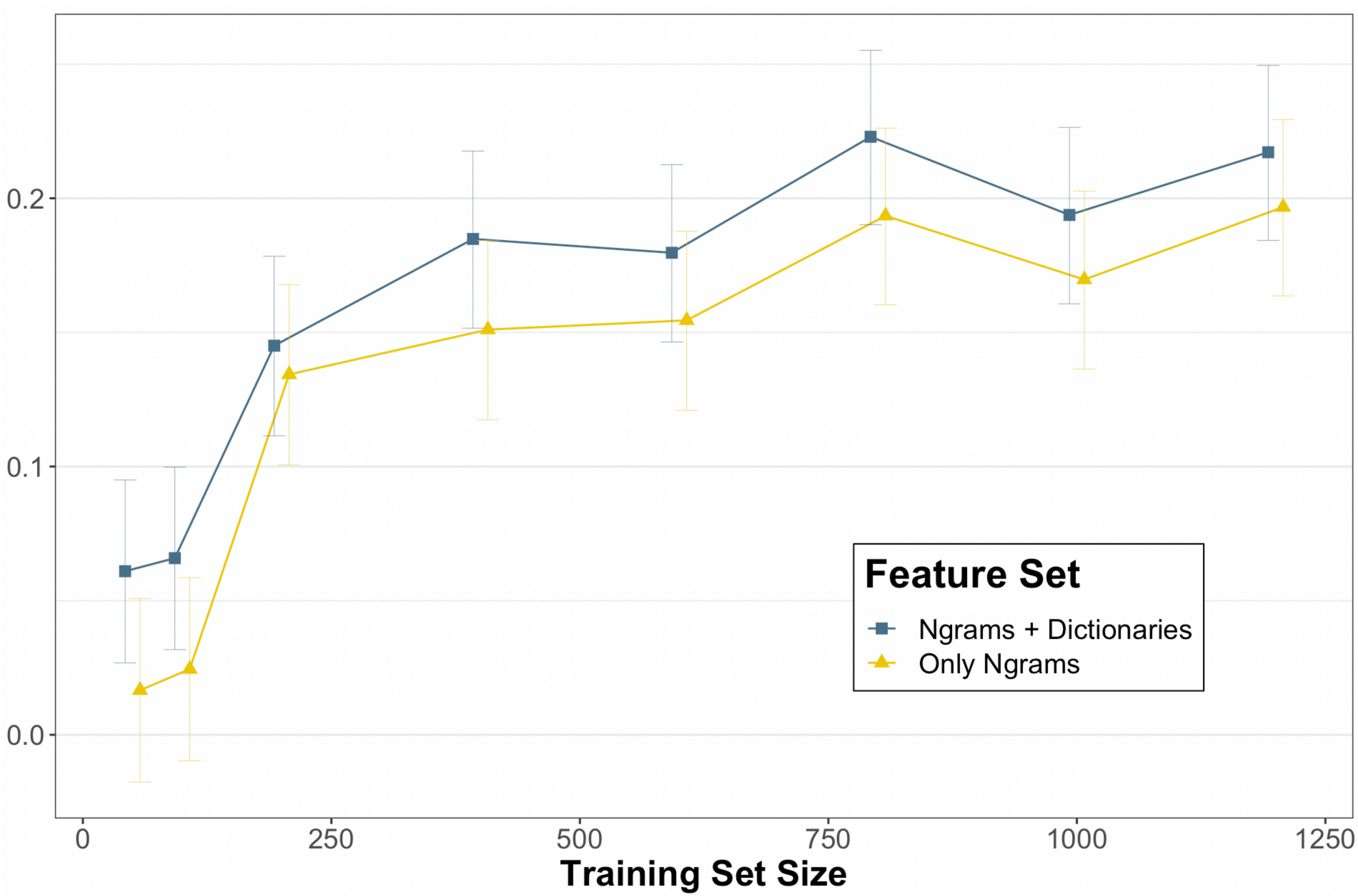
Supervised NLP

	Test Dataset			
Training Dataset	**Advice**	**Plan Distance**	**Plan Specificity**	**Describing**
Advice	**.228** **[.195, .260]**	.004 [-.024, .031]	.258 [.232, .283]	-.113 [-.166, -.059]
Plan Distance	.022 [-.012, .056]	**.339** **[.315, .363]**	.026 [-.001, .053]	-.012 [-.066, .042]
Plan Specificity	.191 [.158, .224]	.038 [.011, .065]	**.733** **[.720, .745]**	-.032 [-.086, .022]
Describing	.119 [.085, .152]	.012 [-.015, .039]	.417 [.394, .439]	**.092** **[.038, .145]**

Best Previous	.155 [.122, .188]	.047 [.020, .075]	.488 [.466, .510]	.363 [.315, .409]

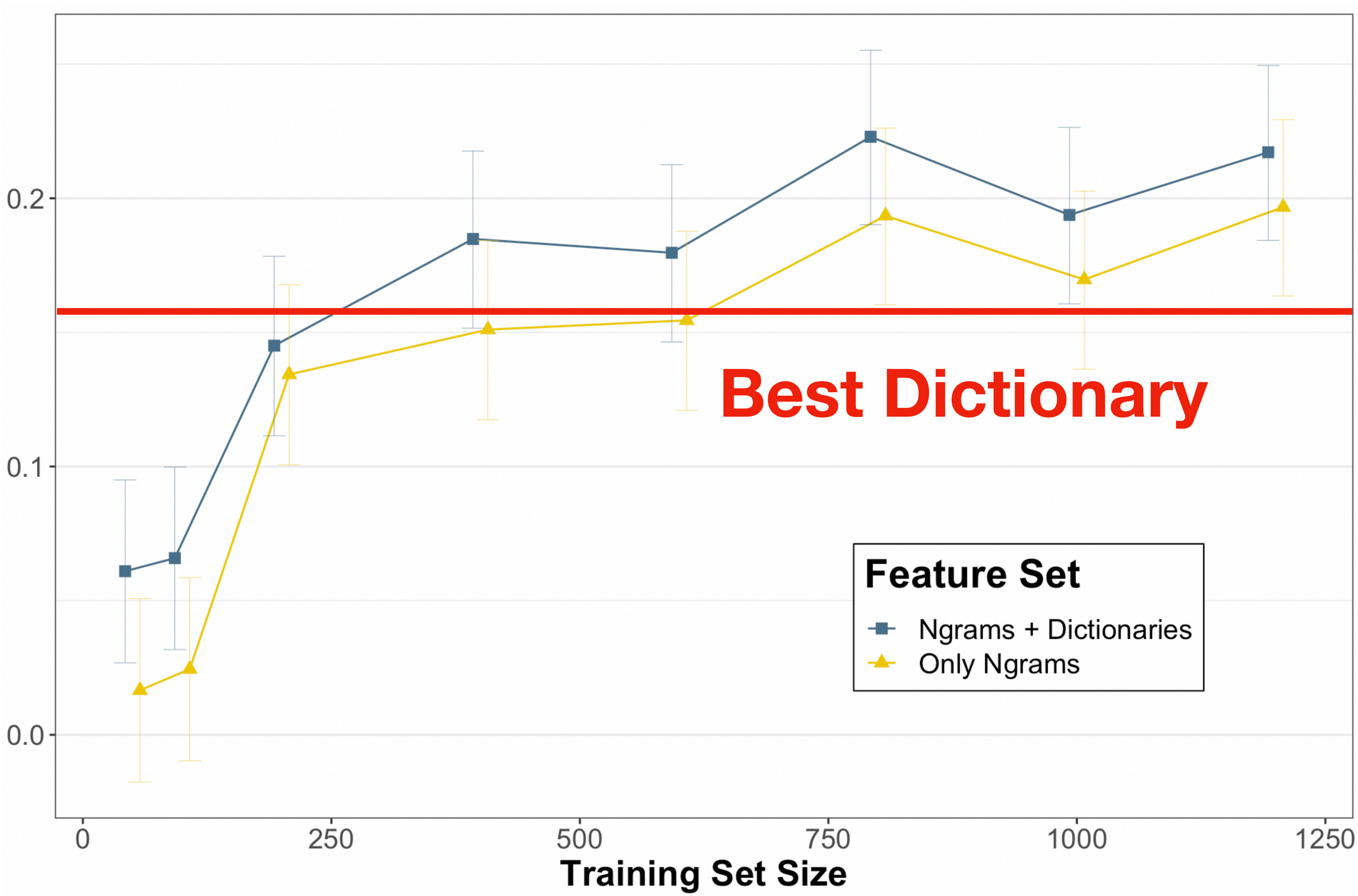
How Many Annotations?

Advice Specificity



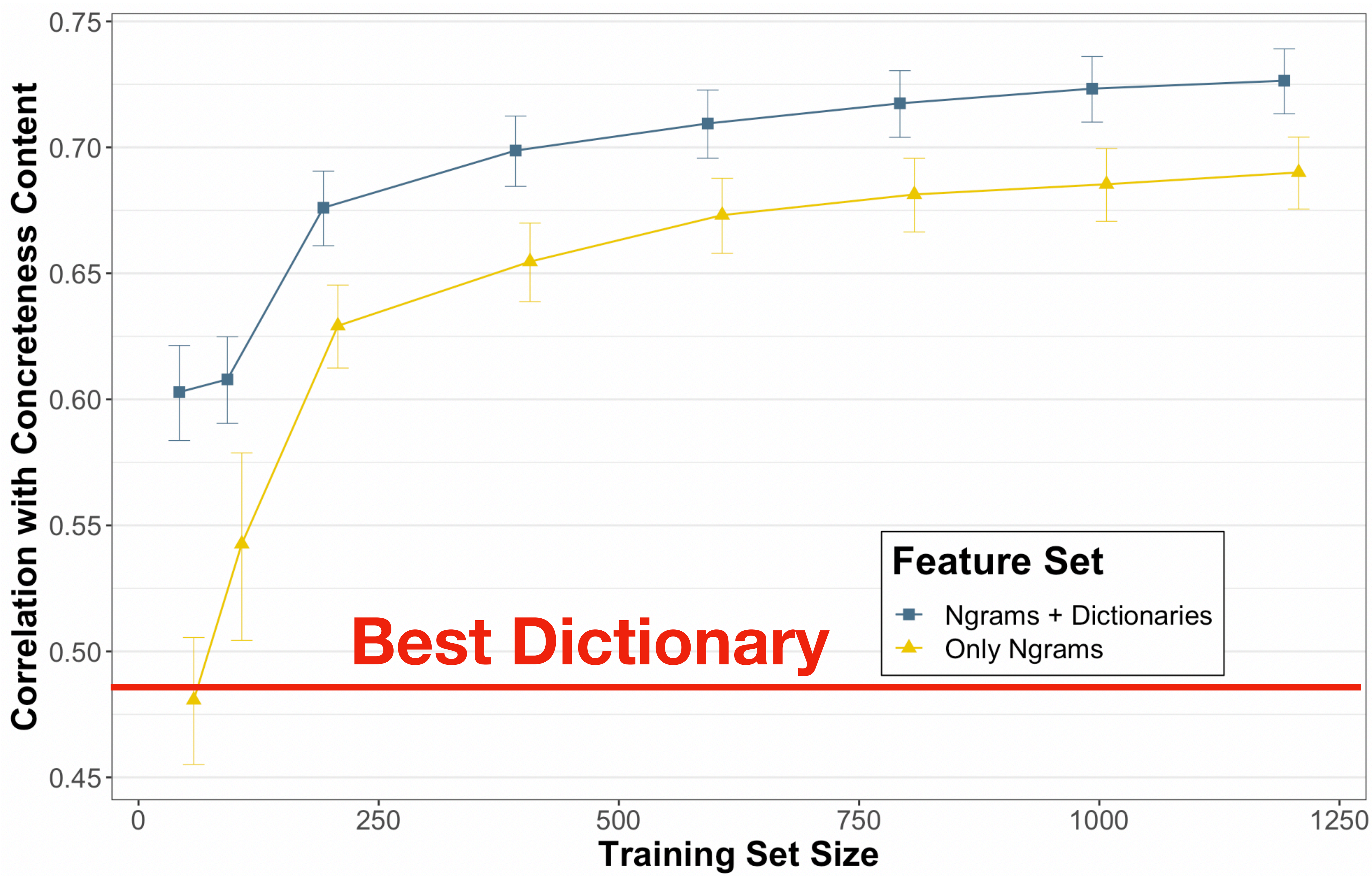
How Many Annotations?

Advice Specificity

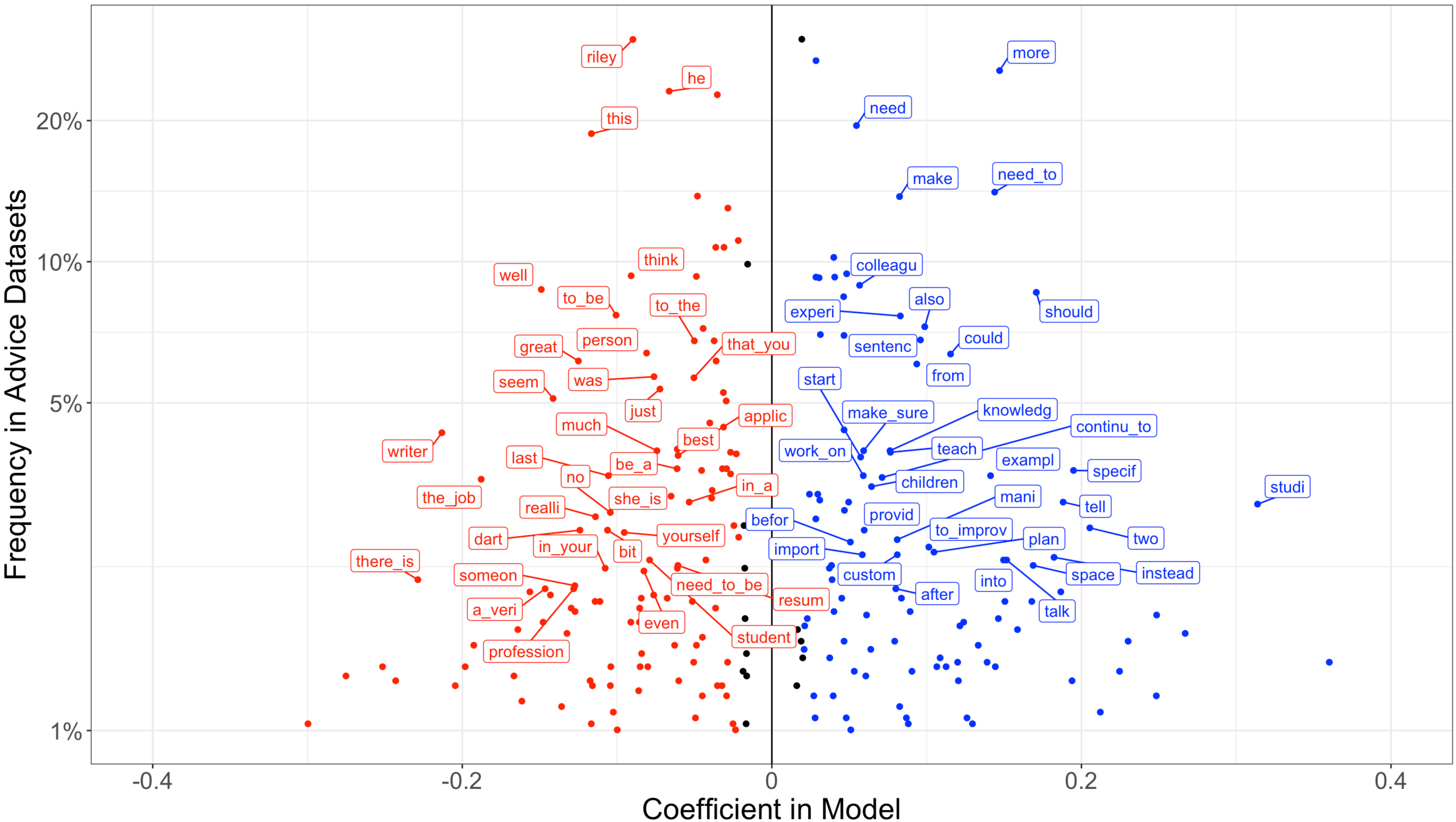


How Many Annotations?

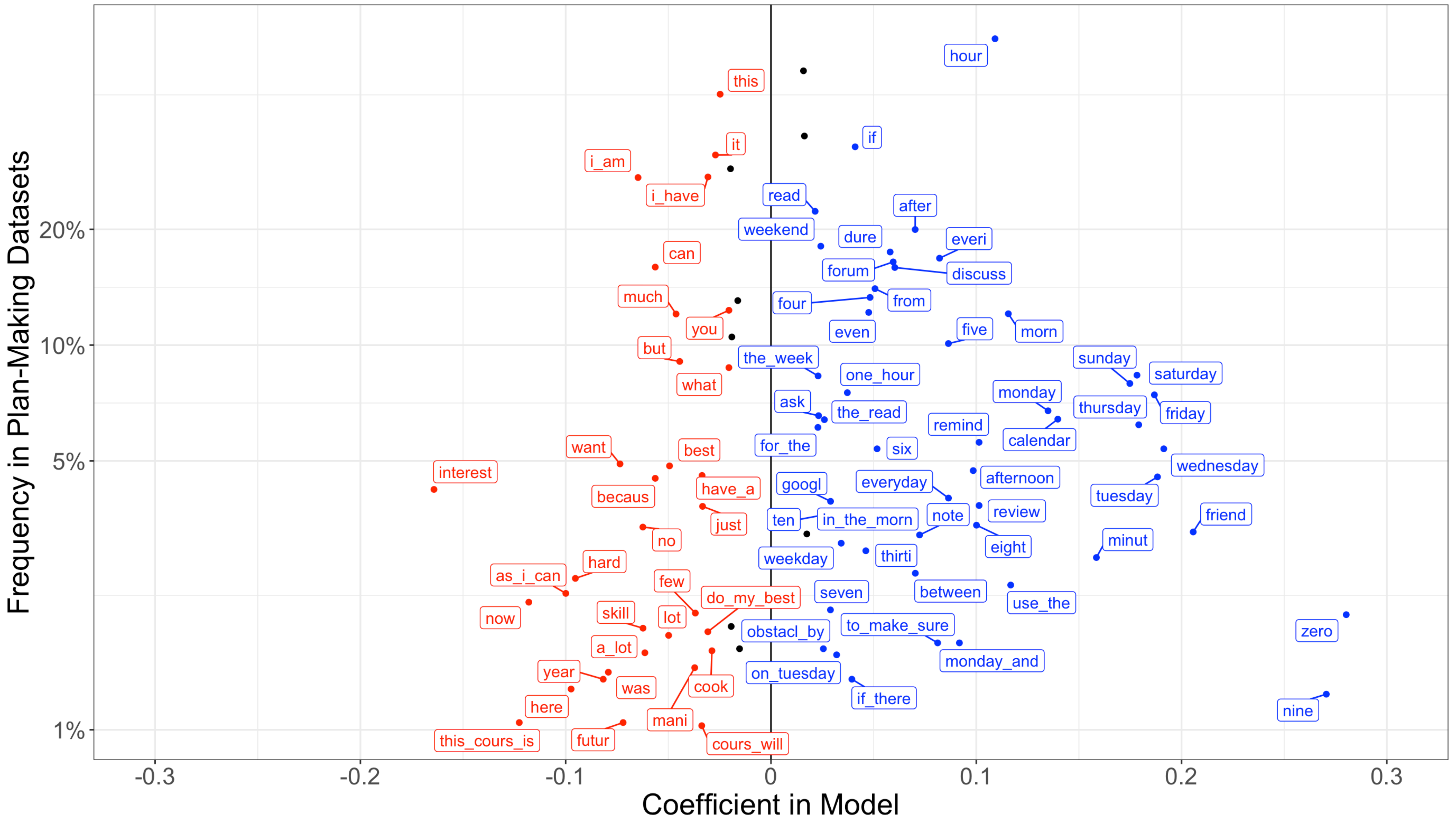
Planning Specificity



Advice Model



Plan Specificity Model



Dictionaries

Dictionaries

Advantages

Easy to implement

Some dictionaries are validated

Limitations

Dictionaries

Advantages

Easy to implement

Some dictionaries are validated

Limitations

No use of word order.   e.g. “bad” vs. “not bad”

Dictionaries

Advantages

Easy to implement

Some dictionaries are validated

Limitations

No use of word order.   e.g. “bad” vs. “not bad”

Interpretation can be muddy   e.g. “authenticity” dictionary

Dictionaries

Advantages

Easy to implement

Some dictionaries are validated

Limitations

No use of word order.   e.g. “bad” vs. “not bad”

Interpretation can be muddy   e.g. “authenticity” dictionary

Assumes context universality

Dictionaries

Advantages

Easy to implement

Some dictionaries are validated

Limitations

No use of word order.   e.g. “bad” vs. “not bad”

Interpretation can be muddy   e.g. “authenticity” dictionary

Assumes context universality

Blurry distinction between style vs. content

Dimension Reduction

Theory-Driven - *"We'll tell you what a word means"*

Dictionaries

Data-Driven - *"You will know a word by the company it keeps"*

Two senses of similarity:

Syntagmatic - often found in the same documents

"GDP", "inflation", "economy", "currency"

e.g. topic models (LDA), Latent Semantic Analysis

Paradigmatic - often have the same neighbours

"wrote", "remarked", "said", "added"

e.g. word2vec, GloVe, BERT, GPT-1/2/3

Word Similarity

Many words have similar meanings or functions

How to reduce?

You will know a word by the company it keeps

Word Similarity

Many words have similar meanings or functions

How to reduce?

You will know a word by the company it keeps

Two senses of similarity:

Syntagmatic - often found near one another

"GDP", "inflation", "economy", "currency"

e.g. Latent Semantic Analysis, Latent Dirichlet Allocation

Word Similarity

Many words have similar meanings or functions

How to reduce?

You will know a word by the company it keeps

Two senses of similarity:

Syntagmatic - often found near one another

“GDP”, “inflation”, “economy”, “currency”

e.g. Latent Semantic Analysis, Latent Dirichlet Allocation

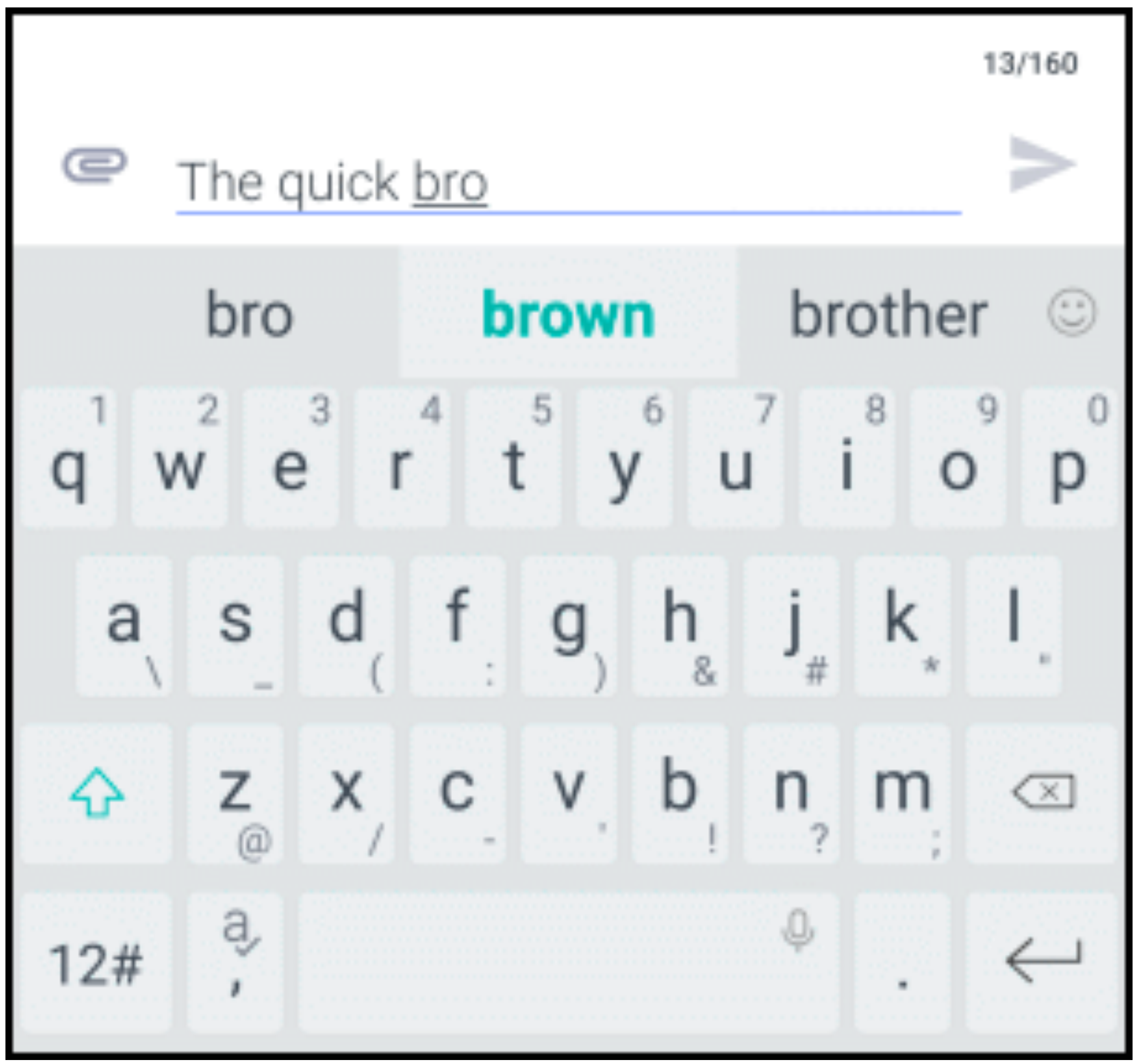
Paradigmatic - often have the same neighbours

“wrote”, “remarked”, “said”, “added”

e.g. word2vec, GloVe

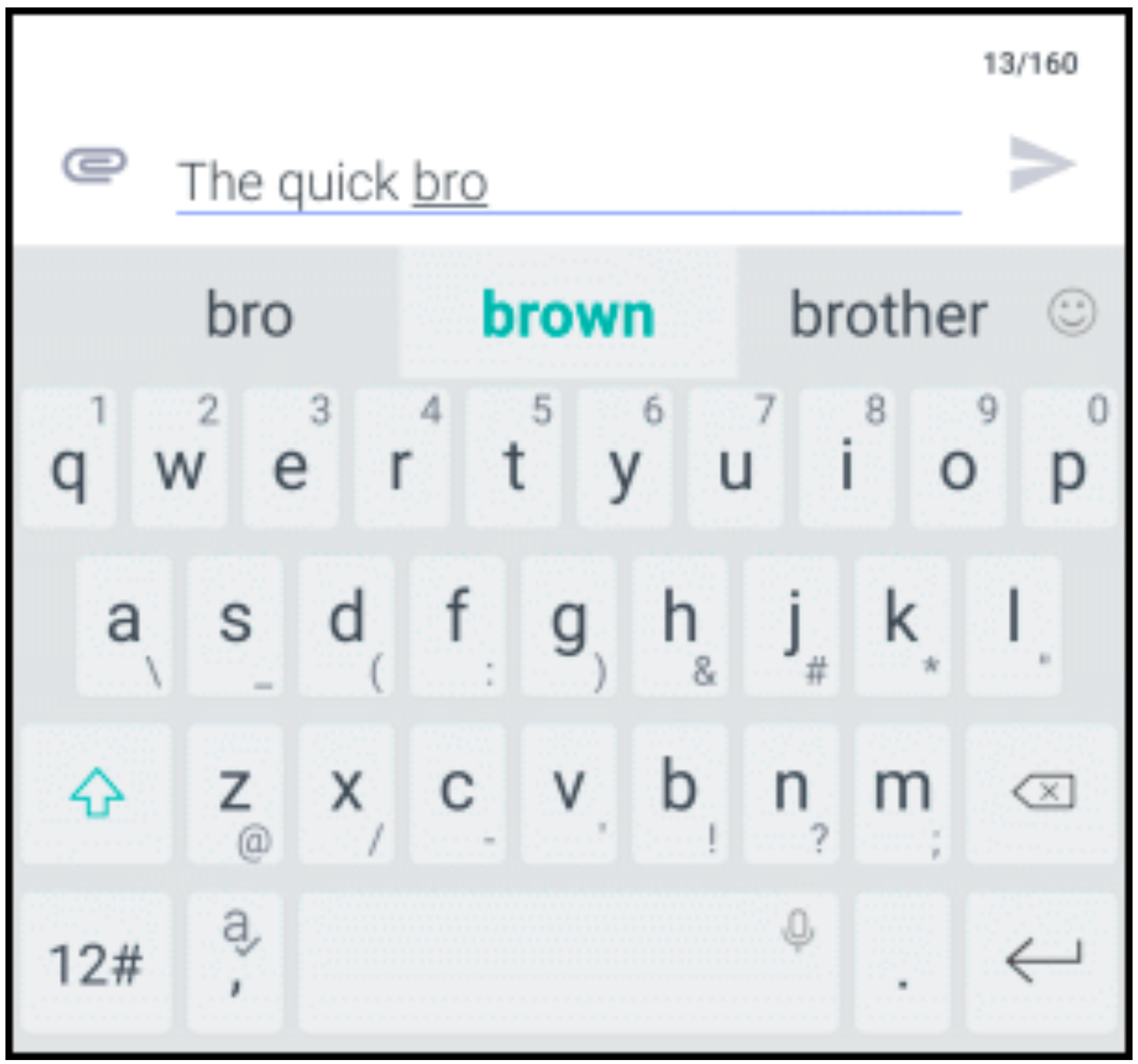
Language Modeling Task

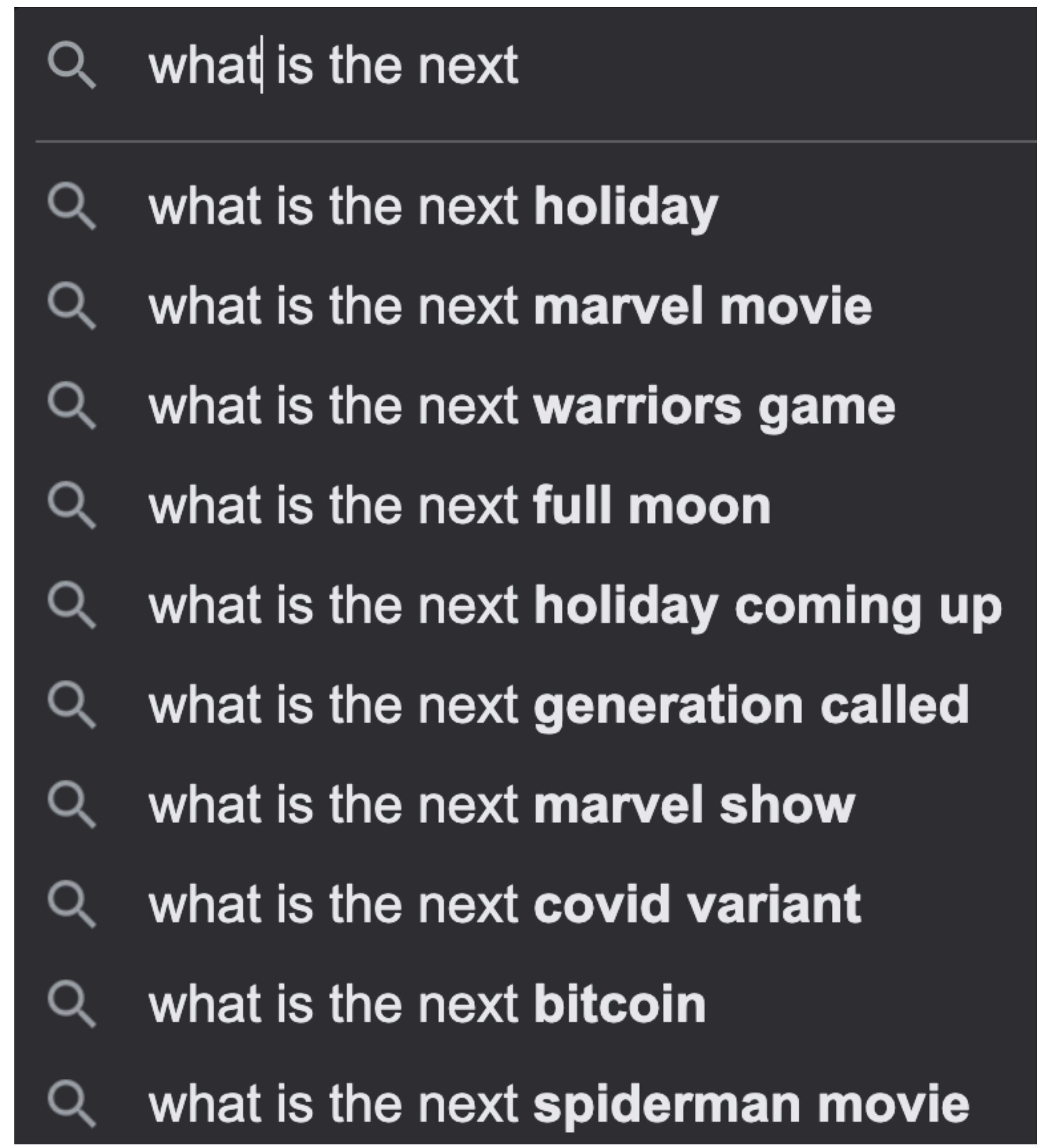
Predictive Text



Language Modeling Task

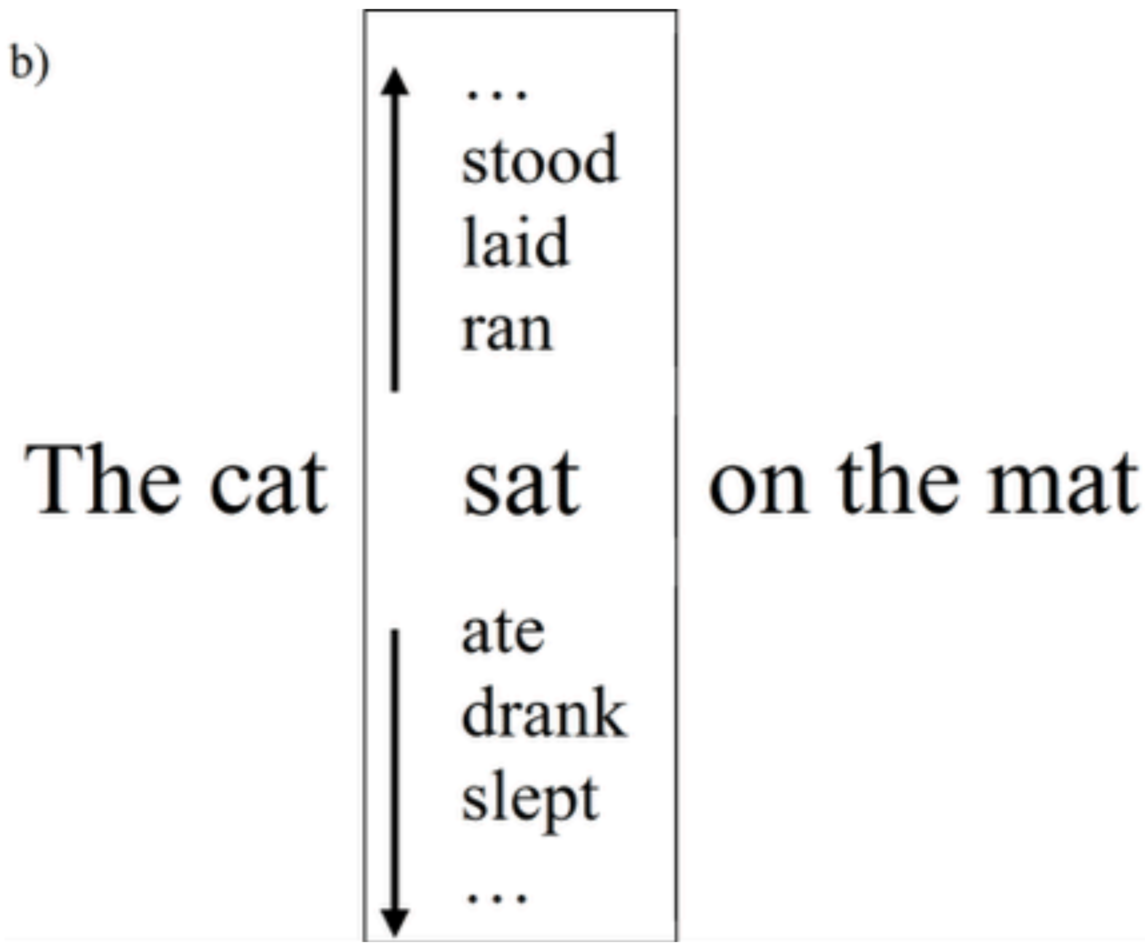
Predictive Text





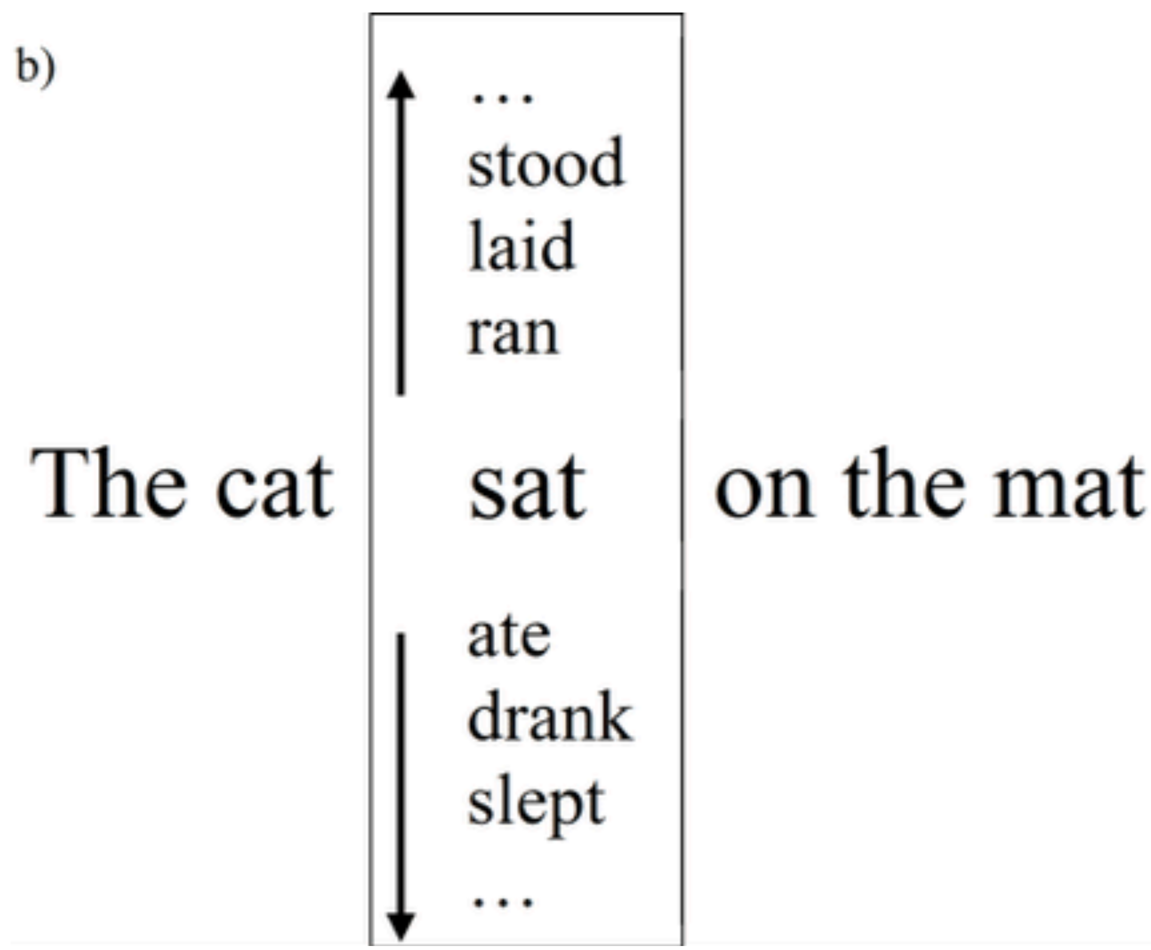
Language Modeling Task

Continuous Bag of Words



Language Modeling Task

Continuous Bag of Words

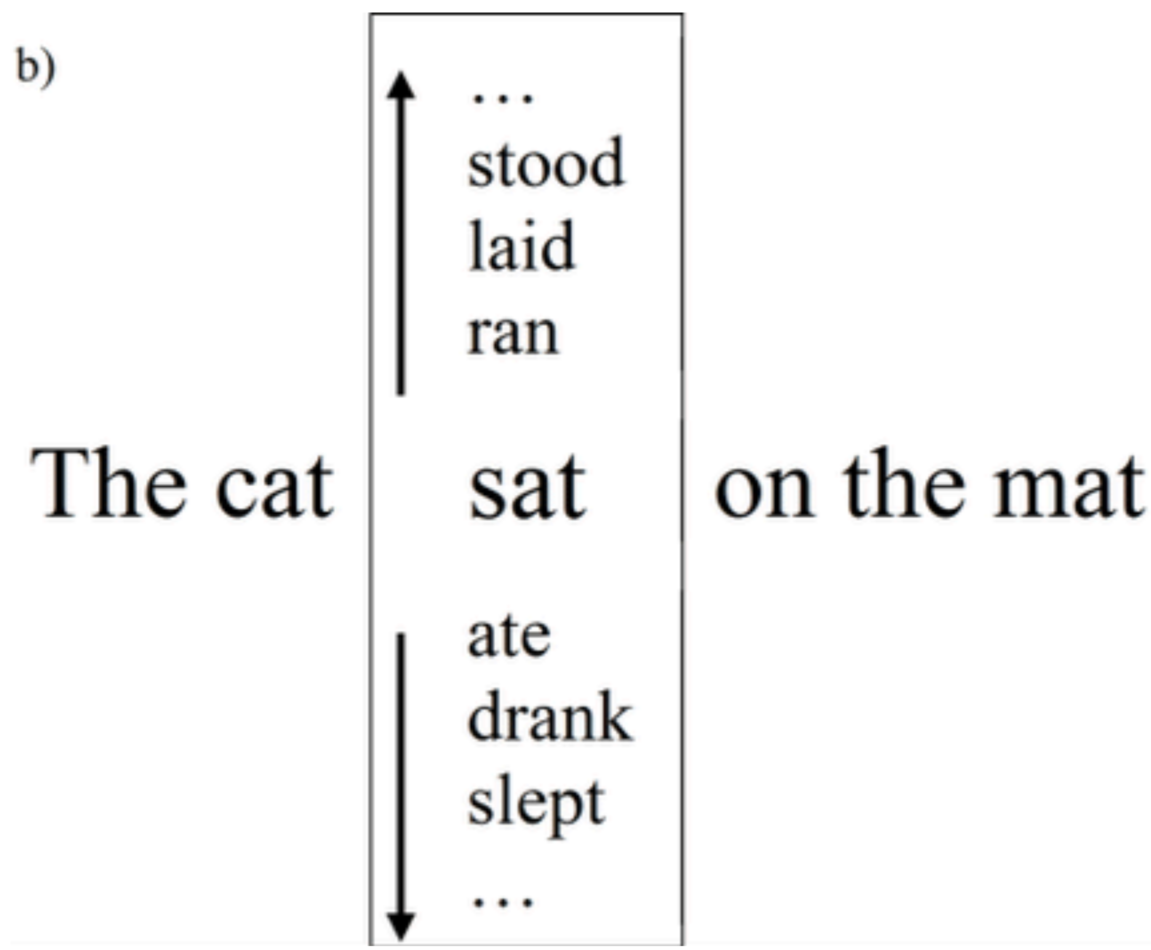


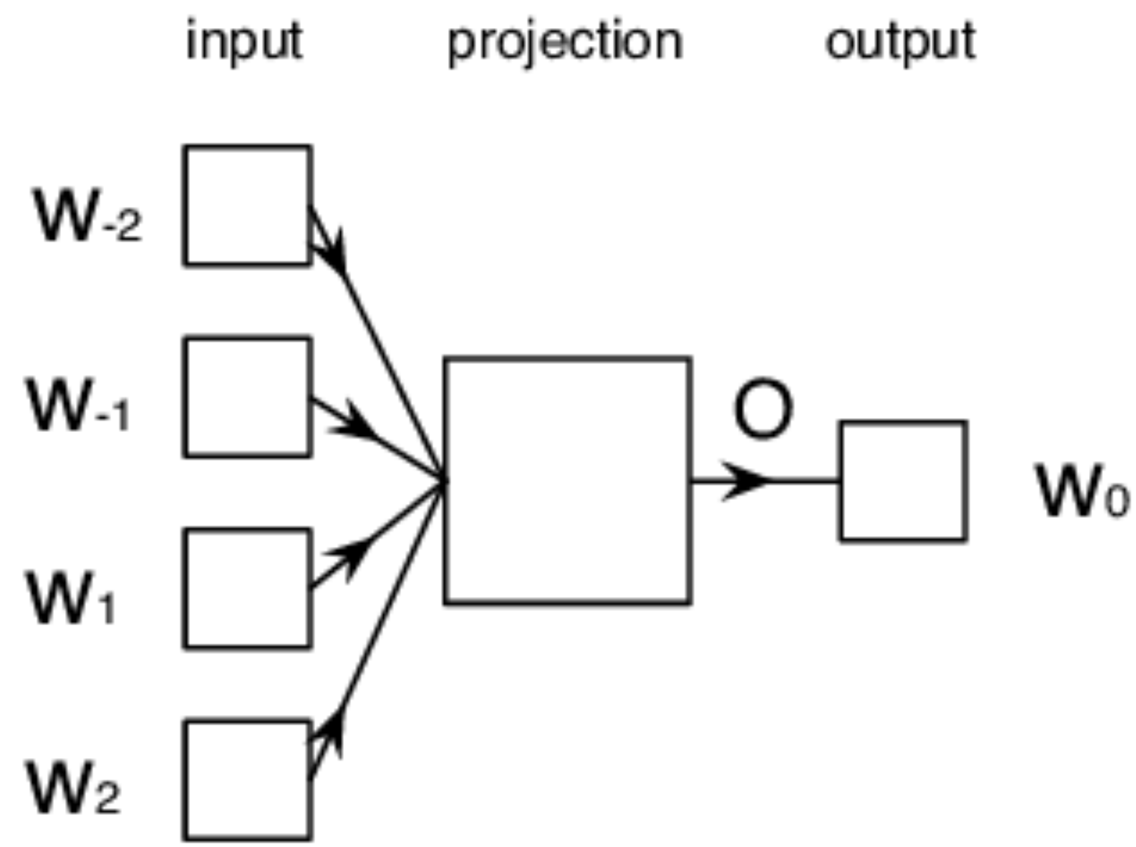
Predict the context words using the masked word

"Self-Supervised Learning"

Language Modeling Task

Continuous Bag of Words



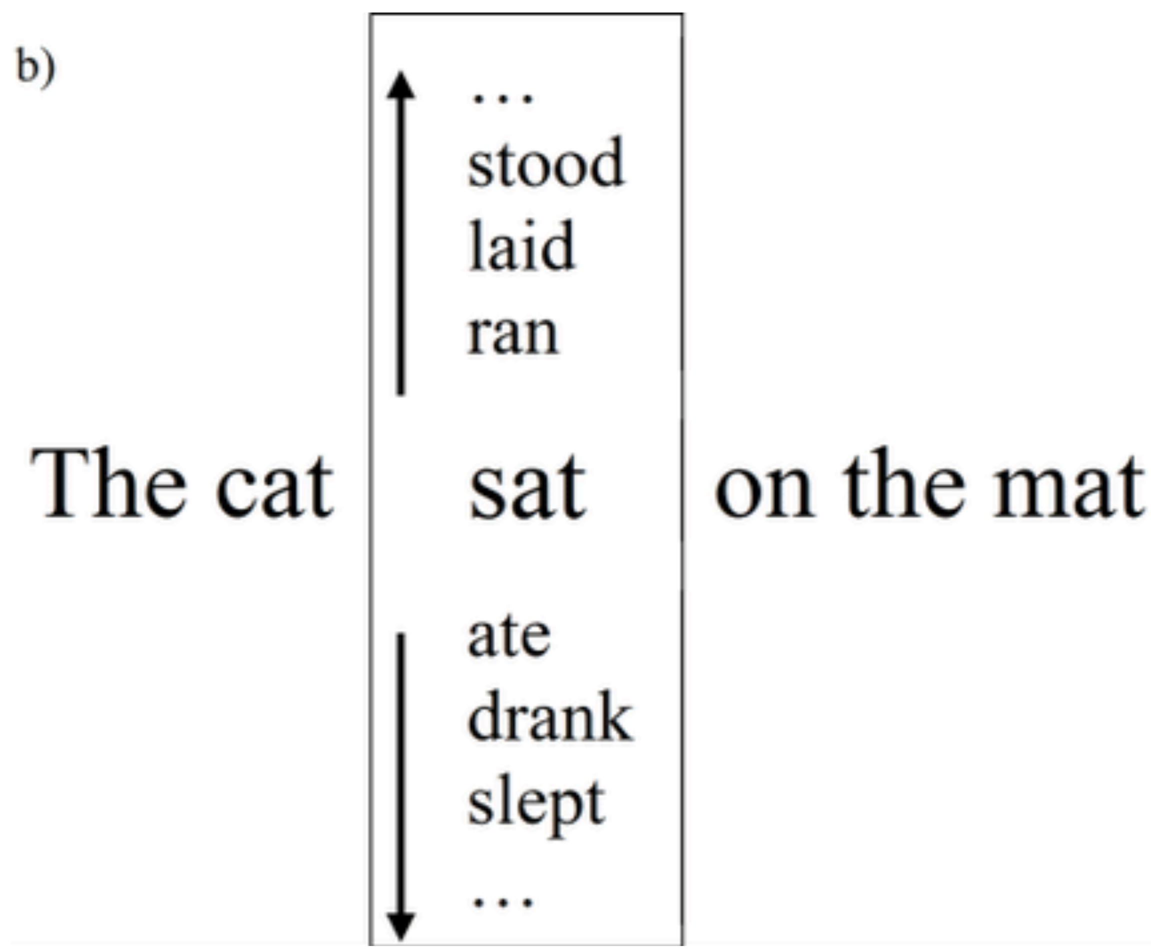


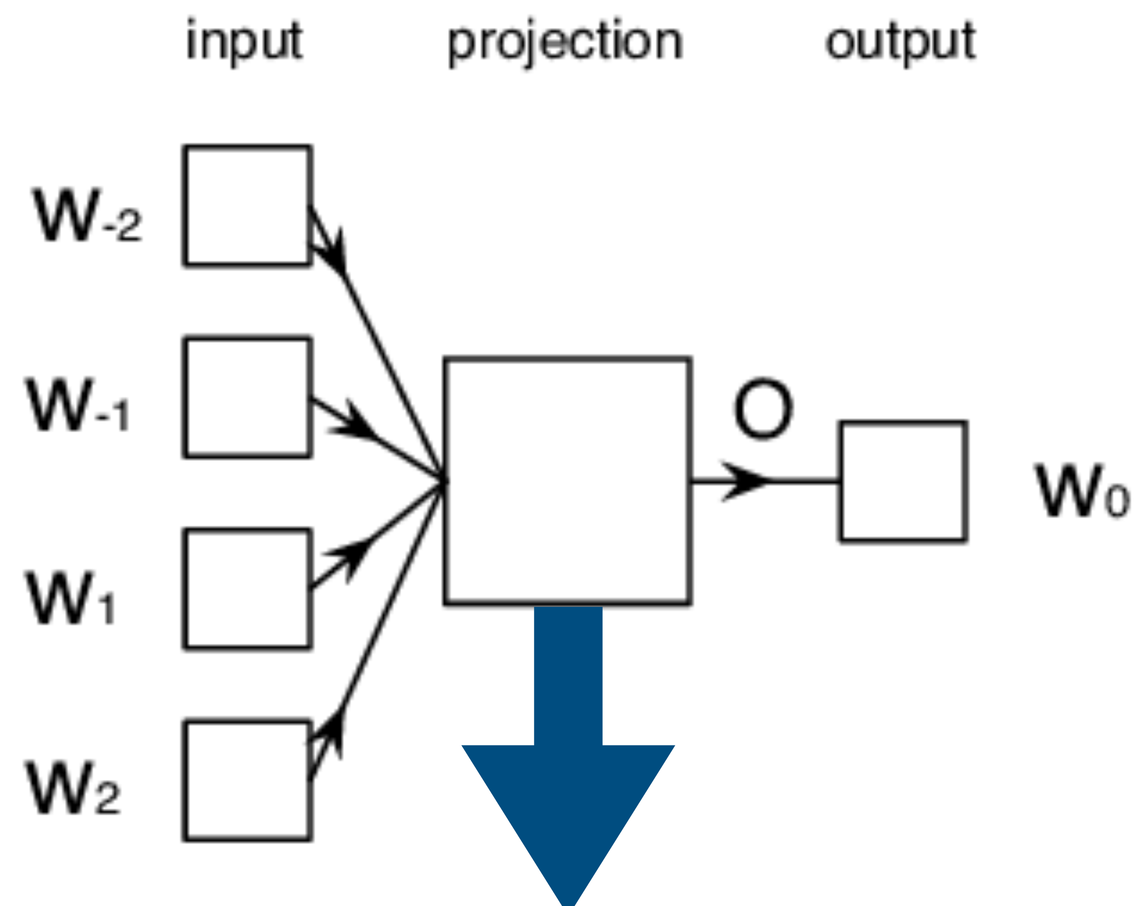
Predict the context words using the masked word

"Self-Supervised Learning"

Language Modeling Task

Continuous Bag of Words





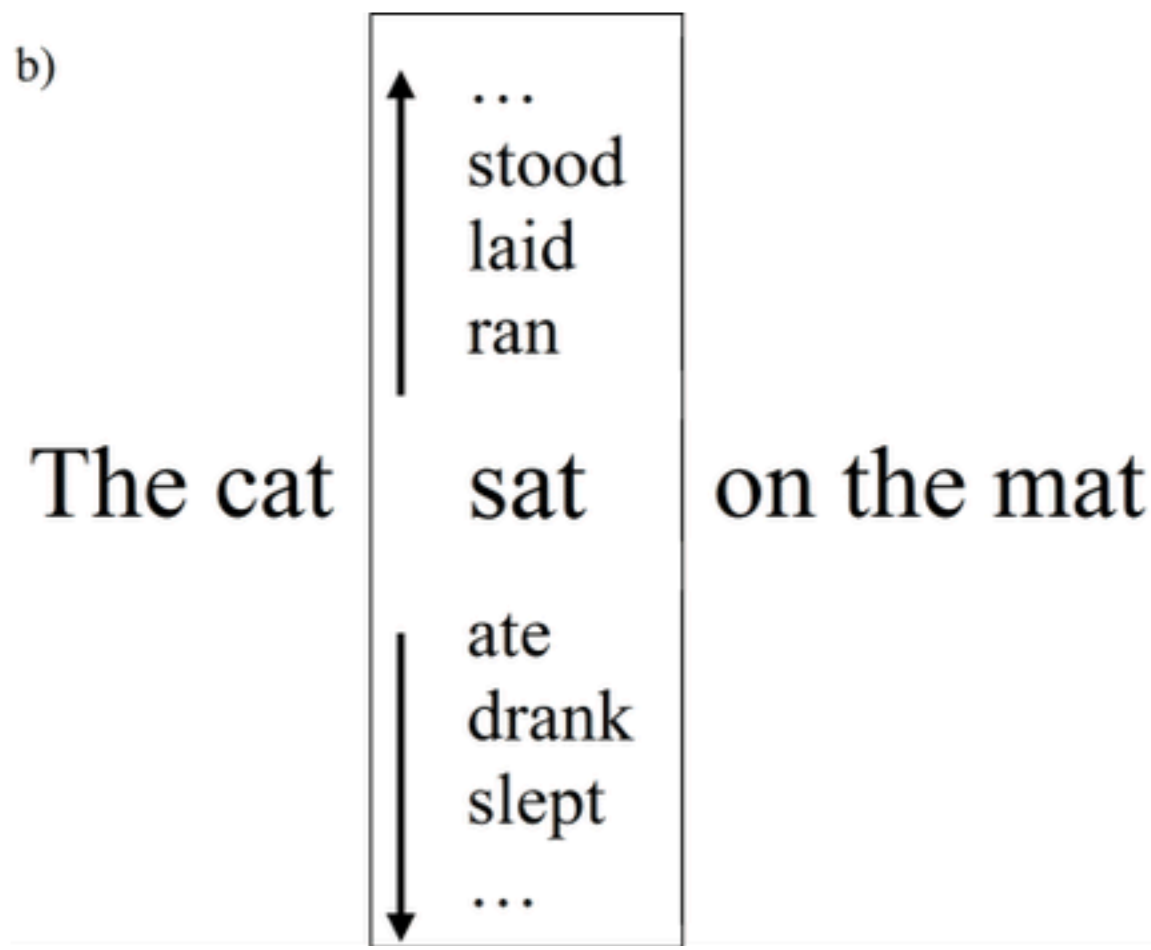
	vec1	vec2	vec3	vec4	vec5	vec6
aardvark	-0.52	-0.49	-0.70	-0.02	0.11	0.37
Aaron	-0.40	-0.86	-0.40	0.32	0.01	0.40
able	-0.46	-0.85	-0.35	0.33	-0.13	0.57

Predict the context words using the masked word

"Self-Supervised Learning"

Language Modeling Task

Skip-grams

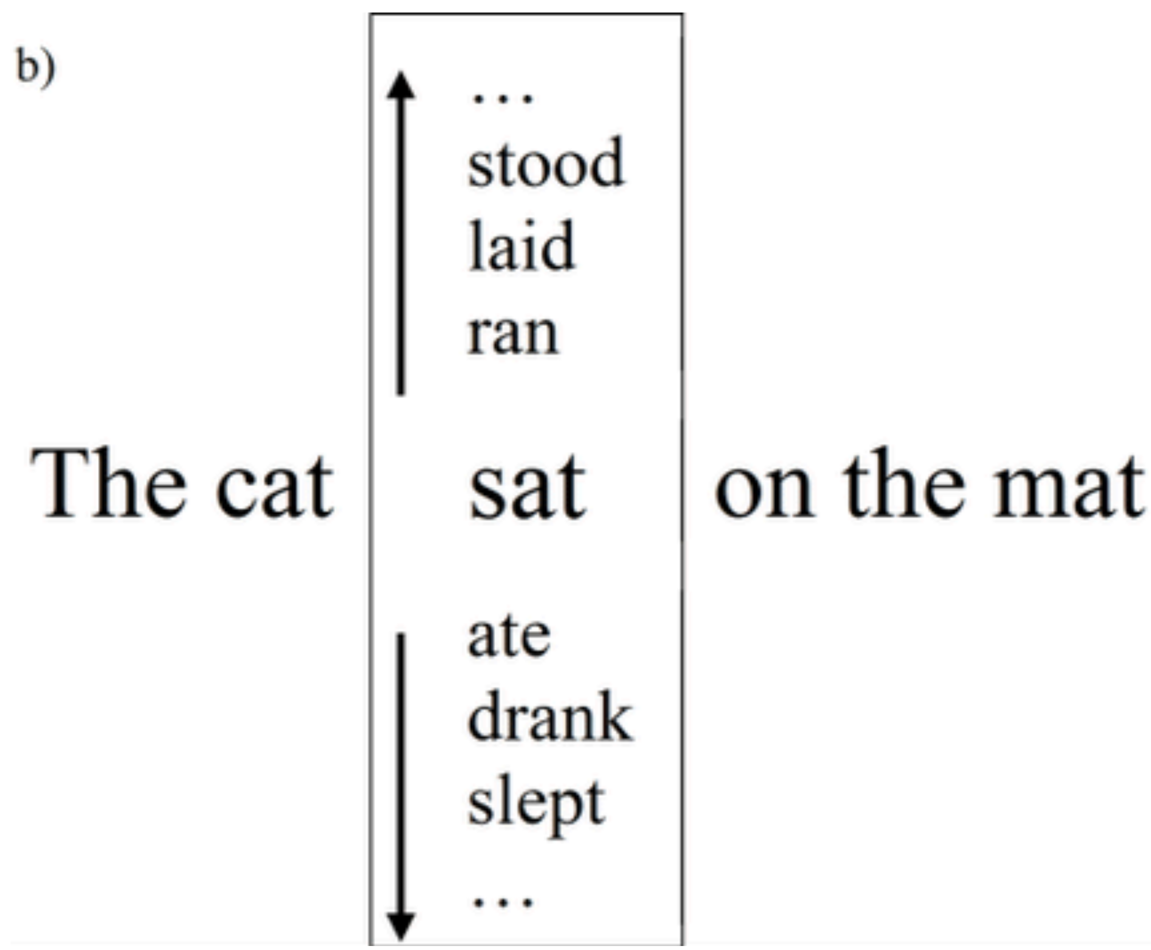


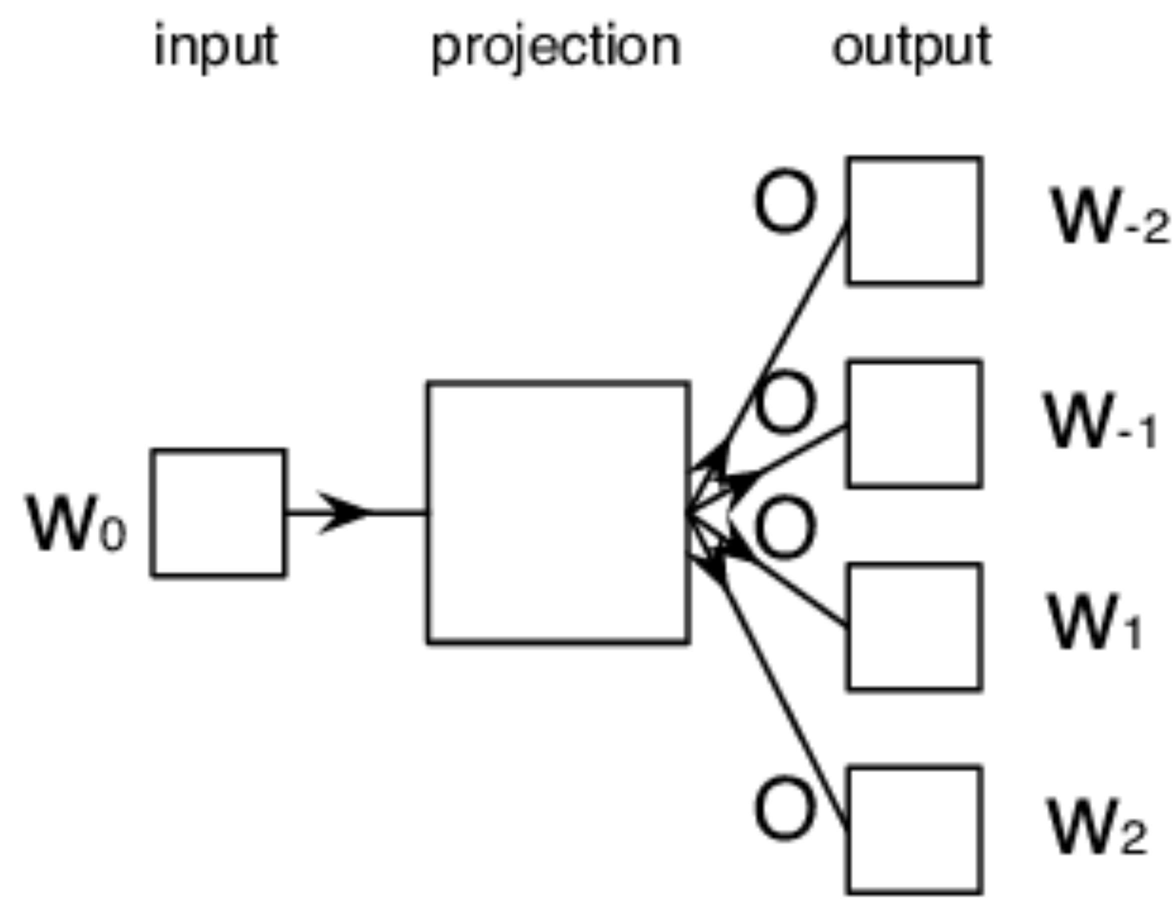
Predict the context words using the masked word

"Self-Supervised Learning"

Language Modeling Task

Skip-grams



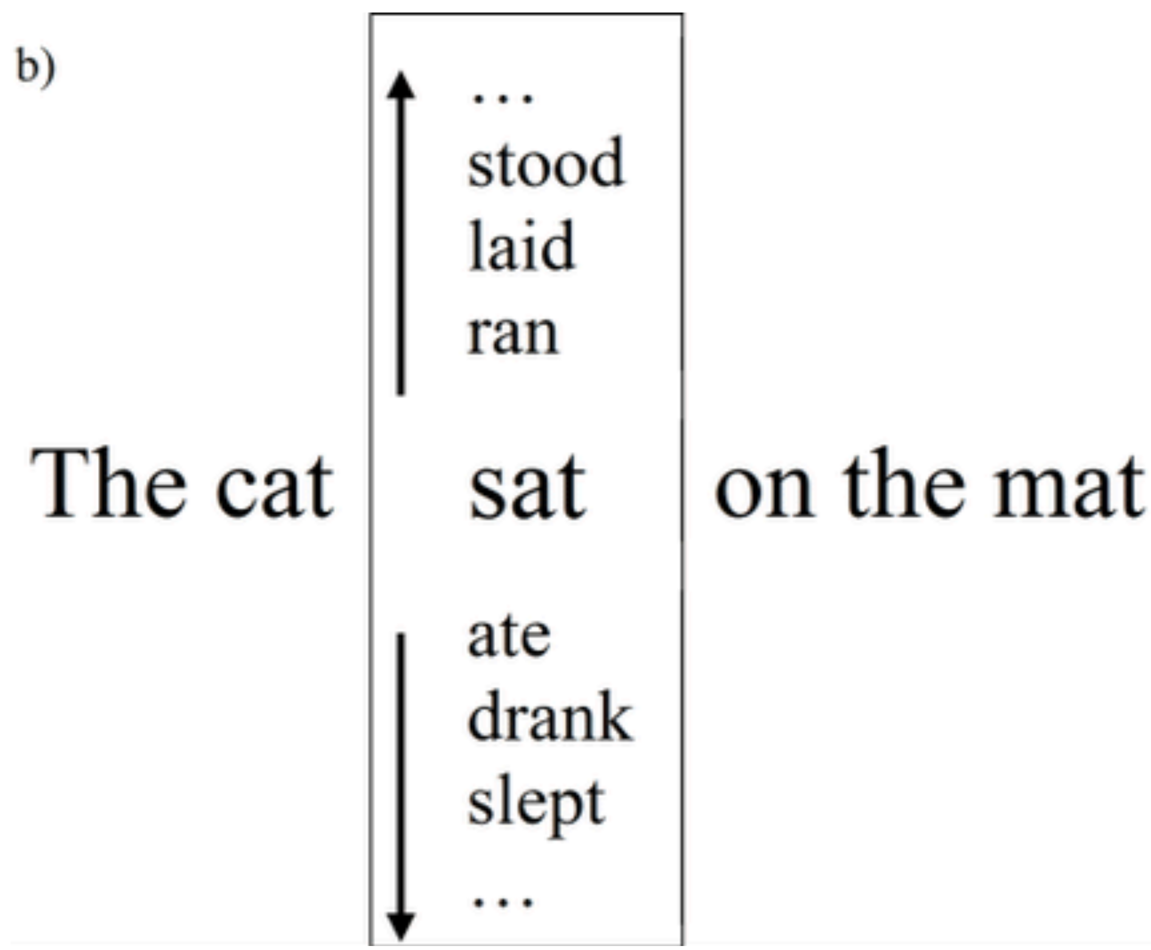


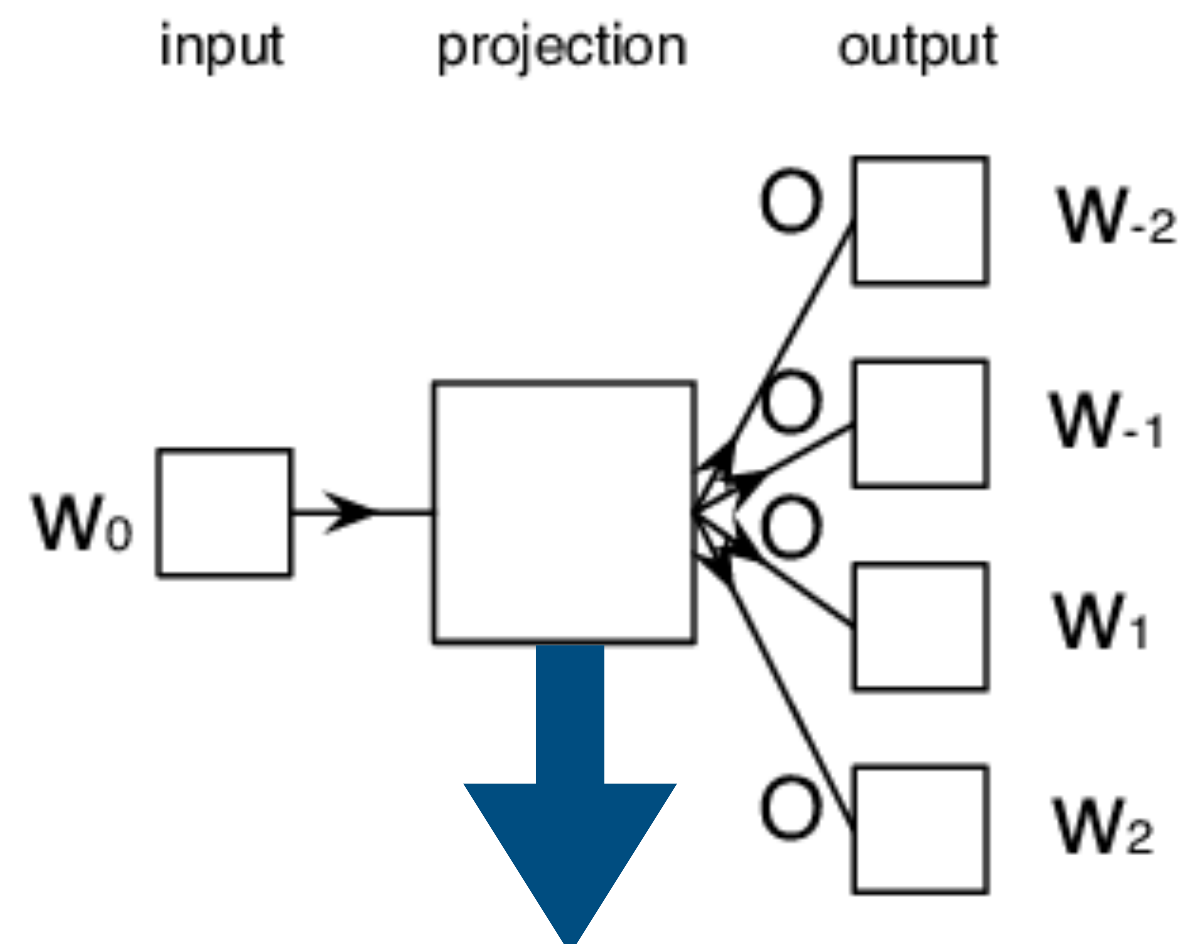
Predict the context words using the masked word

"Self-Supervised Learning"

Language Modeling Task

Skip-grams





	vec1	vec2	vec3	vec4	vec5	vec6
aardvark	-0.52	-0.49	-0.70	-0.02	0.11	0.37
Aaron	-0.40	-0.86	-0.40	0.32	0.01	0.40
able	-0.46	-0.85	-0.35	0.33	-0.13	0.57

Predict the context words using the masked word

"Self-Supervised Learning"

Negative Sampling

The cat [sat] on the mat

Masked Word	Context Word	In data
sat	the	1
sat	cat	1
sat	on	1
sat	the	1

Negative Sampling

The cat [sat] on the mat

Masked Word	Context Word	In data
sat	the	1
sat	cat	1
sat	on	1
sat	the	1
sat	horse	0
sat	flower	0
sat	running	0
sat	nice	0

Negative Sampling

The cat [sat] on the mat

random other words
not from context

Masked Word	Context Word	In data
sat	the	1
sat	cat	1
sat	on	1
sat	the	1
sat	horse	0
sat	flower	0
sat	running	0
sat	nice	0

Negative Sampling

The cat [sat] on the mat

random other words
not from context

Masked Word	Context Word	In data
sat	the	1
sat	cat	1
sat	on	1
sat	the	1
sat	horse	0
sat	flower	0
sat	running	0
sat	nice	0

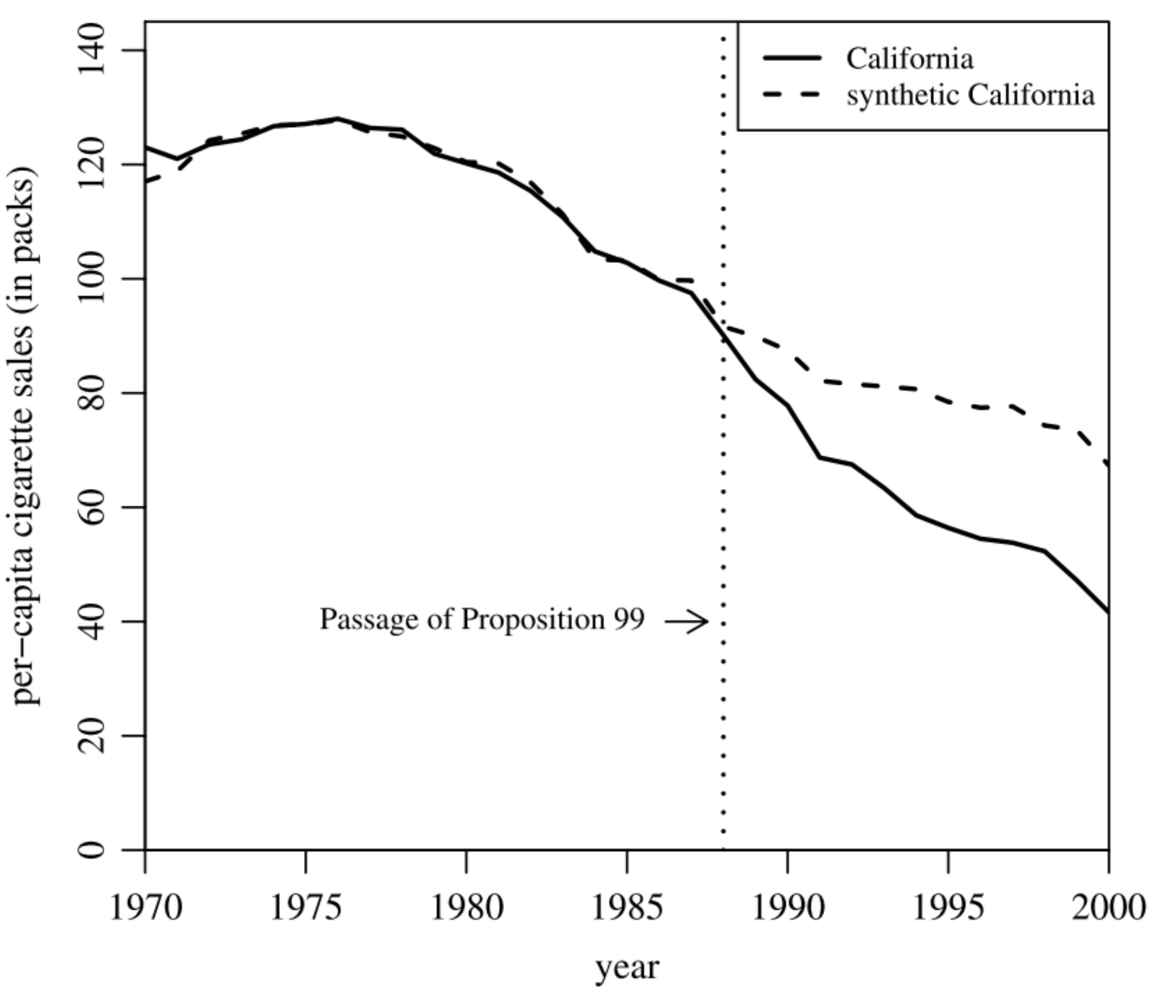
Similar to synthetic control
- generate comparison set

Negative Sampling

The cat [sat] on the mat

random other words
not from context

Masked Word	Context Word	In data
sat	the	1
sat	cat	1
sat	on	1
sat	the	1
sat	horse	0
sat	flower	0
sat	running	0
sat	nice	0

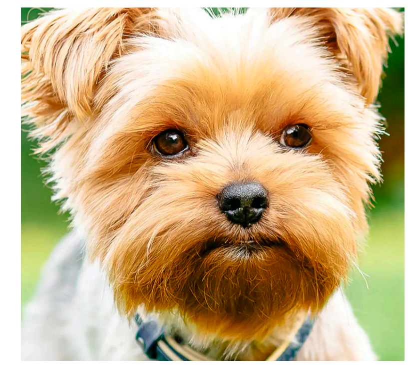


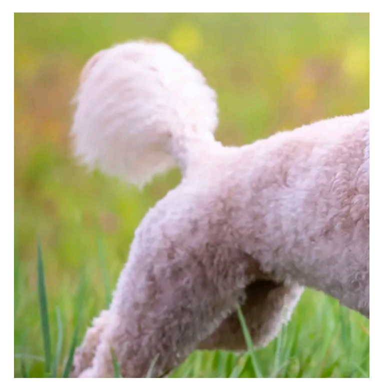
Similar to synthetic control
- generate comparison set

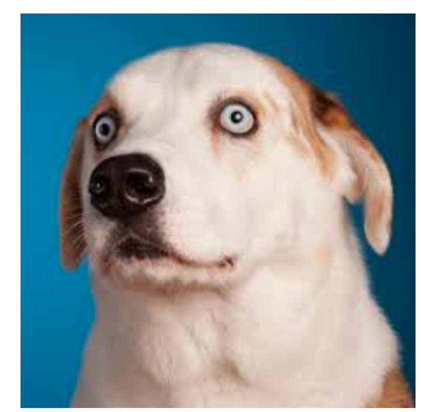
Self-Supervised Image Learning

Chen et al., 2020

Self-Supervised Image Learning







Chen et al., 2020

Self-Supervised Image Learning

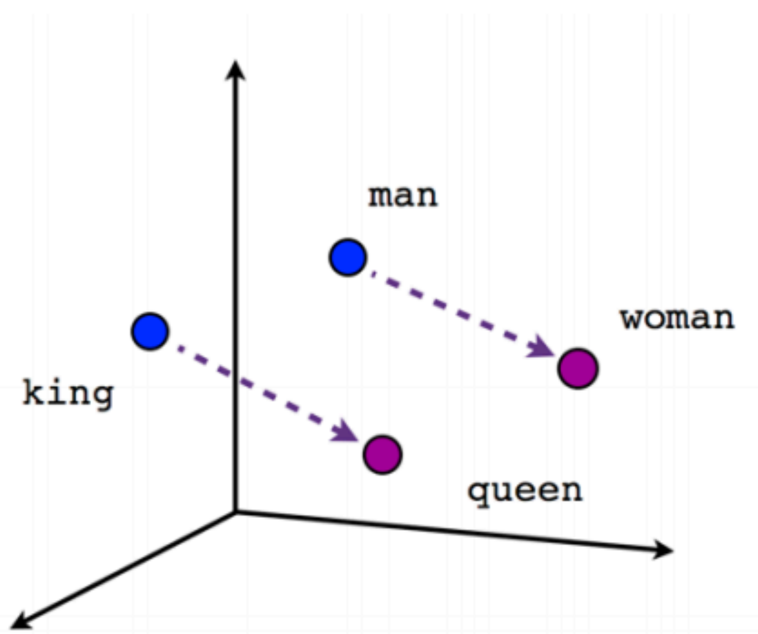
(a) Original (b) Crop and resize (c) Crop, resize (and flip) (d) Color distort. (drop) (e) Color distort. (jitter)

(f) Rotate $\{90°, 180°, 270°\}$ (g) Cutout (h) Gaussian noise (i) Gaussian blur (j) Sobel filtering

Chen et al., 2020

Embedding Spaces

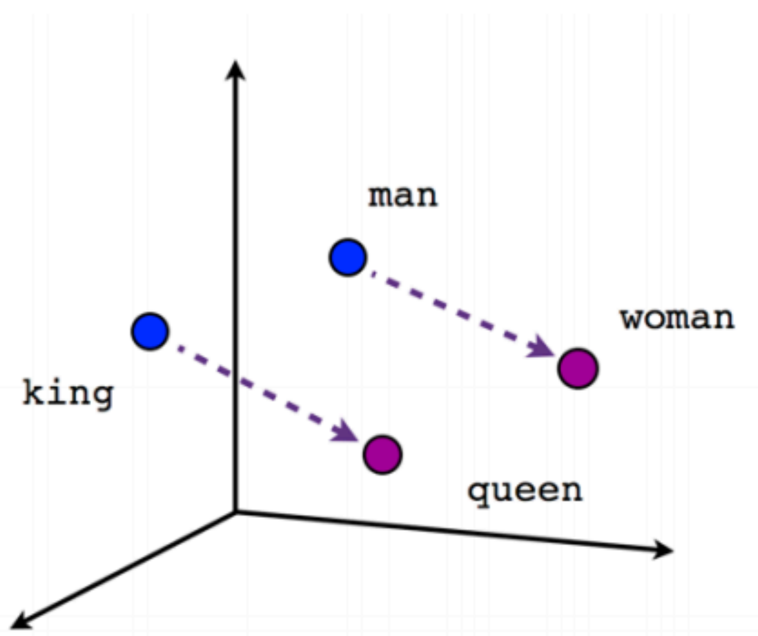
What does meaning look like?



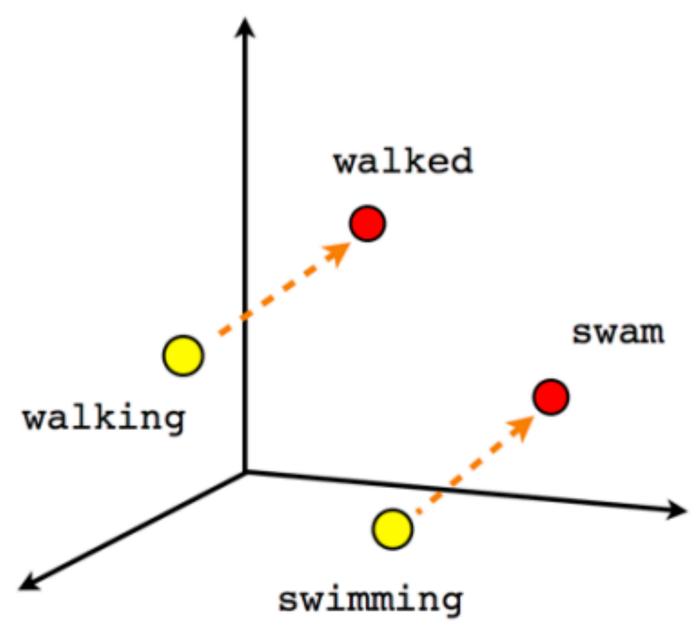
Male-Female

Embedding Spaces

What does meaning look like?



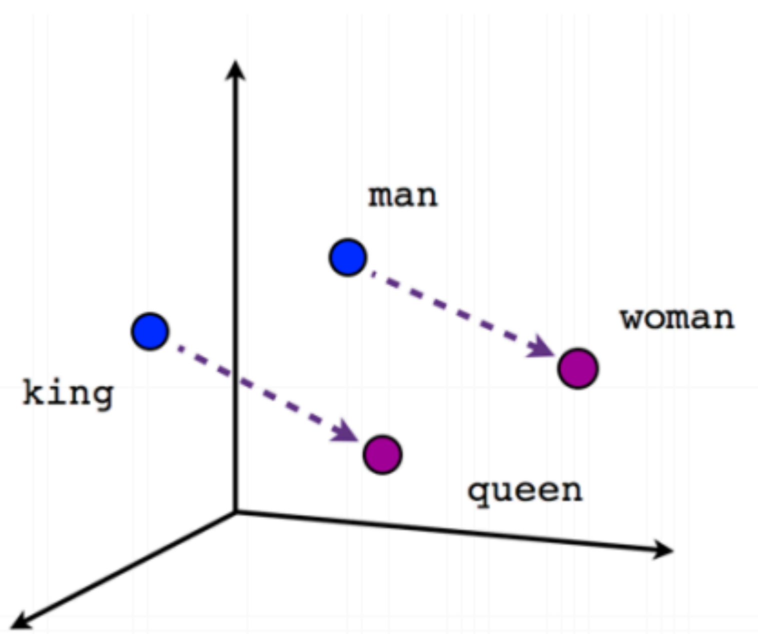
Male-Female



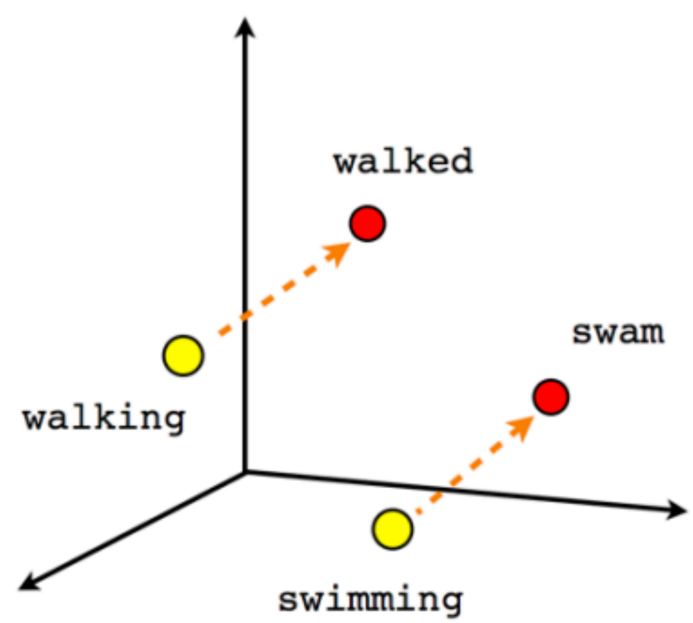
Verb tense

Embedding Spaces

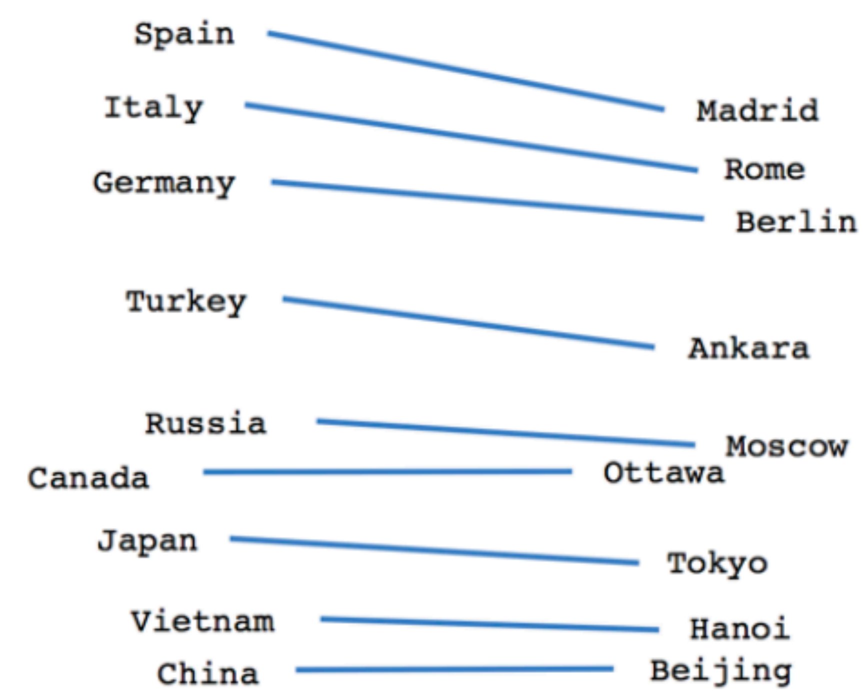
What does meaning look like?



Male-Female



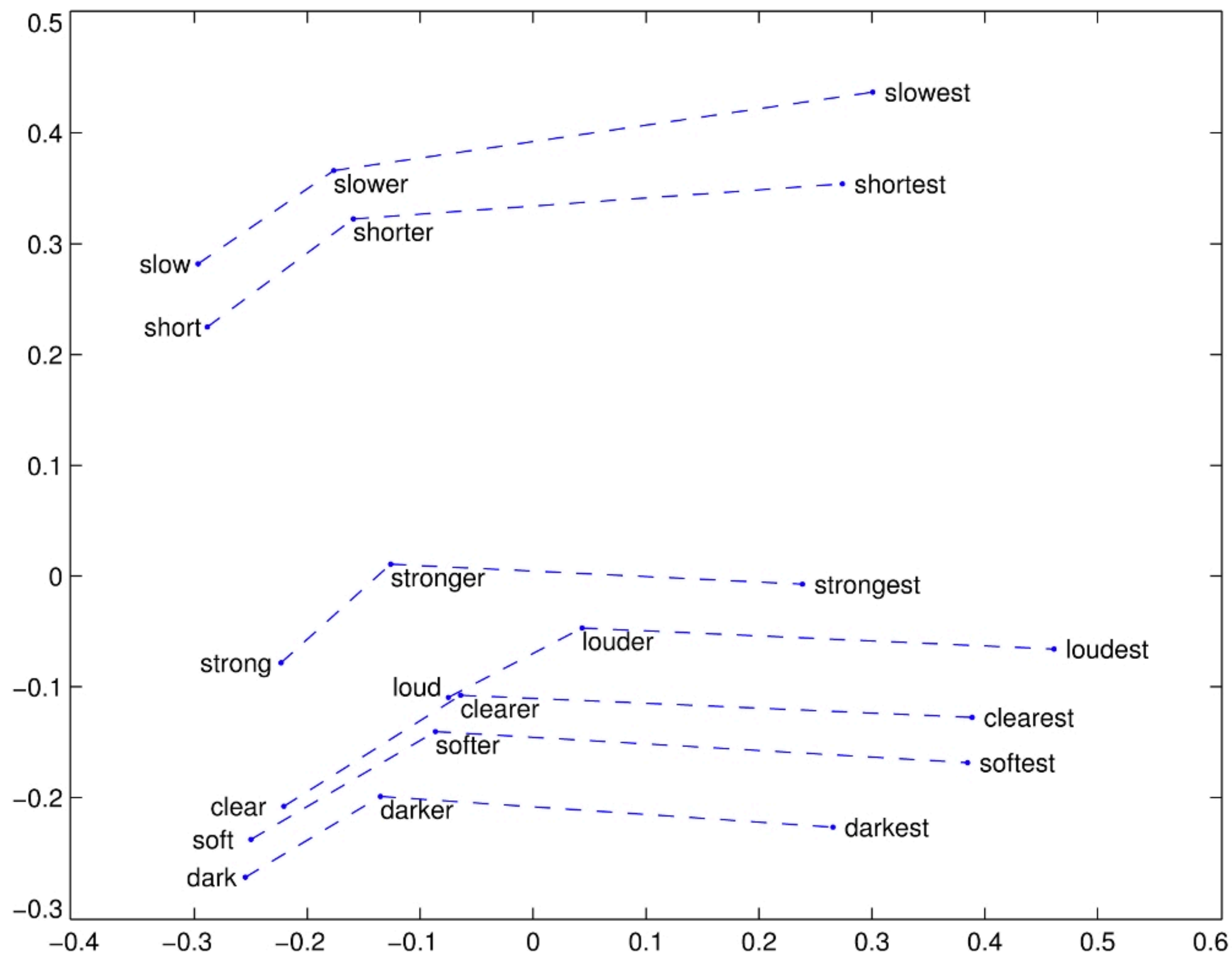
Verb tense



Country-Capital

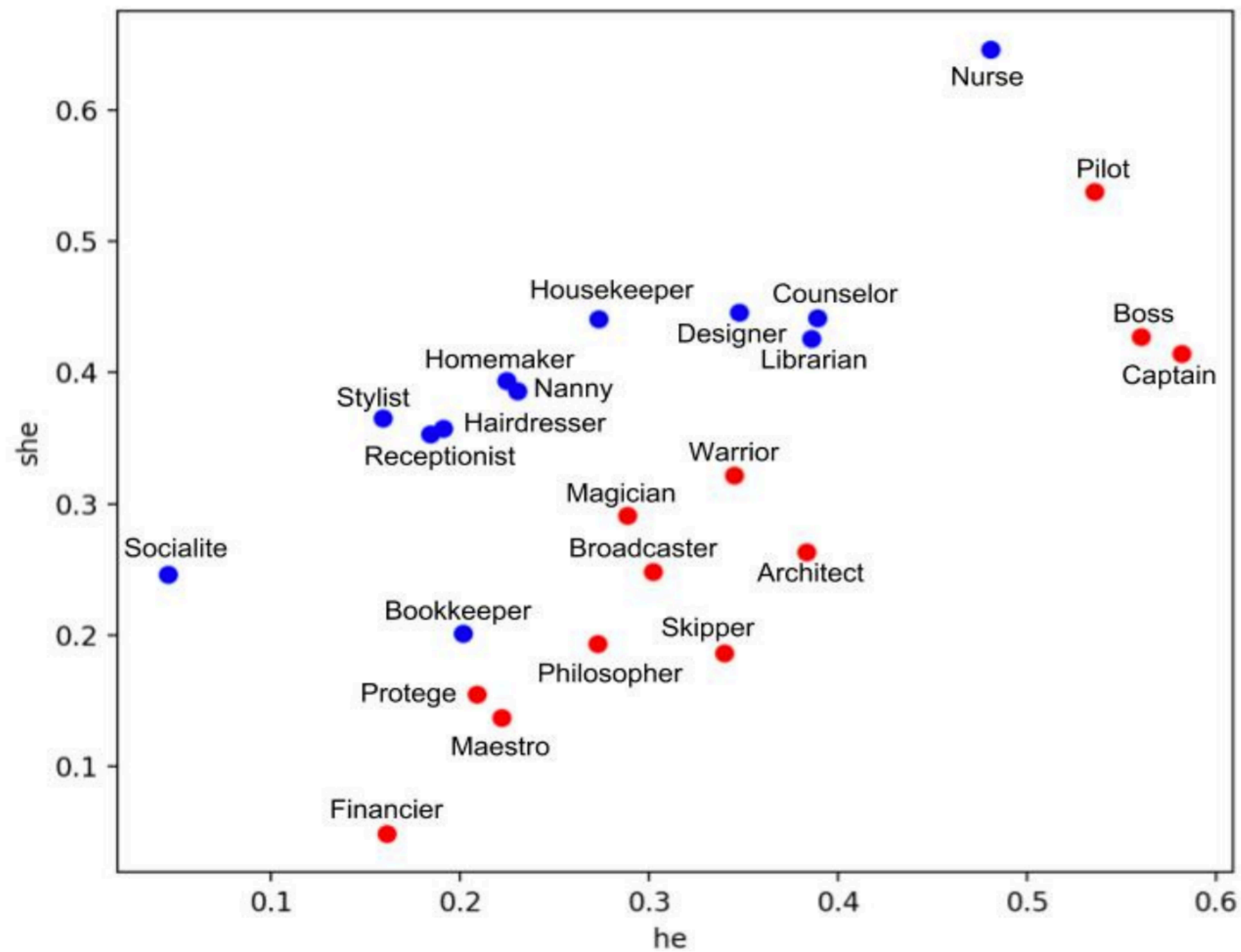
Embedding Spaces

What does meaning look like?



Embedding Spaces

What does meaning look like?

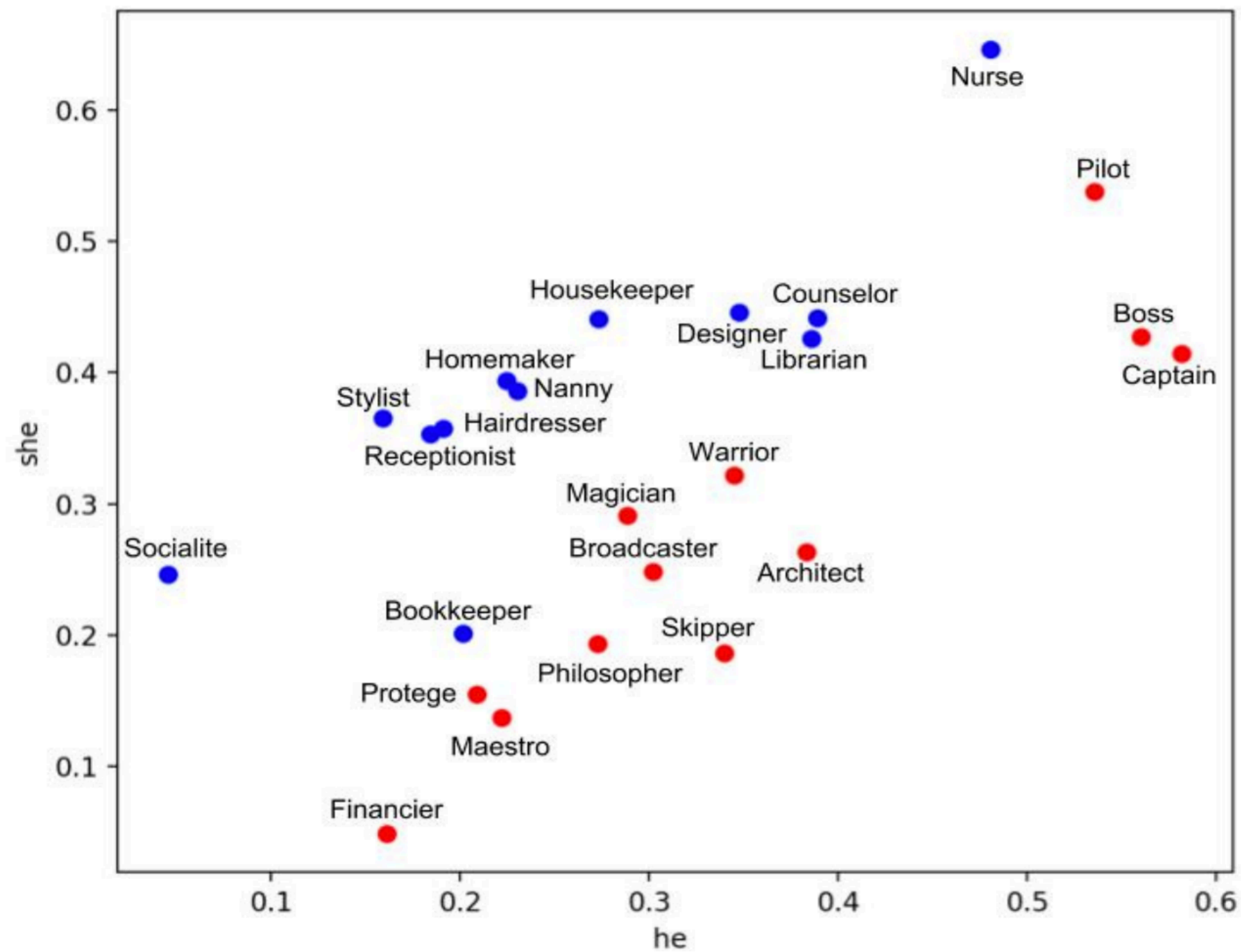


Caliskan, Bryson & Narayanan, 2017

Embedding Spaces

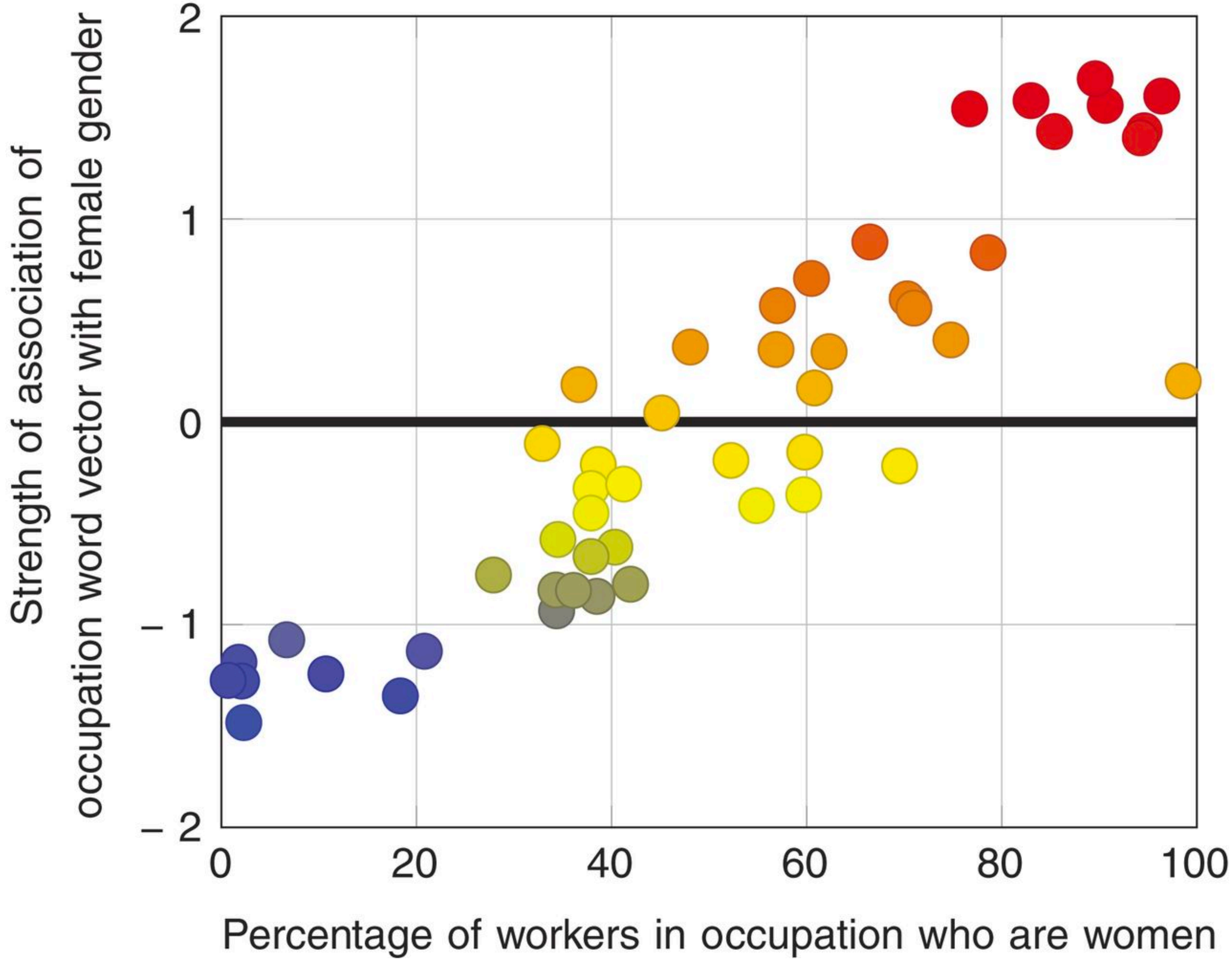
What does meaning look like?

Where is the bias?



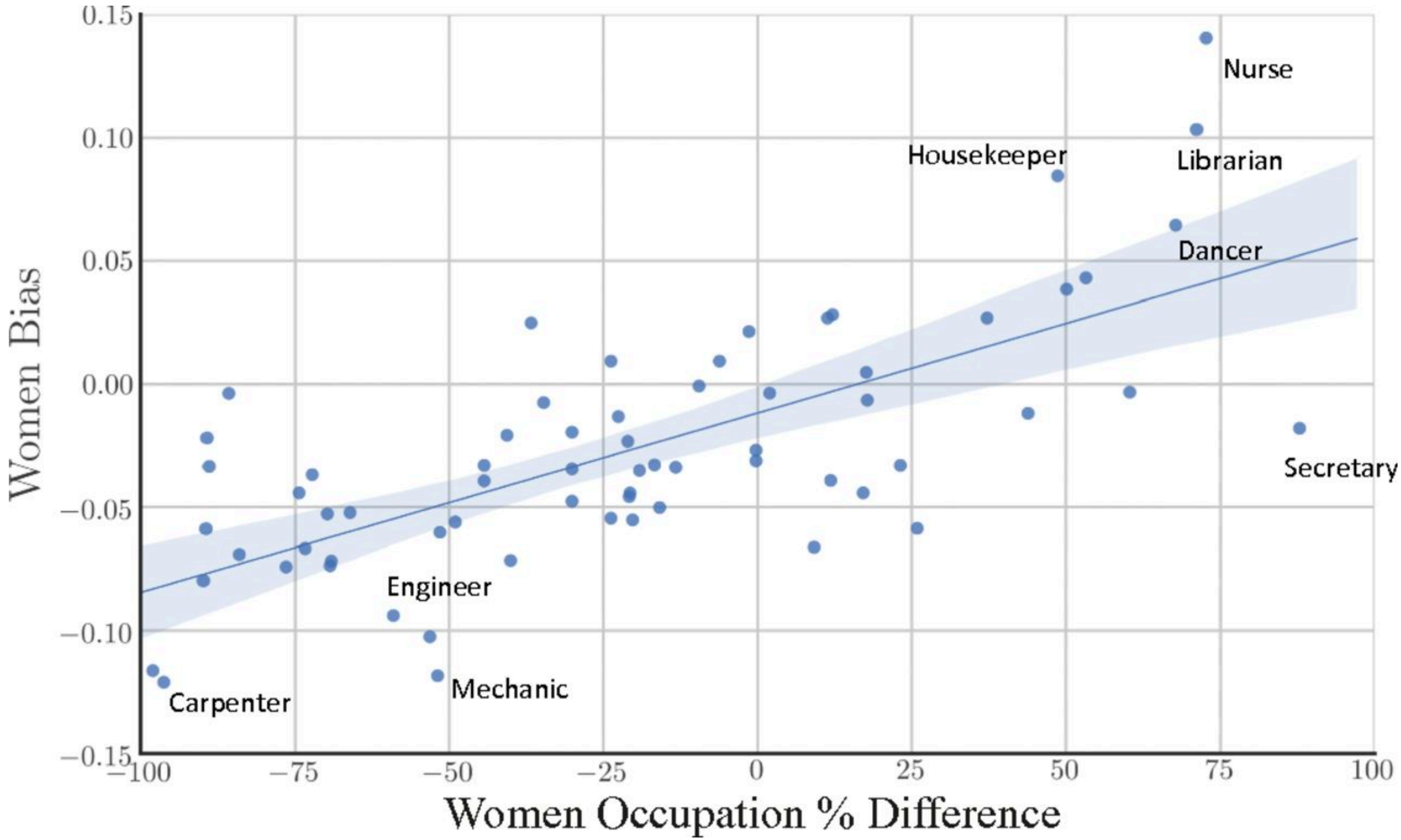
Caliskan, Bryson & Narayanan, 2017

Applications of Word Similarity



Caliskan, Bryson & Narayanan, 2017

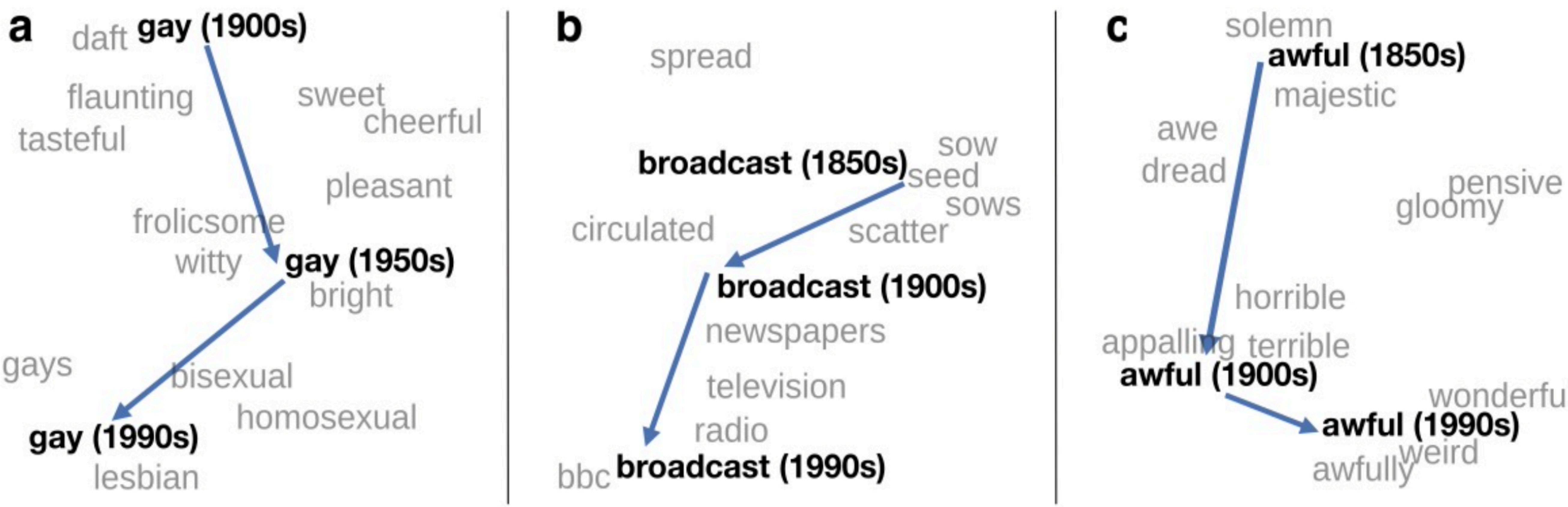
Applications of Word Similarity



Garg et al., 2018

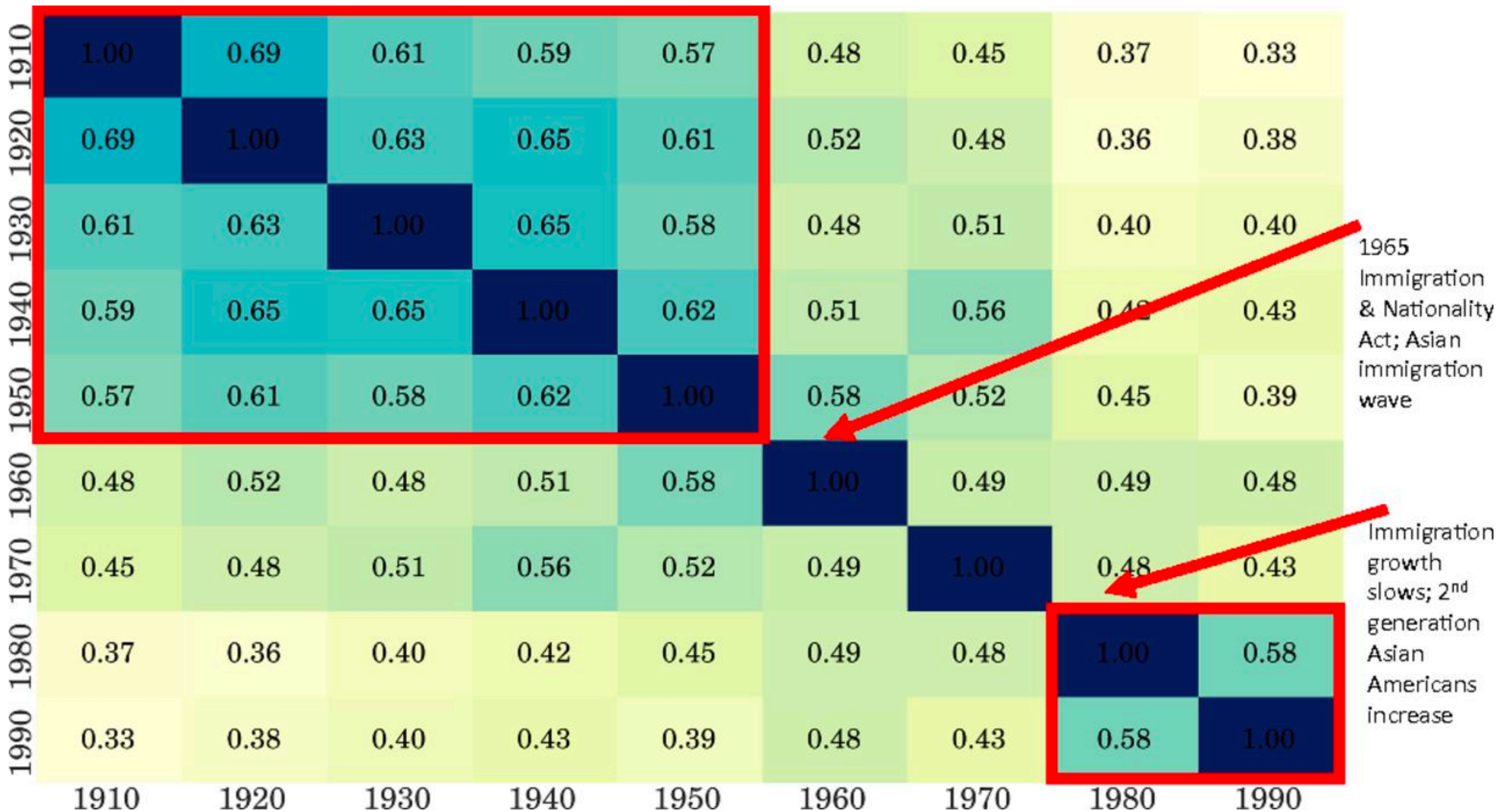
Applications of Word Similarity

~30 million books, 1850-1990, Google Books data



Hamilton et al., 2016

Applications of Word Similarity



	1910	1920	1930	1940	1950	1960	1970	1980	1990
1910	1.00	0.69	0.61	0.59	0.57	0.48	0.45	0.37	0.33
1920	0.69	1.00	0.63	0.65	0.61	0.52	0.48	0.36	0.38
1930	0.61	0.63	1.00	0.65	0.58	0.48	0.51	0.40	0.40
1940	0.59	0.65	0.65	1.00	0.62	0.51	0.56	0.42	0.43
1950	0.57	0.61	0.58	0.62	1.00	0.58	0.52	0.45	0.39
1960	0.48	0.52	0.48	0.51	0.58	1.00	0.49	0.49	0.48
1970	0.45	0.48	0.51	0.56	0.52	0.49	1.00	0.48	0.43
1980	0.37	0.36	0.40	0.42	0.45	0.49	0.48	1.00	0.58
1990	0.33	0.38	0.40	0.43	0.39	0.48	0.43	0.58	1.00

Garg et al., 2018

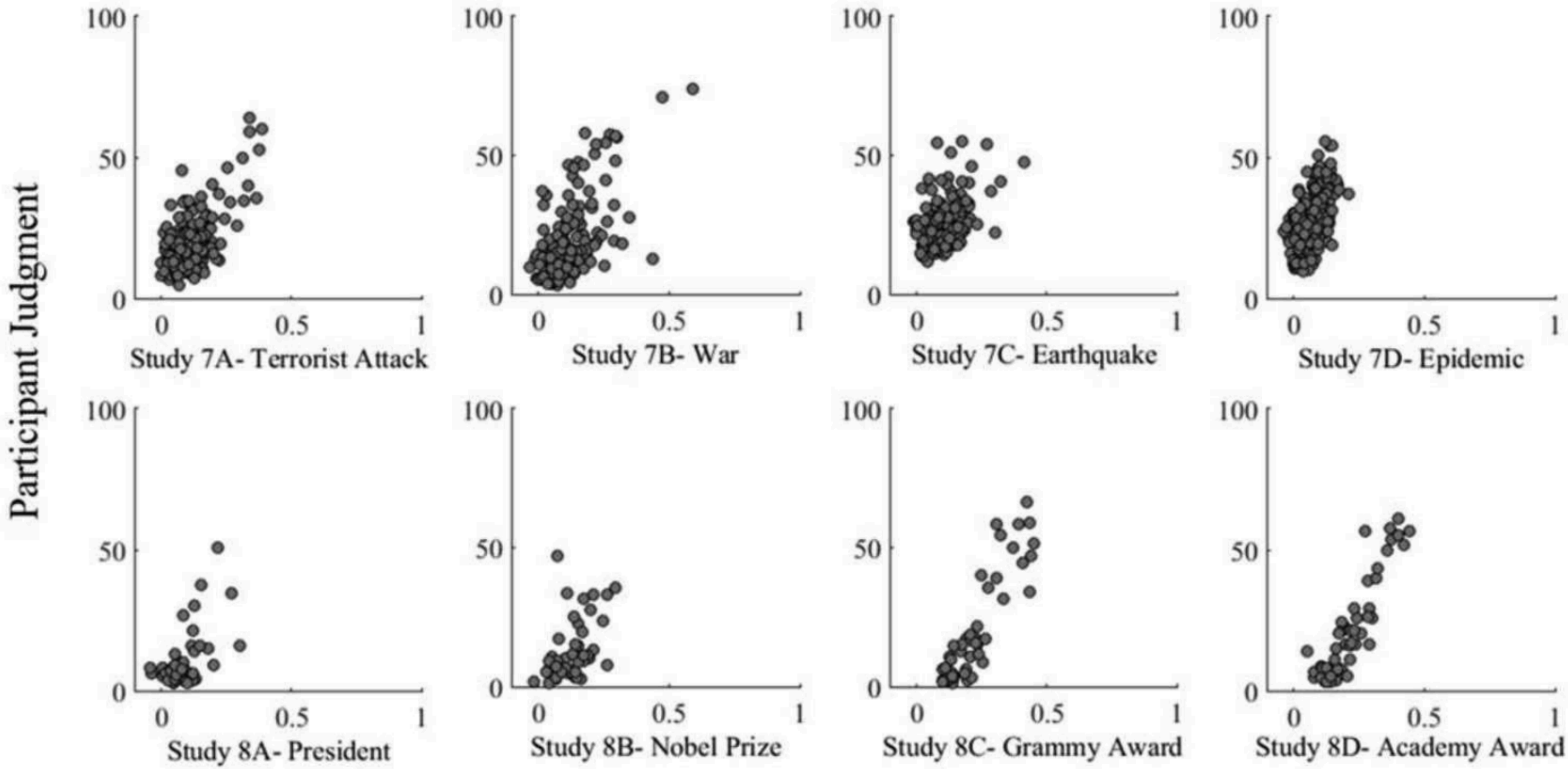
Applications of Word Similarity

Table 3. Top Asian (vs. White) adjectives in 1910, 1950, and 1990 by relative norm difference in the COHA embedding

1910	1950	1990
Irresponsible	Disorganized	Inhibited
Envious	Outrageous	Passive
Barbaric	Pompous	Dissolute
Aggressive	Unstable	Haughty
Transparent	Effeminate	Complacent
Monstrous	Unprincipled	Forceful
Hateful	Venomous	Fixed
Cruel	Disobedient	Active
Greedy	Predatory	Sensitive
Bizarre	Boisterous	Hearty

Garg et al., 2018

Applications of Word Similarity



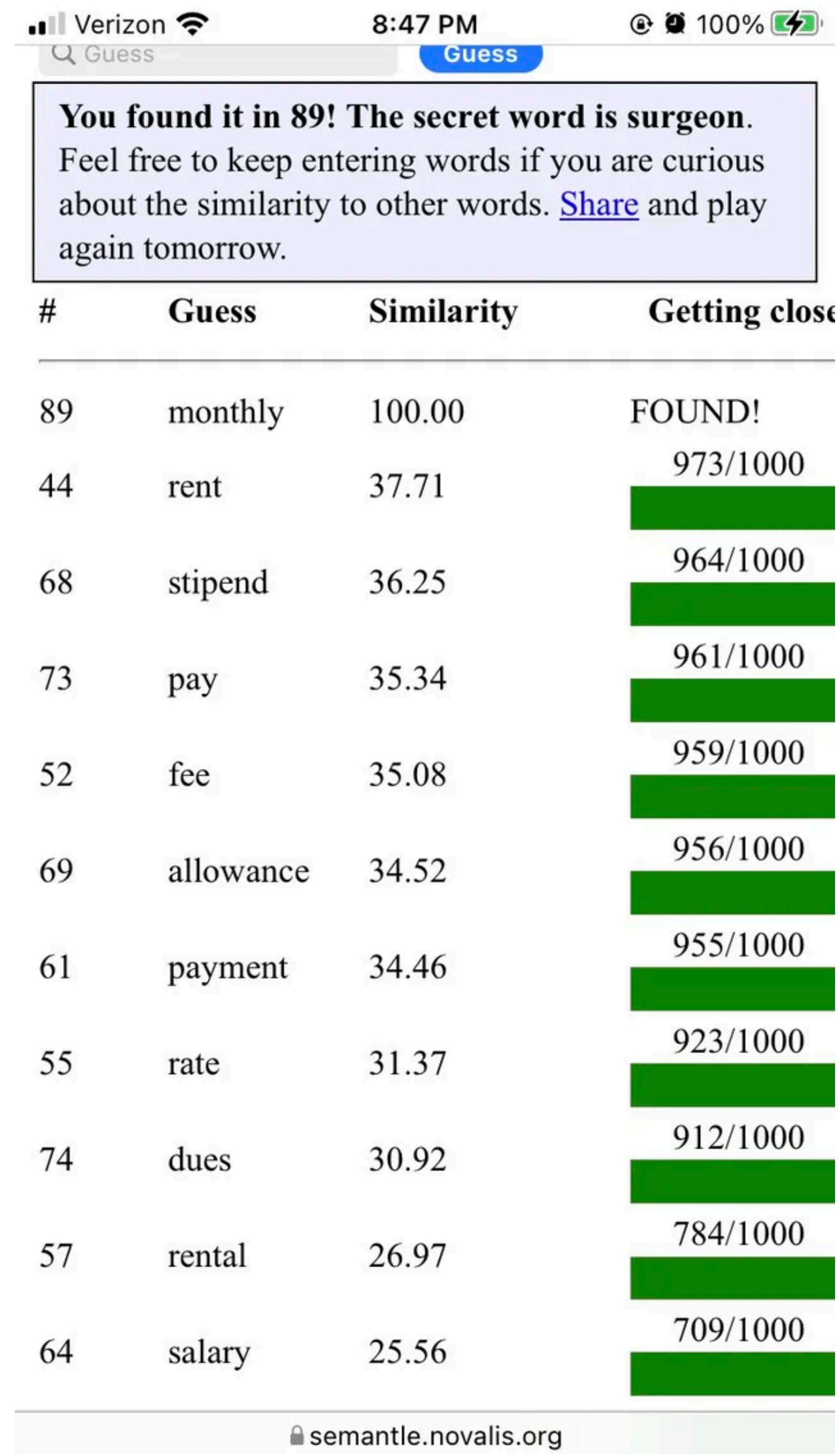
Bhatia, 2017

Applications of Word Similarity

https://semantle.novalis.org/

Guess the hidden word with

similarity of previous guesses


Verizon 8:47 PM 100%

Guess

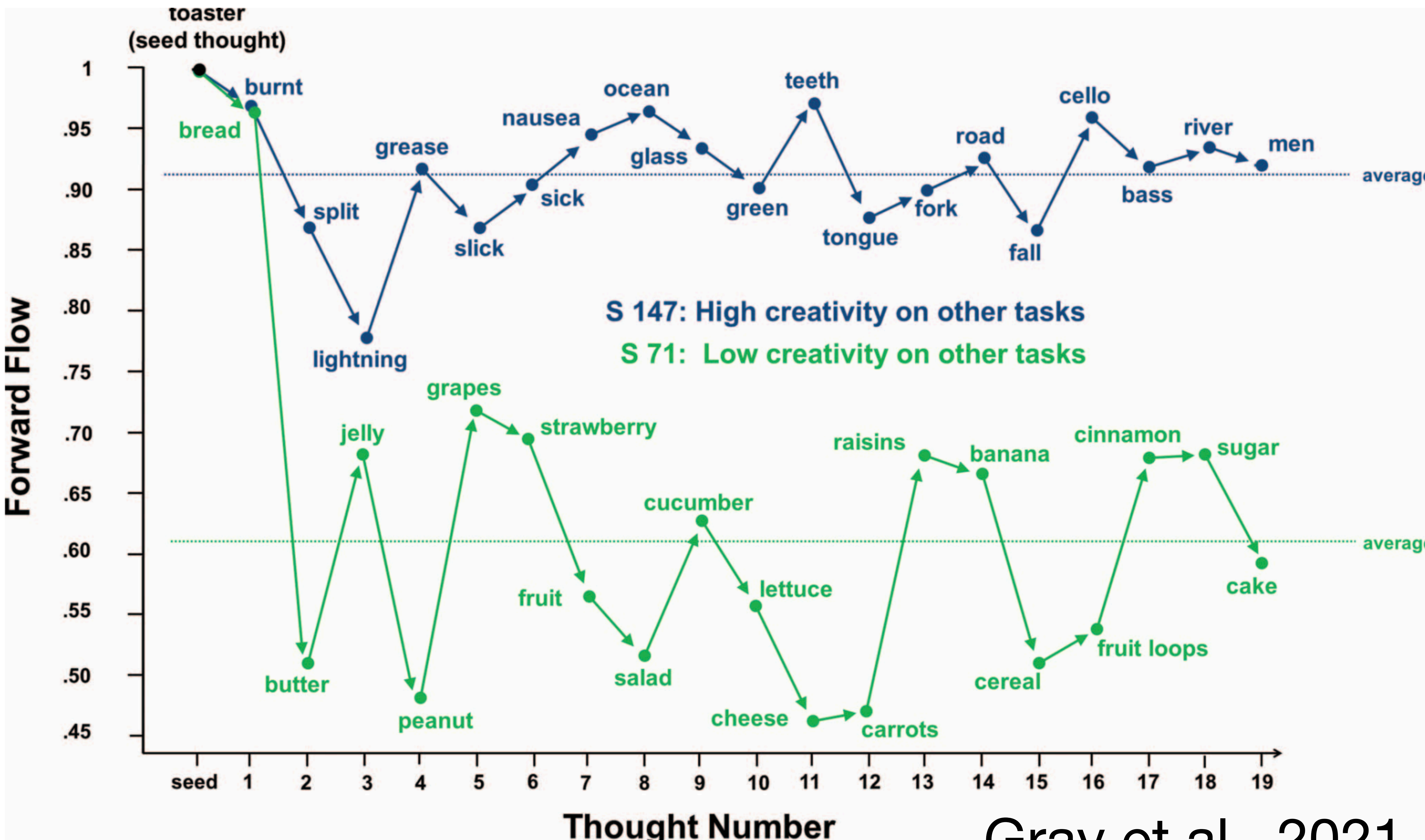
Guess

You found it in 89! The secret word is surgeon. Feel free to keep entering words if you are curious about the similarity to other words. Share and play again tomorrow.

#	Guess	Similarity	Getting close
89	monthly	100.00	FOUND!
44	rent	37.71	973/1000
68	stipend	36.25	964/1000
73	pay	35.34	961/1000
52	fee	35.08	959/1000
69	allowance	34.52	956/1000
61	payment	34.46	955/1000
55	rate	31.37	923/1000
74	dues	30.92	912/1000
57	rental	26.97	784/1000
64	salary	25.56	709/1000

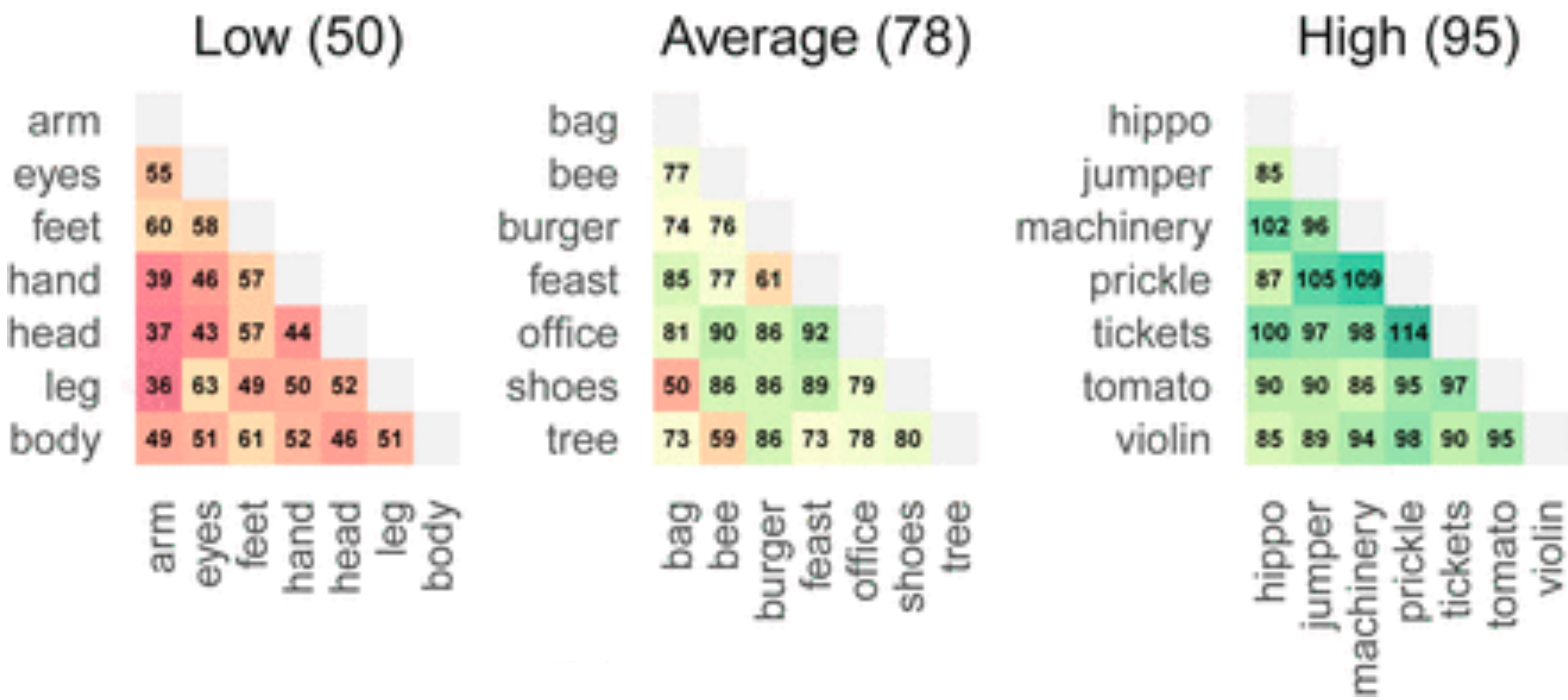
semantle.novalis.org

Applications of Word Similarity



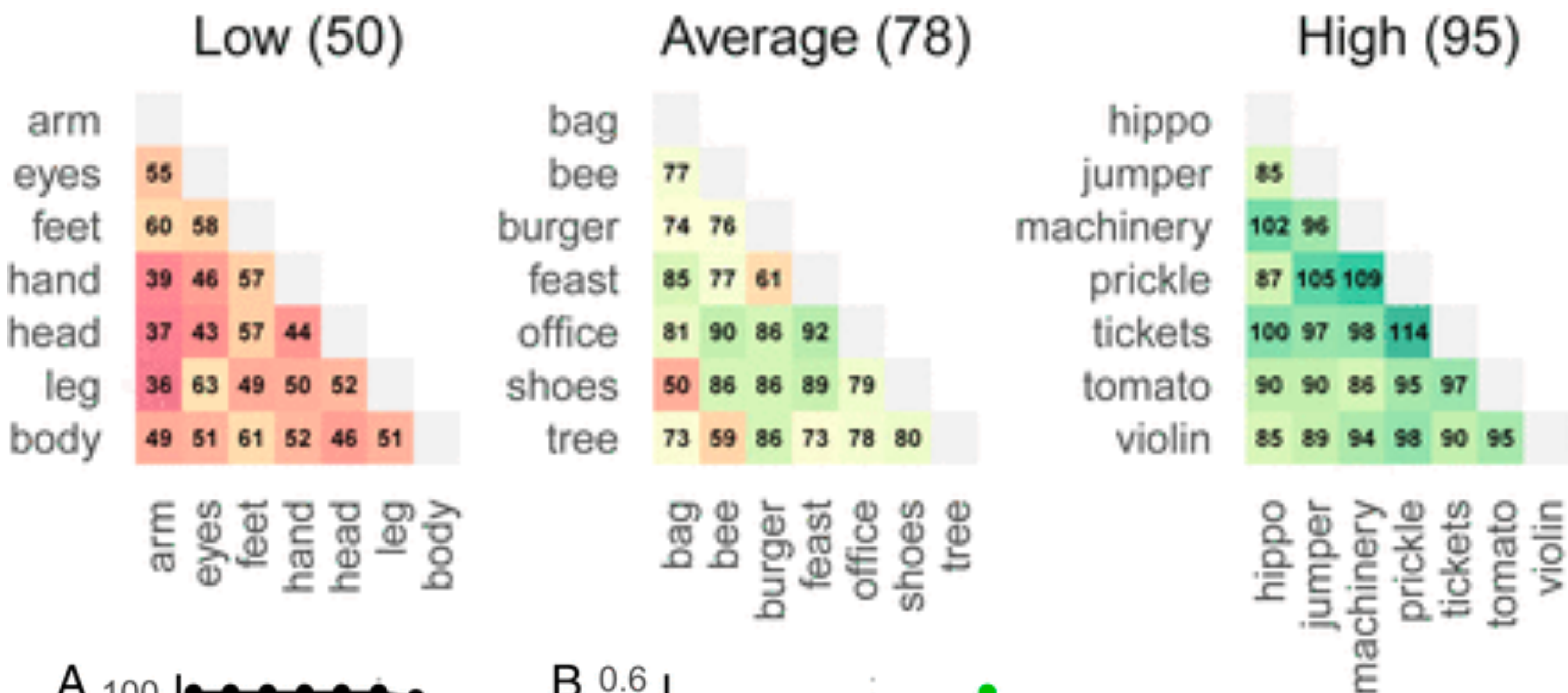
Gray et al., 2021

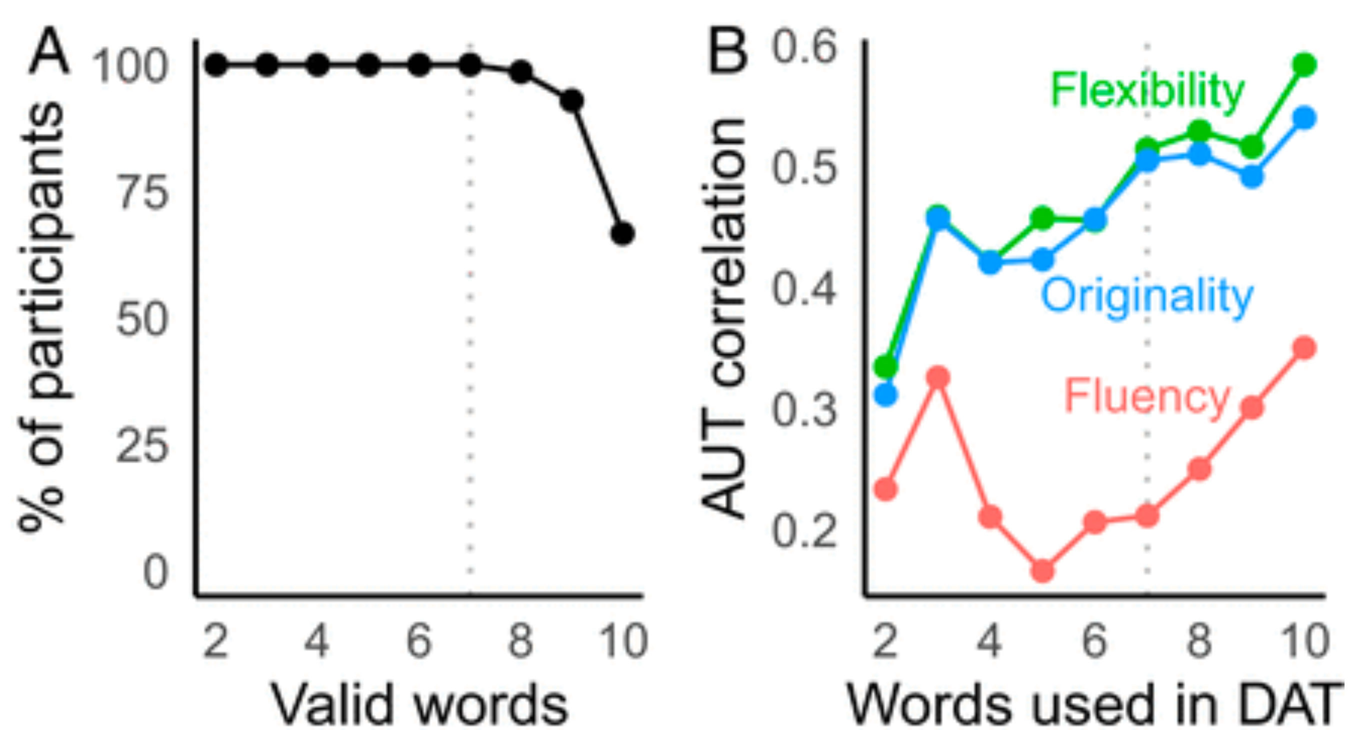
Applications of Word Similarity



Gray et al., 2021

Applications of Word Similarity



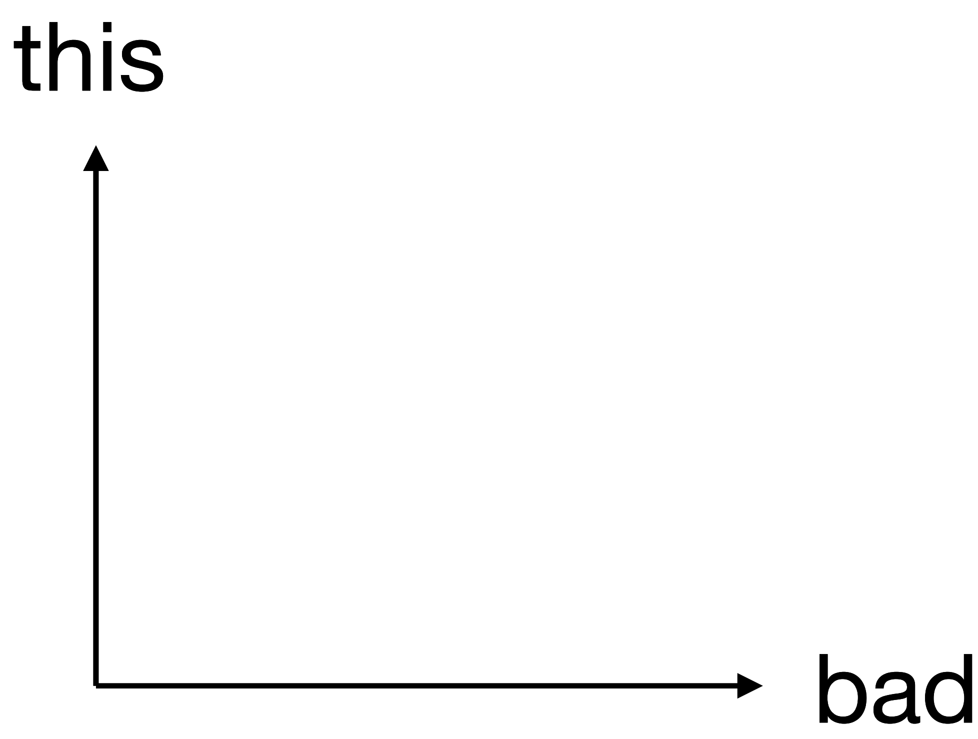


Gray et al., 2021

Vector Space Model of Text

"This is bad. So bad."
"This is good"
"That is bad, very bad"

this	is	bad	good	that	very
1	1	2	0	0	0
1	1	0	1	0	0
0	1	2	0	1	1



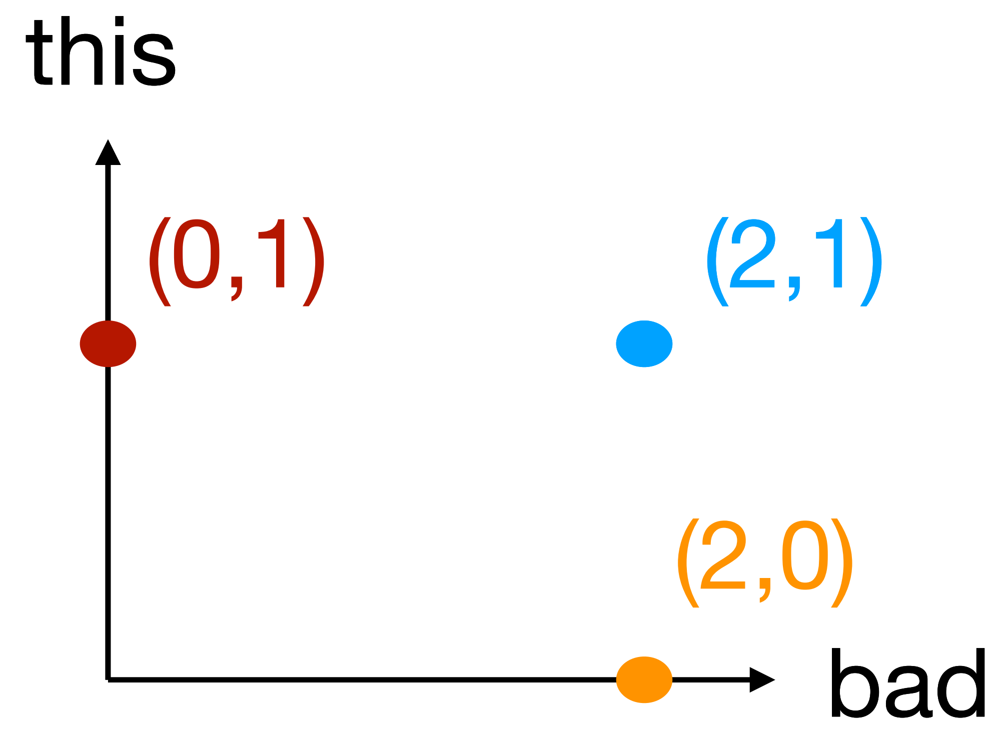
Vector Space Model of Text

"This is bad. So bad."

"This is good"

"That is bad, very bad"

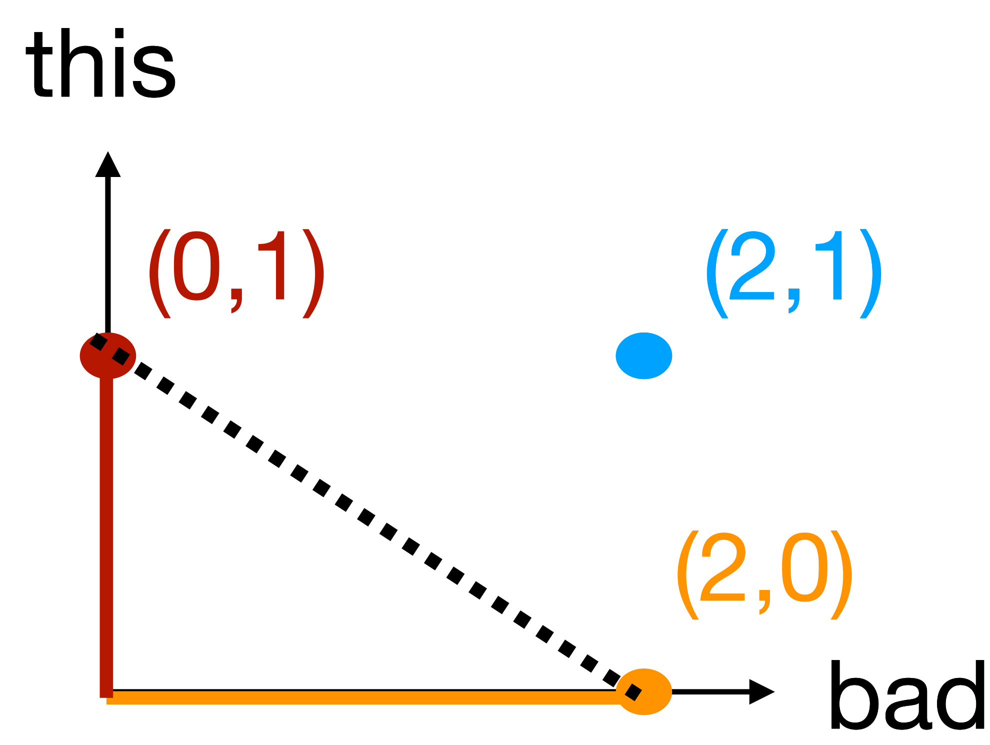
this	is	bad	good	that	very
1	1	2	0	0	0
1	1	0	1	0	0
0	1	2	0	1	1



Vector Space Model of Text

"This is bad. So bad."
"This is good"
"That is bad, very bad"

this	is	bad	good	that	very
1	1	2	0	0	0
1	1	0	1	0	0
0	1	2	0	1	1



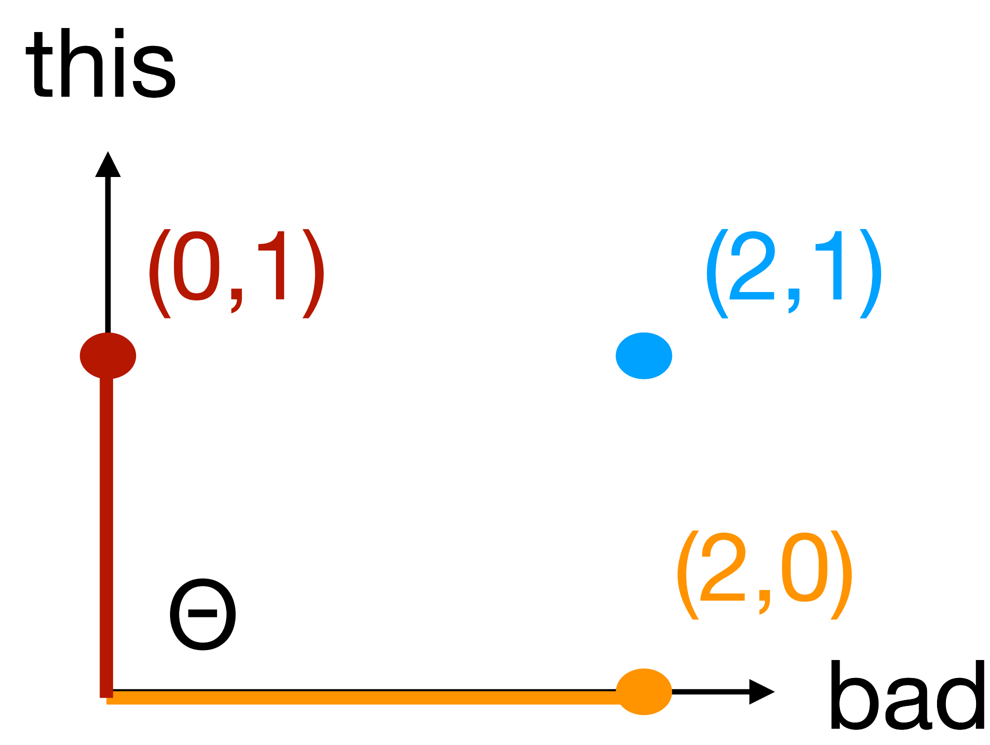
How far is document 2 from document 3?

Euclidian Distance $(d_2,d_3) = \mathrm{sqrt}((0-2)^2+(1-0)^2)$

Vector Space Model of Text

"This is bad. So bad."
"This is good"
"That is bad, very bad"

this	is	bad	good	that	very
1	1	2	0	0	0
1	1	0	1	0	0
0	1	2	0	1	1



How far is document 2 from document 3?

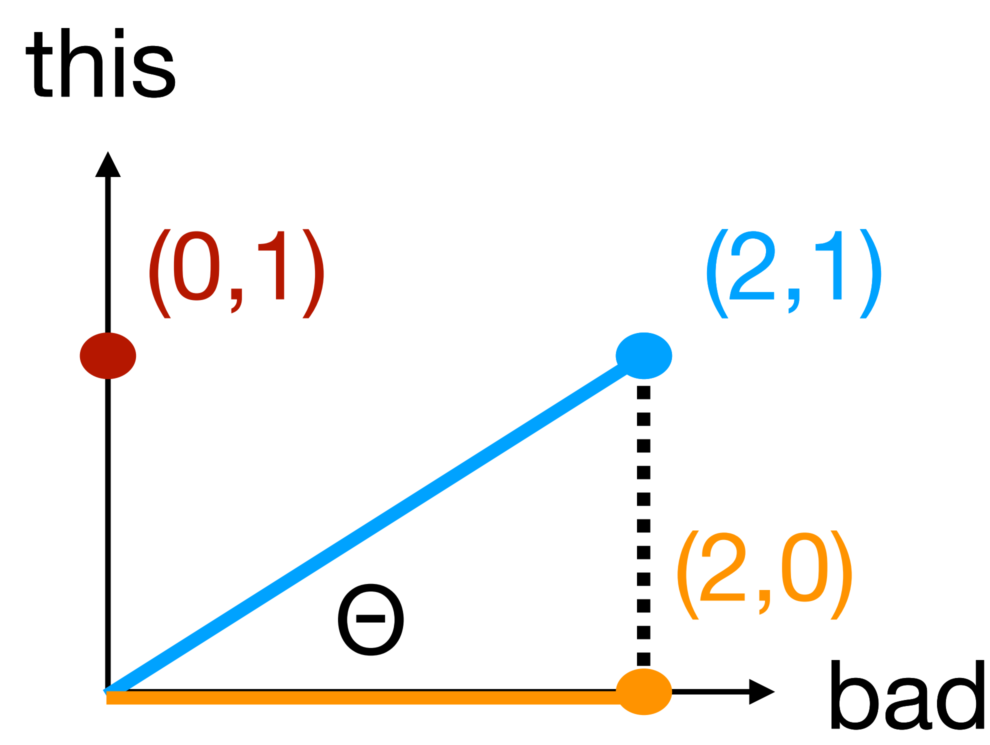
Euclidian Distance (d_2,d_3) = $\mathrm{sqrt}((0-2)^2+(1-0)^2)$

Cosine Distance (d_2,d_3) $=\cos(\Theta) = (d_2 \cdot d_3)/||d_2||*||d_3||$

Vector Space Model of Text

"This is bad. So bad."
"This is good"
"That is bad, very bad"

this	is	bad	good	that	very
1	1	2	0	0	0
1	1	0	1	0	0
0	1	2	0	1	1



How far is document 1 from document 3?

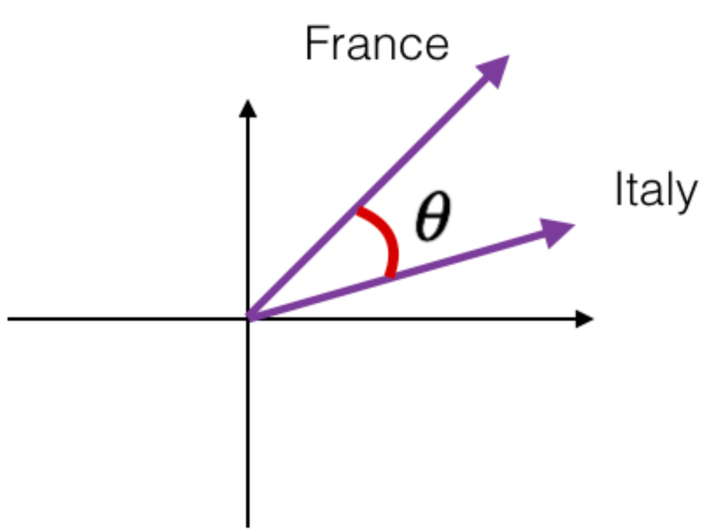
Euclidian Distance $(d_1,d_3) = \text{sqrt}((2-2)^2+(1-0)^2)$

Cosine Distance $(d_1,d_3) = \cos(\Theta) = (d_1 \cdot d_3)/||d_1||*||d_3||$

Vector Space Model of Text

Cosine similarity

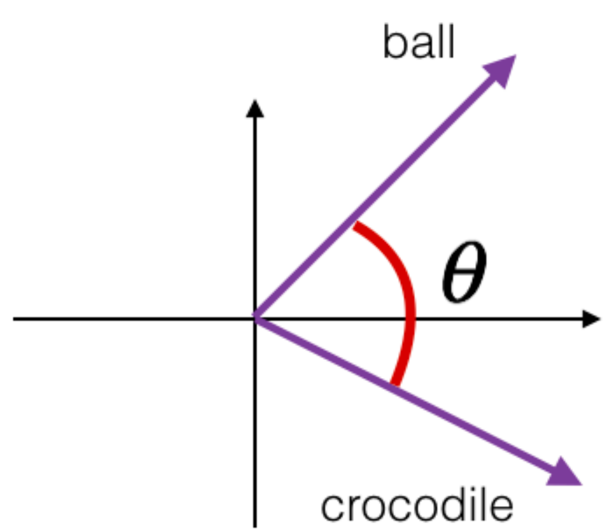
Indifferent to document length



France and Italy are quite similar

θ is close to 0°

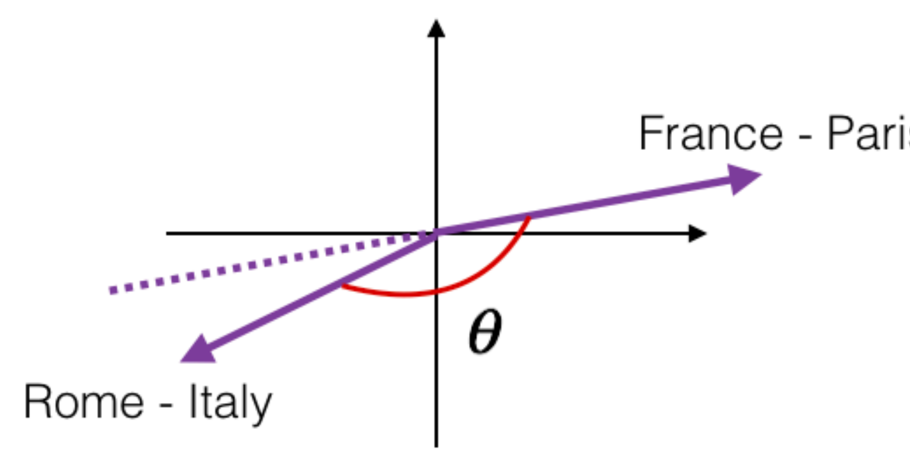
$\cos(\theta) \approx 1$



ball and crocodile are not similar

θ is close to 90°

$\cos(\theta) \approx 0$



the two vectors are similar but opposite
the first one encodes (city - country)
while the second one encodes (country - city)

θ is close to 180°

$\cos(\theta) \approx -1$

Embedding Models

Calculating similarity

Consider three sentences

“I am very sad”, “I am very happy”, “I am thrilled”

Which of the first two is more similar to the third?

Embedding Models

Calculating similarity

Consider three sentences

"I am very sad", "I am very happy", "I am thrilled"

Which of the first two is more similar to the third?

According to
bag of words

Embedding Models

Calculating similarity

Consider three sentences
“I am very sad”, “I am very happy”, “I am thrilled”

Which of the first two is more similar to the third?

According to bag of words

	i	am	veri	sad	happi	thrill
text1	1	1	1	1	0	0
text2	1	1	1	0	1	0
text3	1	1	0	0	0	1

Embedding Models

Calculating similarity

Consider three sentences

“I am very sad”, “I am very happy”, “I am thrilled”

Which of the first two is more similar to the third?

According to bag of words

	i	am	veri	sad	happi	thrill
text1	1	1	1	1	0	0
text2	1	1	1	0	1	0
text3	1	1	0	0	0	1

$\cos(t1, t3) = .57$

$\cos(t2, t3) = .57$

Embedding Models

Calculating similarity

Consider three sentences

“I am very sad”, “I am very happy”, “I am thrilled”

Which of the first two is more similar to the third?

According to bag of words

	i	am	veri	sad	happi	thrill
text1	1	1	1	1	0	0
text2	1	1	1	0	1	0
text3	1	1	0	0	0	1

cos(t1, t3) = .57

cos(t2, t3) = .57

According to word2vec

Embedding Models

Calculating similarity

Consider three sentences
"I am very sad", "I am very happy", "I am thrilled"

Which of the first two is more similar to the third?

According to bag of words

	i	am	veri	sad	happi	thrill
text1	1	1	1	1	0	0
text2	1	1	1	0	1	0
text3	1	1	0	0	0	1

cos(t1, t3) = .57
cos(t2, t3) = .57

According to word2vec

	vec1	vec2	vec3	vec4	vec5	vec6
text1	-0.52	-0.49	-0.70	-0.02	0.11	0.37
text2	-0.40	-0.86	-0.40	0.32	0.01	0.40
text3	-0.46	-0.85	-0.35	0.33	-0.13	0.57

Embedding Models

Calculating similarity

Consider three sentences

"I am very sad", "I am very happy", "I am thrilled"

Which of the first two is more similar to the third?

According to bag of words

	i	am	veri	sad	happi	thrill
text1	1	1	1	1	0	0
text2	1	1	1	0	1	0
text3	1	1	0	0	0	1

cos(t1, t3) = .57
cos(t2, t3) = .57

According to word2vec

	vec1	vec2	vec3	vec4	vec5	vec6
text1	-0.52	-0.49	-0.70	-0.02	0.11	0.37
text2	-0.40	-0.86	-0.40	0.32	0.01	0.40
text3	-0.46	-0.85	-0.35	0.33	-0.13	0.57

cos(t1, t3) = .81
cos(t2, t3) = .87

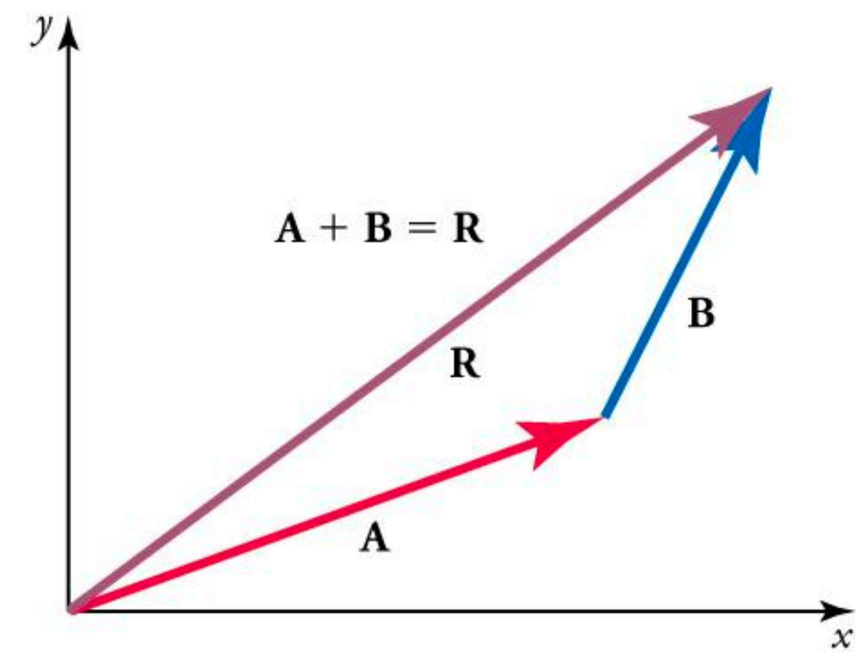
Sentence Embeddings

Simple

Sentence Embeddings

Simple

Add up the vectors of individual words

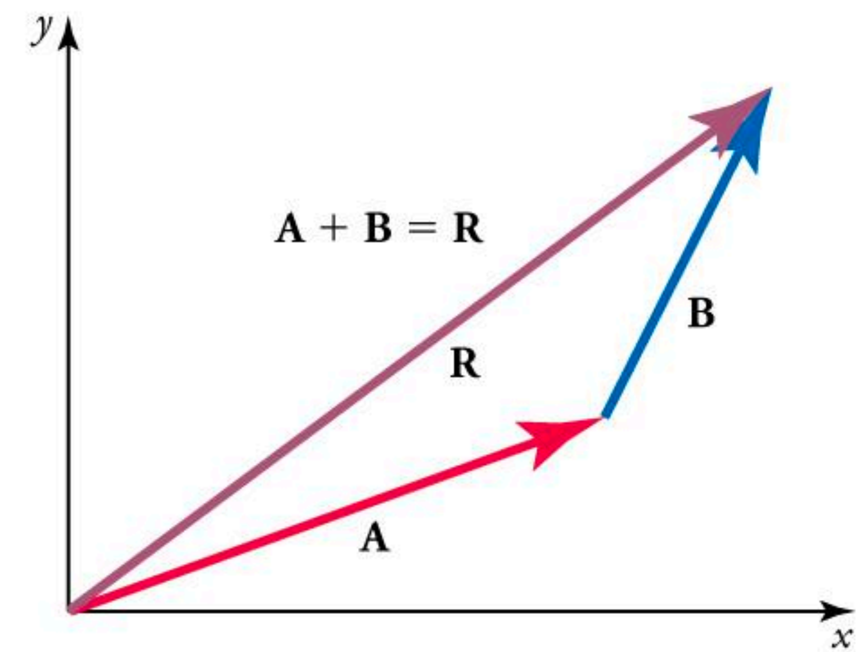


(Arora et al., 2017)

Sentence Embeddings

Simple

Add up the vectors of individual words



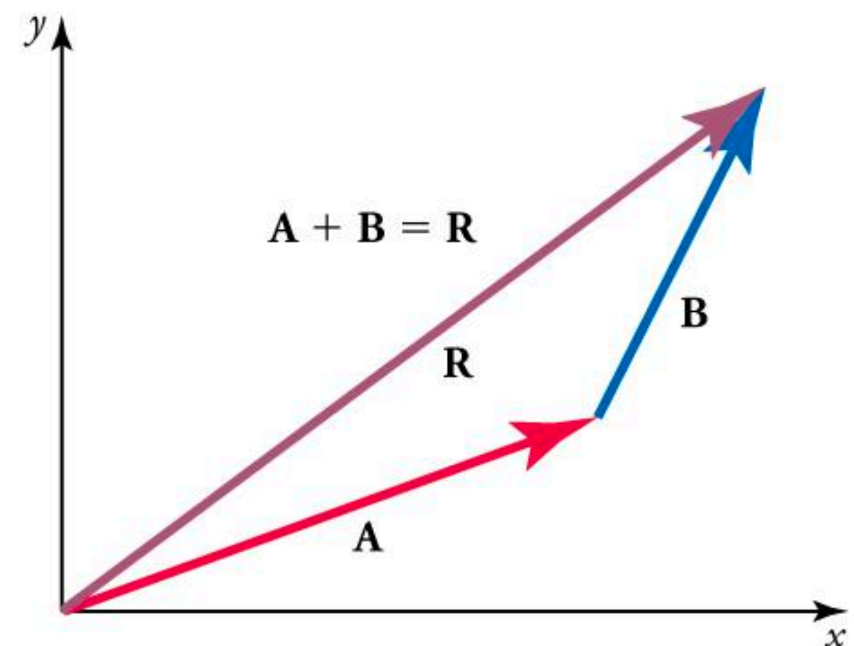
How to combine?
-Unweighted average

(Arora et al., 2017)

Sentence Embeddings

Simple

Add up the vectors of individual words



How to combine?
-Unweighted average

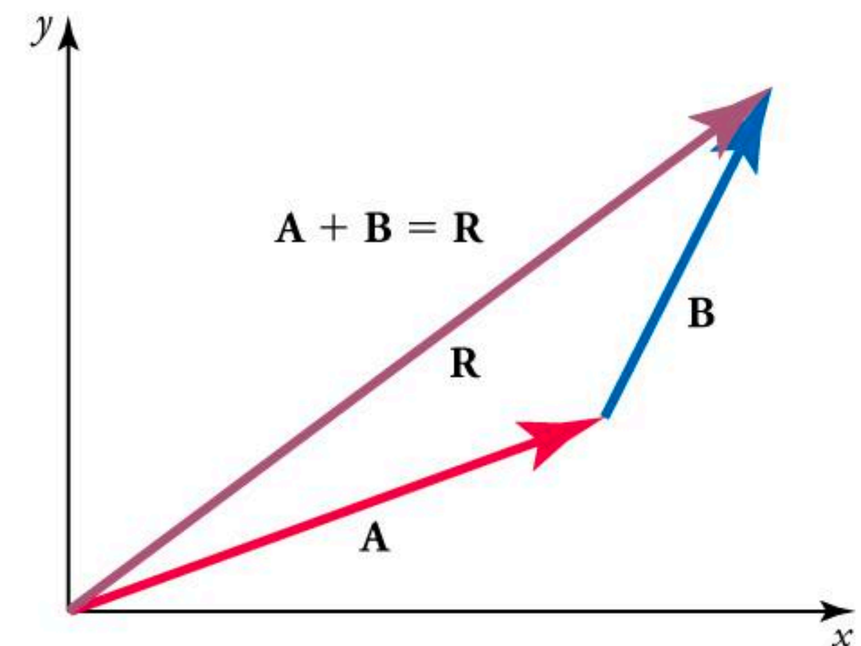
-Weight by inverse word frequency
contextual/universal frequency

(Arora et al., 2017)

Sentence Embeddings

Simple

Add up the vectors of individual words



How to combine?
-Unweighted average

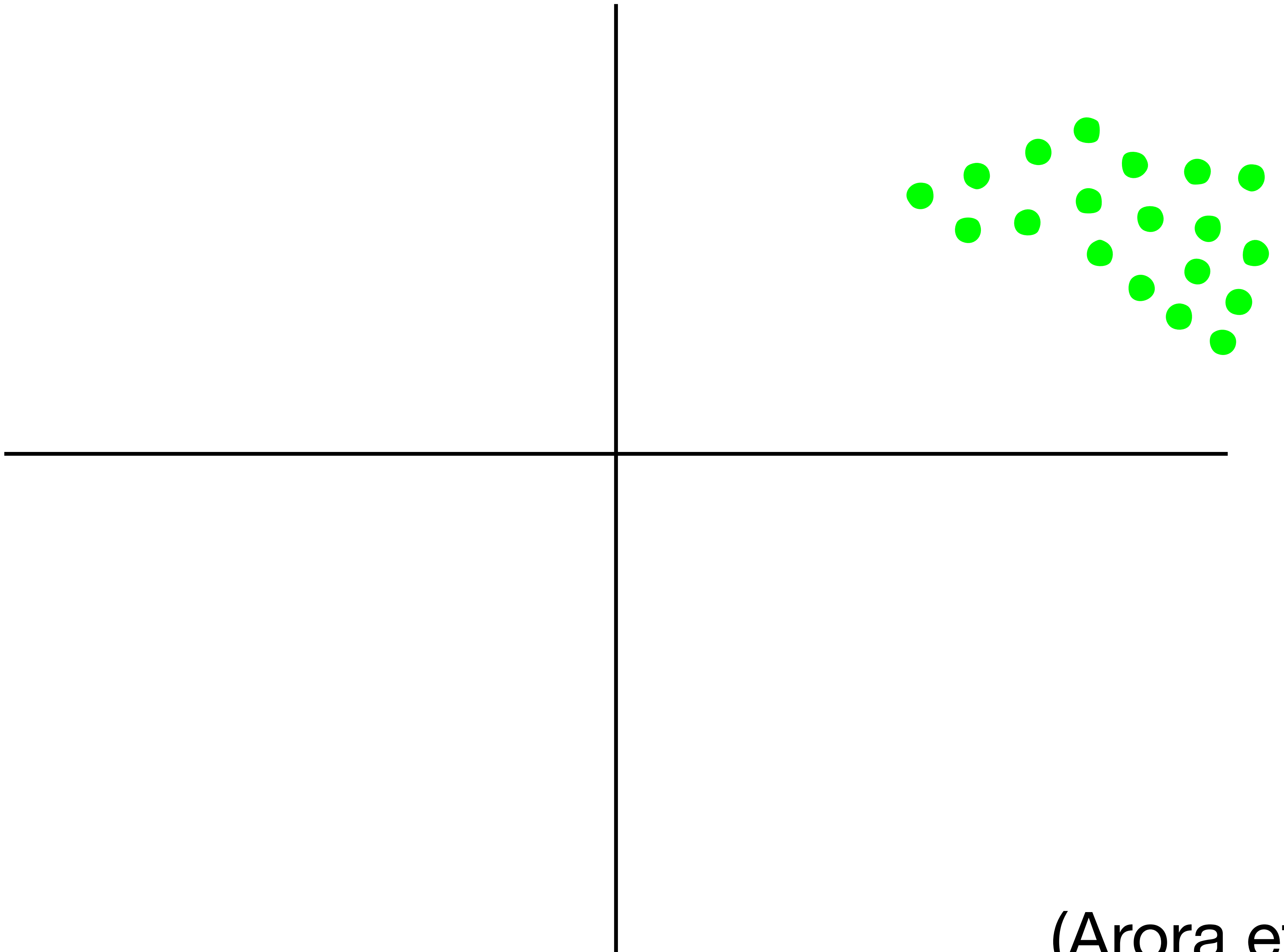
-Weight by inverse word frequency
contextual/universal frequency

- Remove first principal component
(driven by in-context topics)

(Arora et al., 2017)

Sentence Embeddings

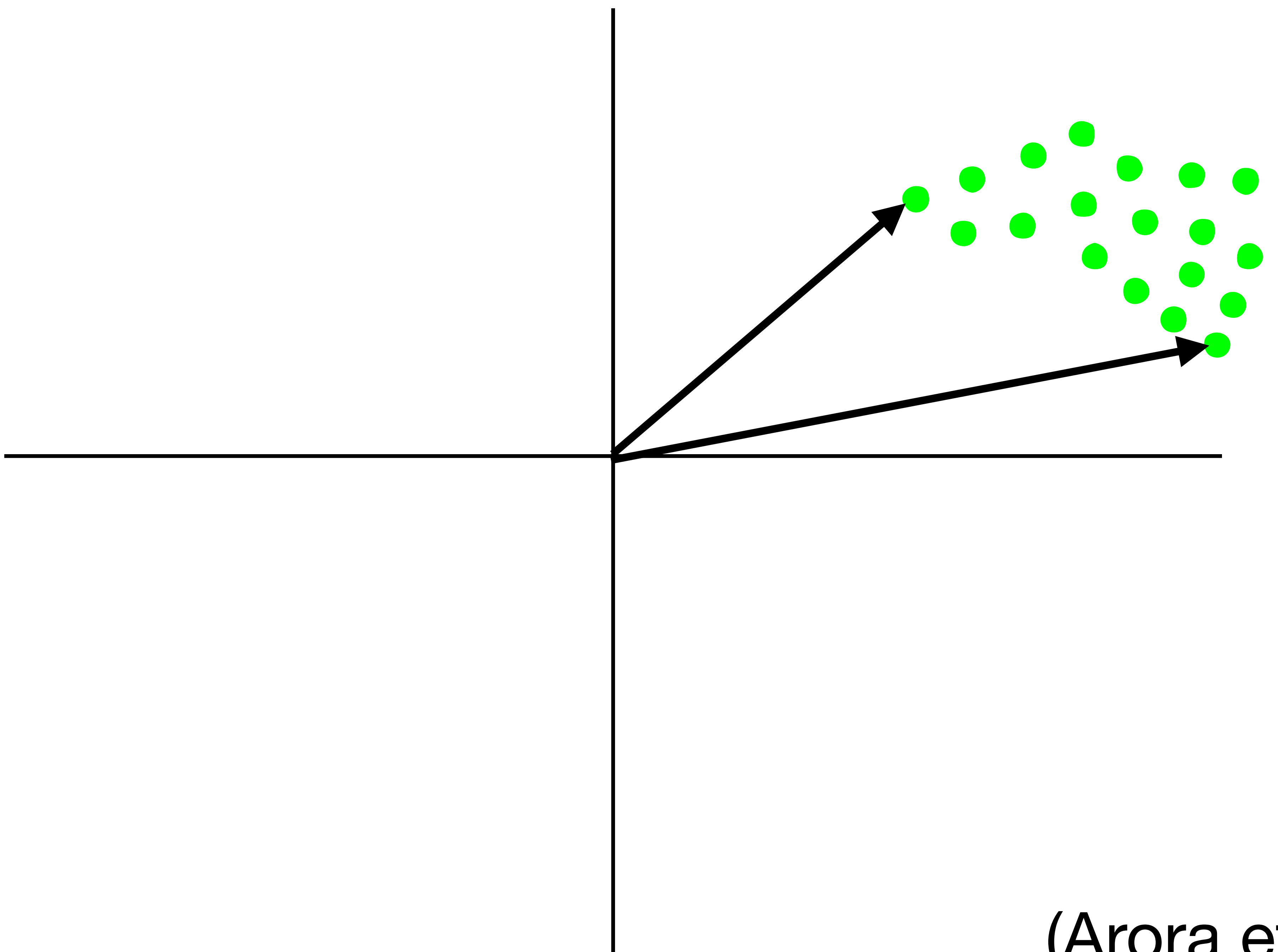
- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

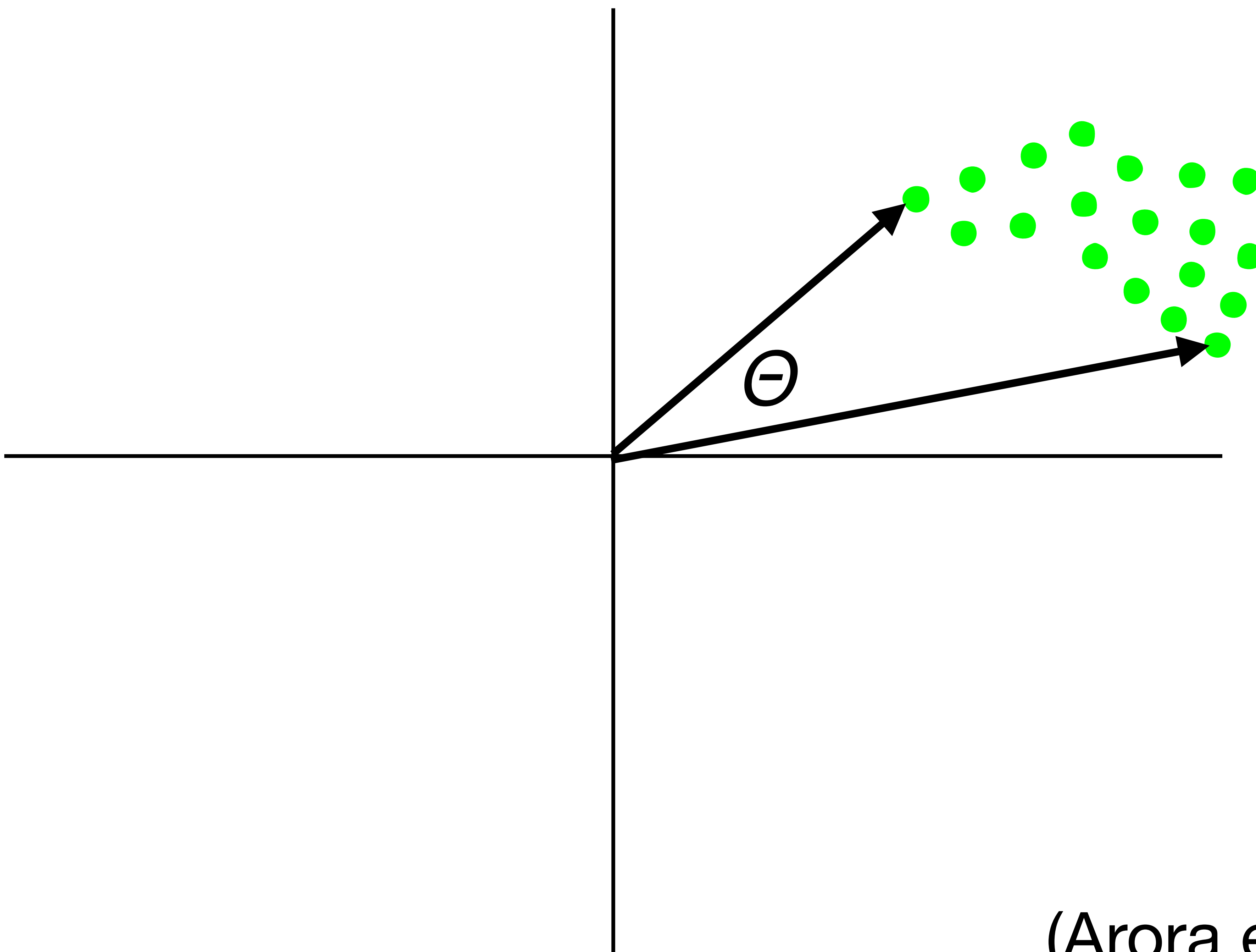
- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

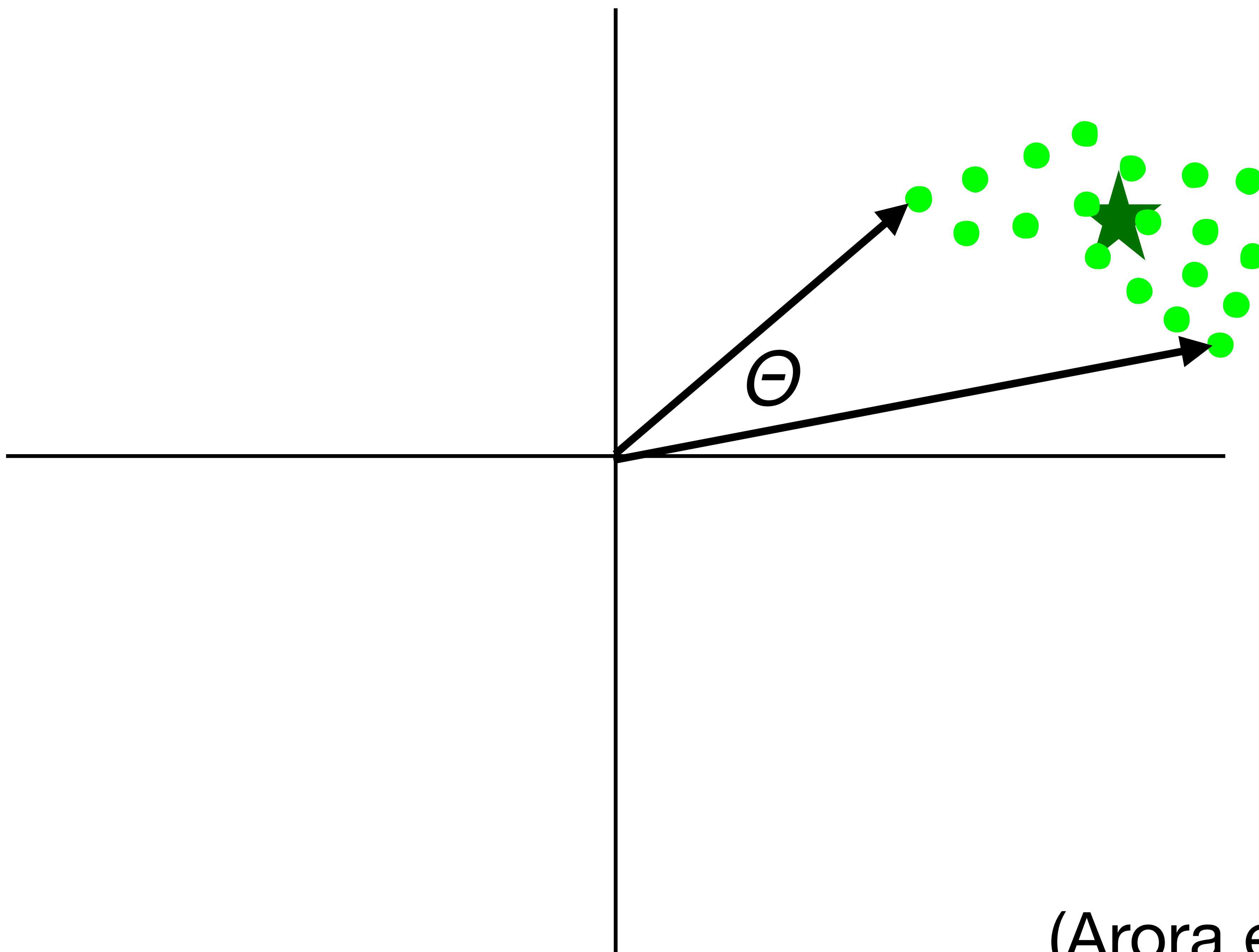
- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

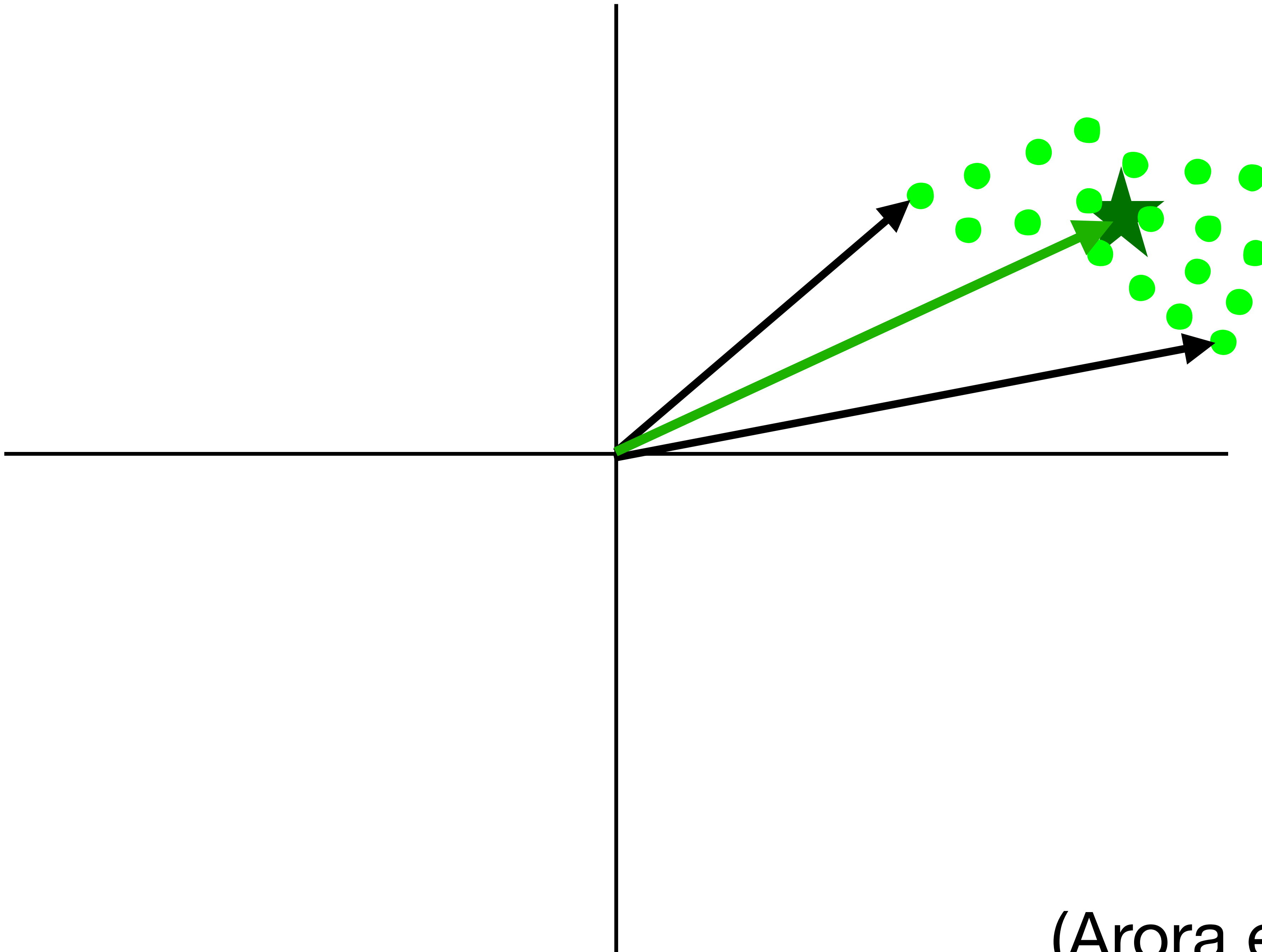
- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

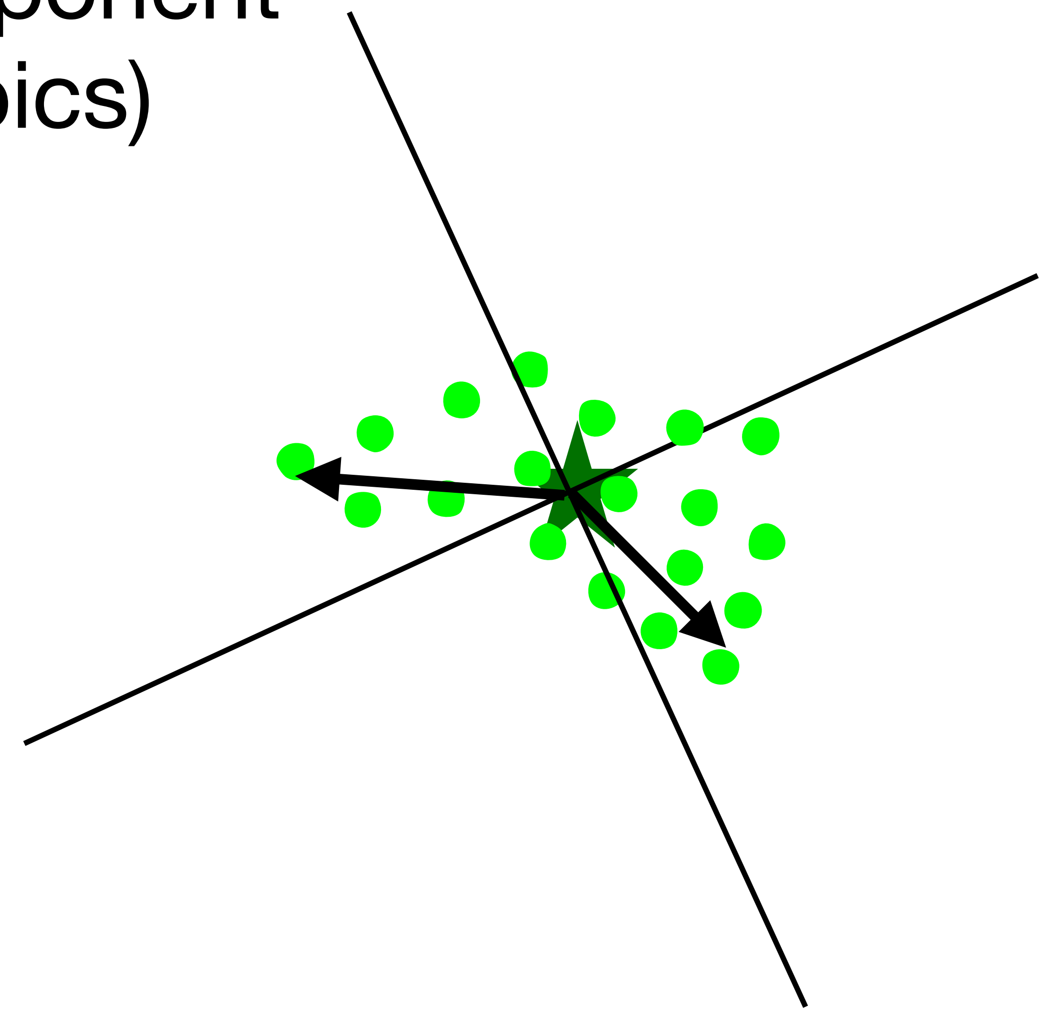
- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

- Remove first principal component
  (driven by in-context topics)



(Arora et al., 2017)

Sentence Embeddings

More complex

Train embeddings for whole sentences
(BERT, GPT-2, etc.)

Sentence Embeddings

More complex

Train embeddings for whole sentences
(BERT, GPT-2, etc.)

Many more layers, and attention to whole sentence

Sentence Embeddings

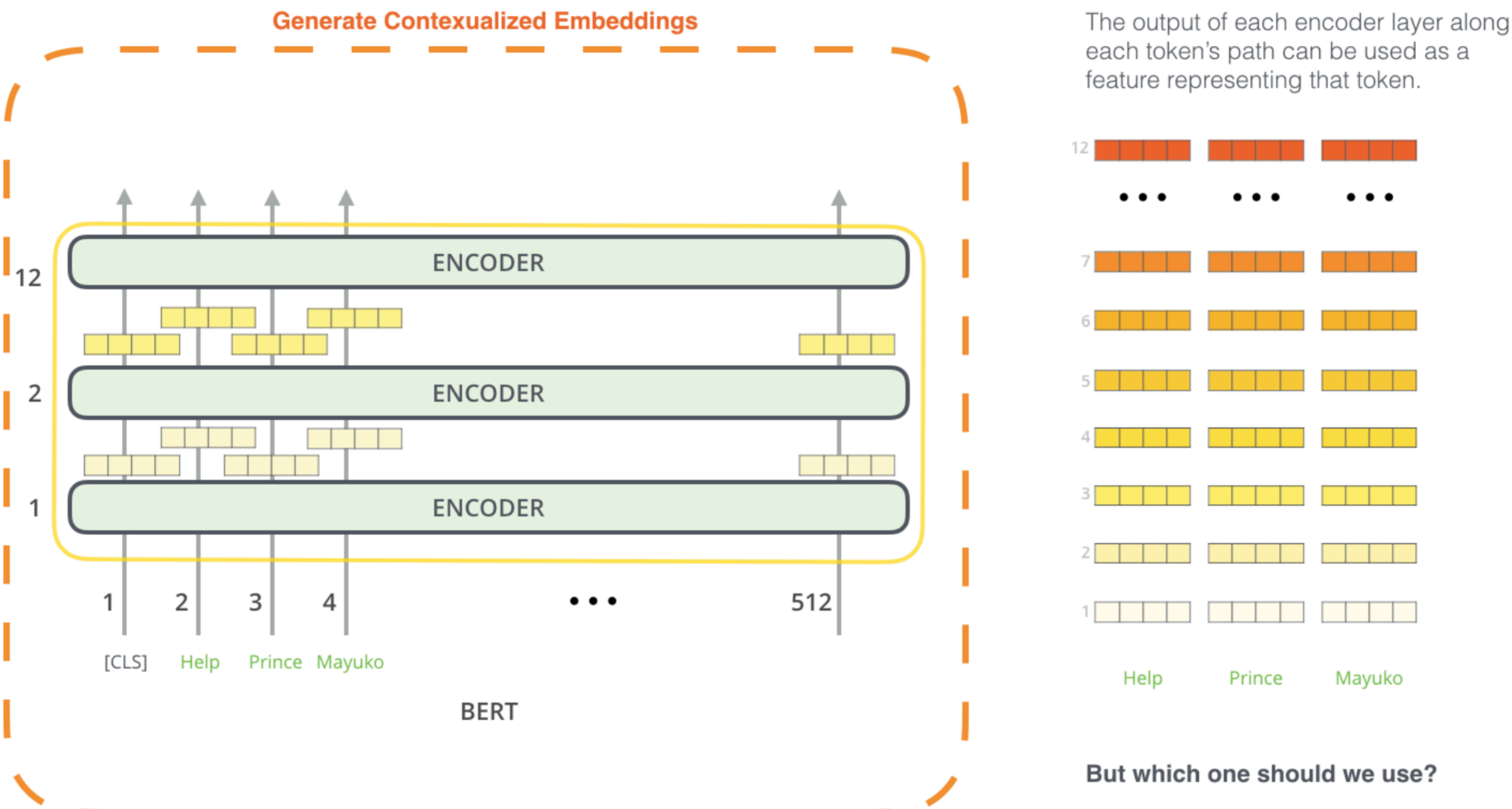
More complex

Train embeddings for whole sentences
(BERT, GPT-2, etc.)

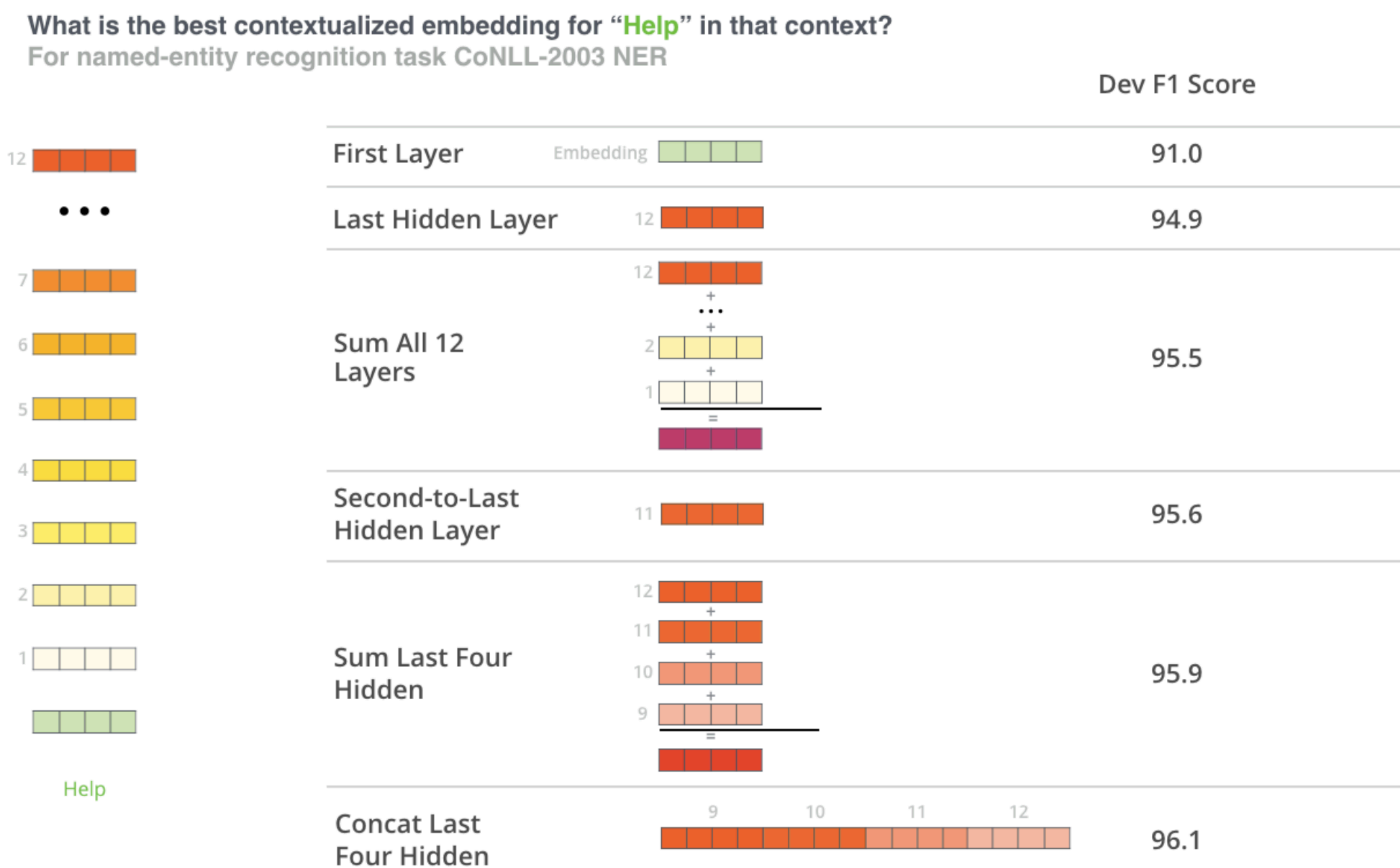
Many more layers, and attention to whole sentence

Two objectives:
- Masked word prediction
- Two-sentence pairs: does one follow the other?

Sentence Embeddings

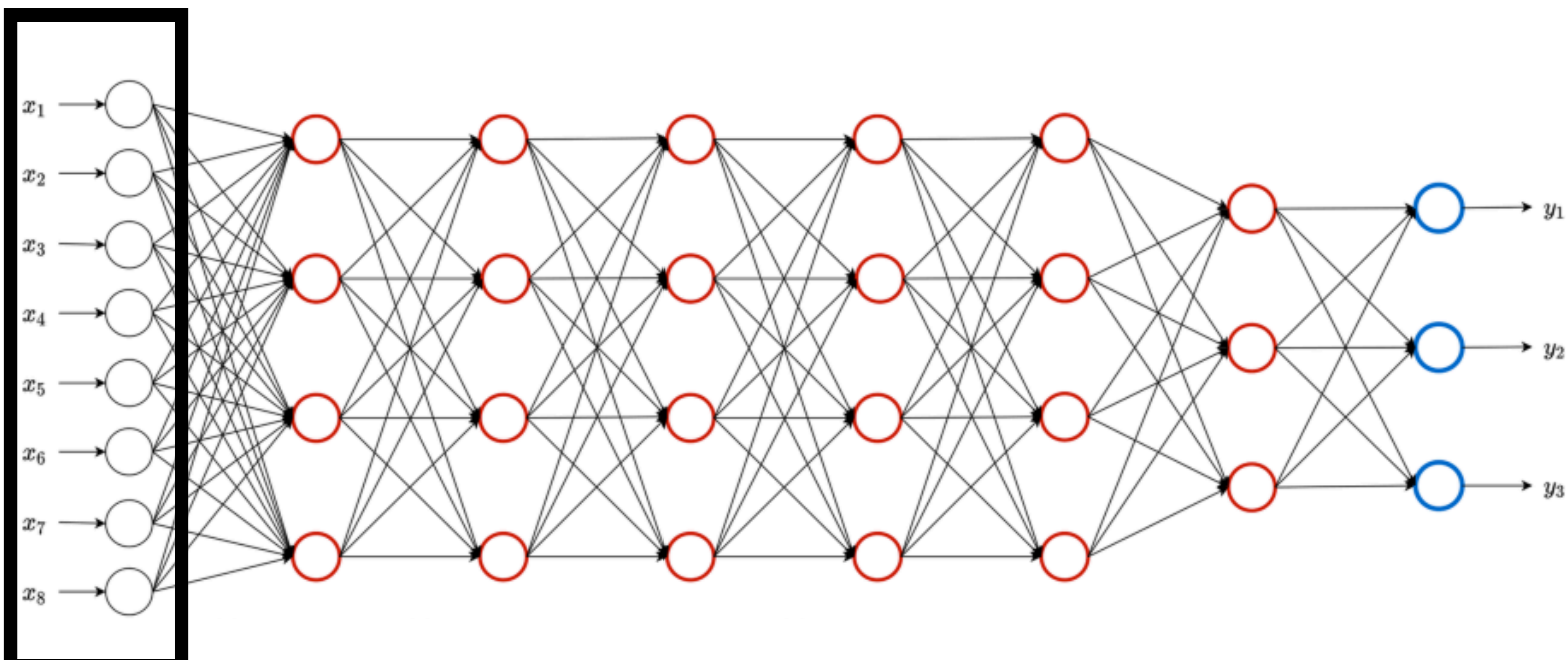


Sentence Embeddings



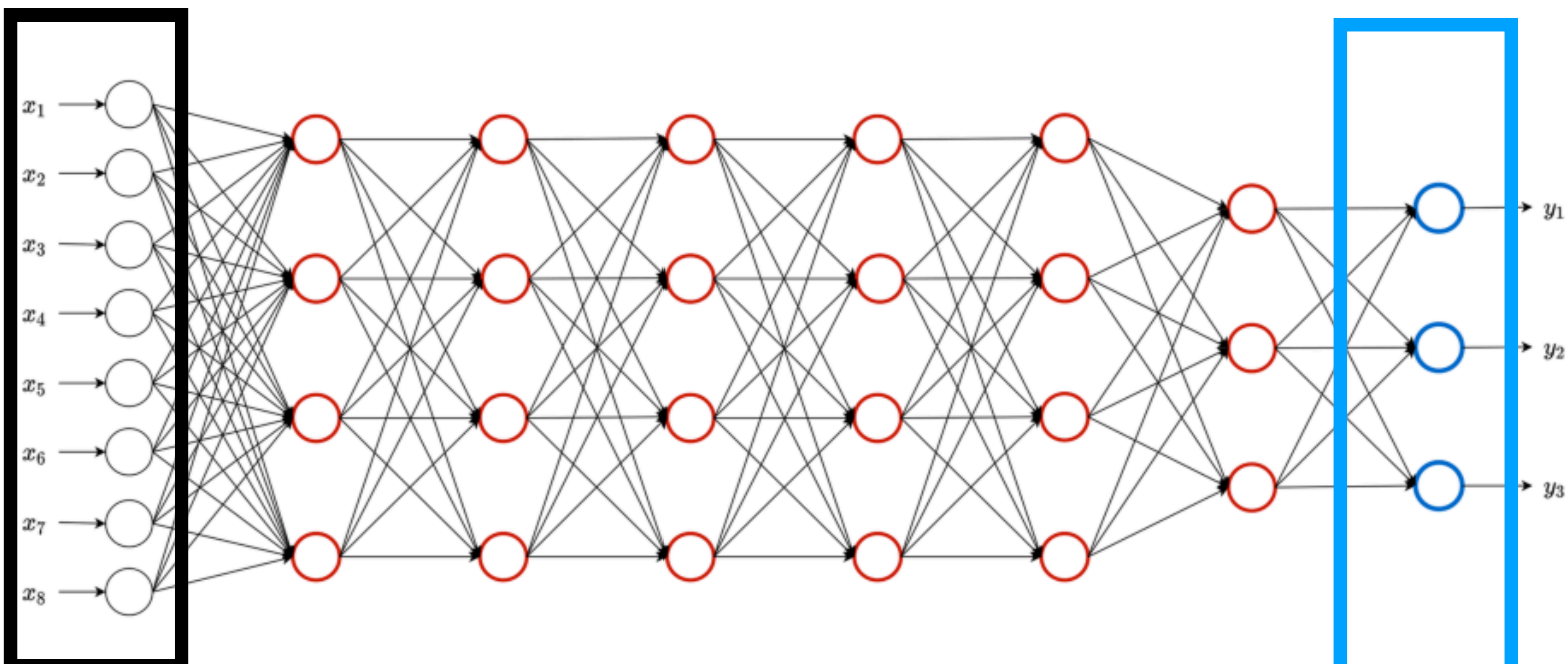
Pre-Trained Models

Words in documents



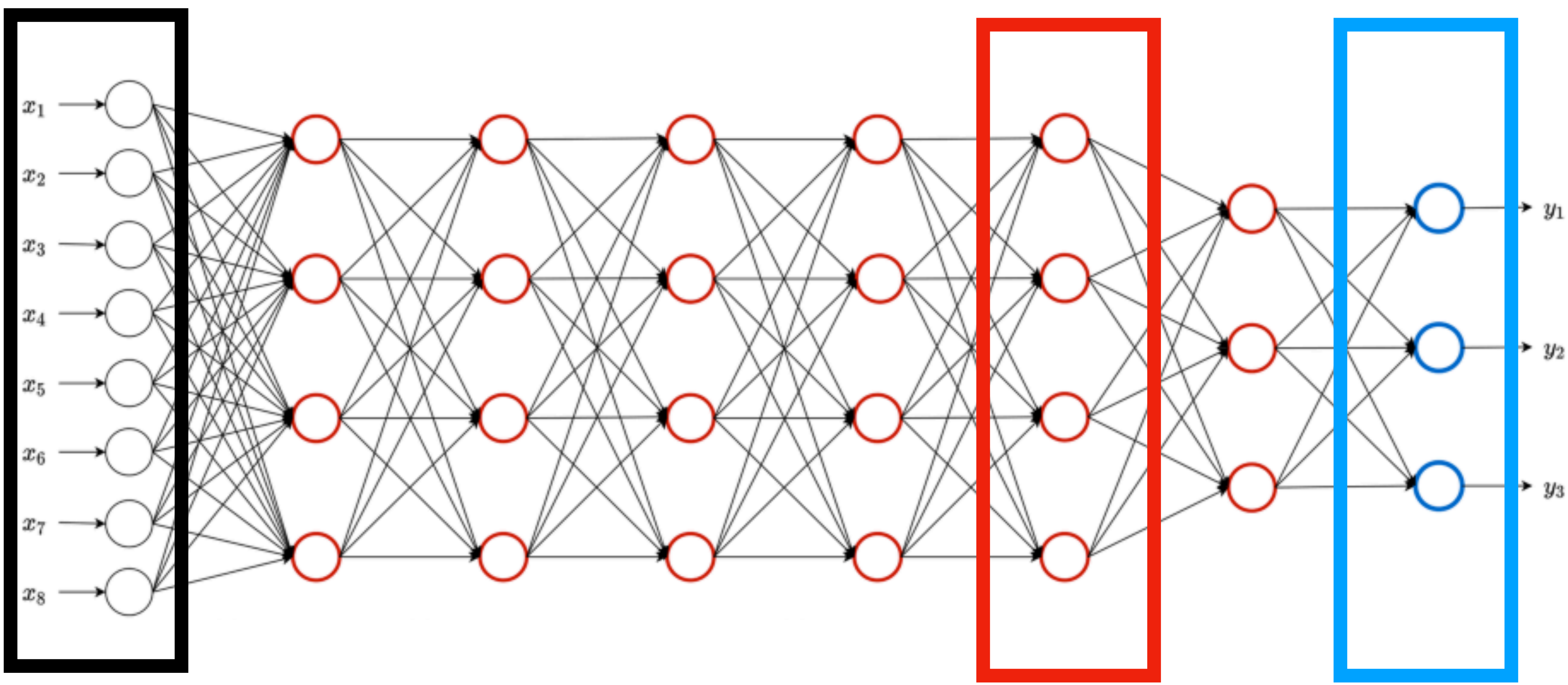
Pre-Trained Models

Masked word Prediction



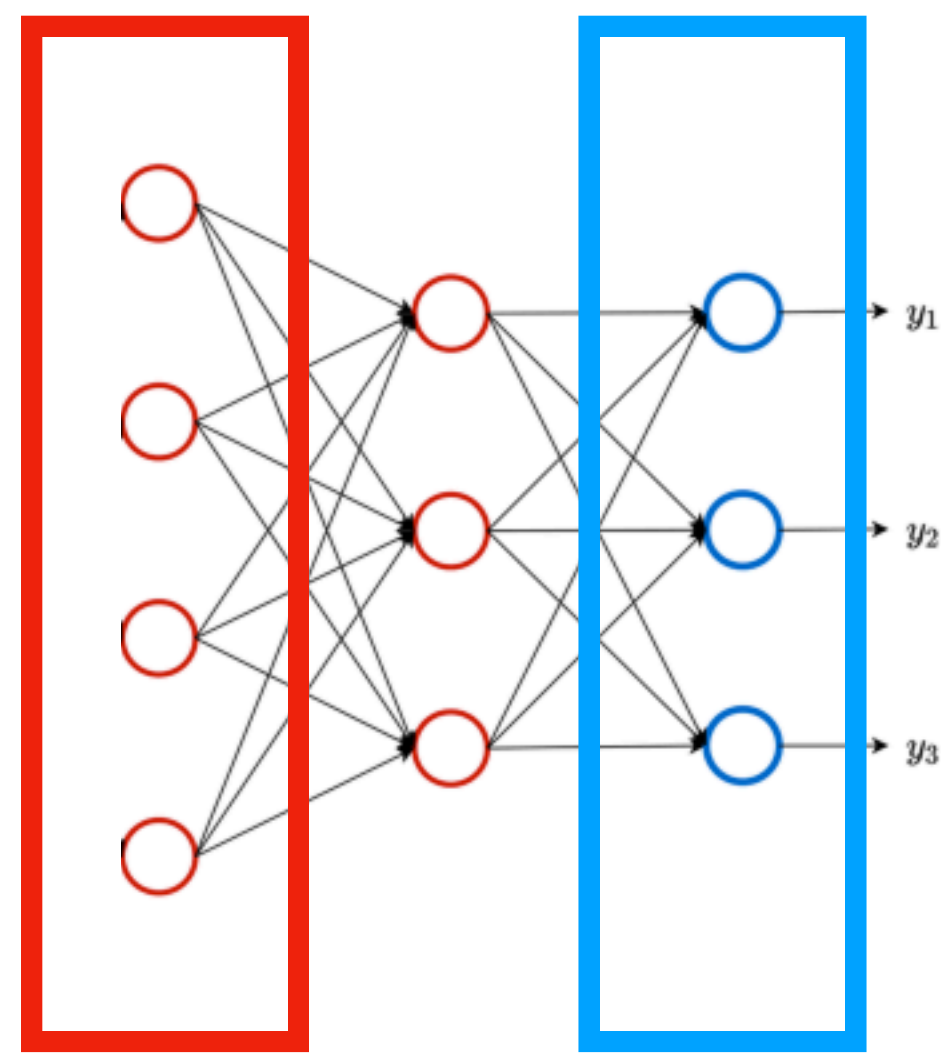
Pre-Trained Models

The vector representation



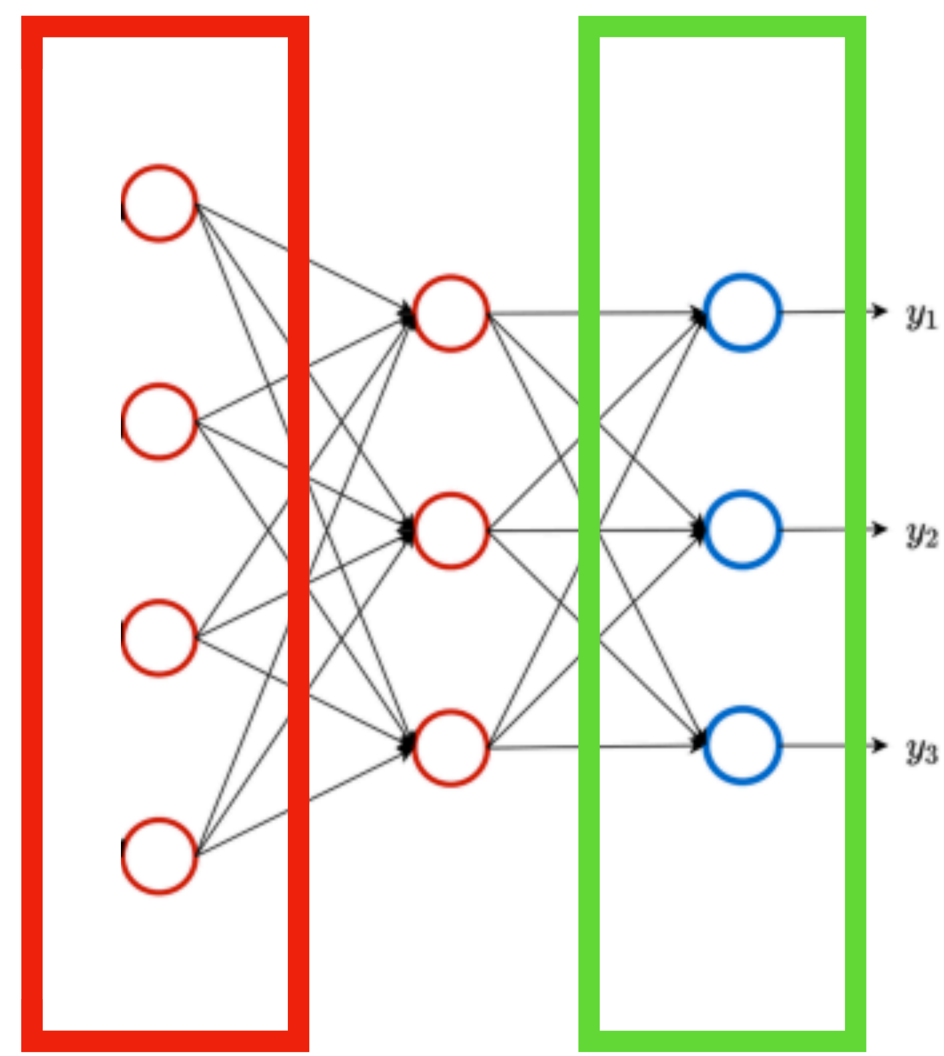
Pre-Trained Models

The vector representation

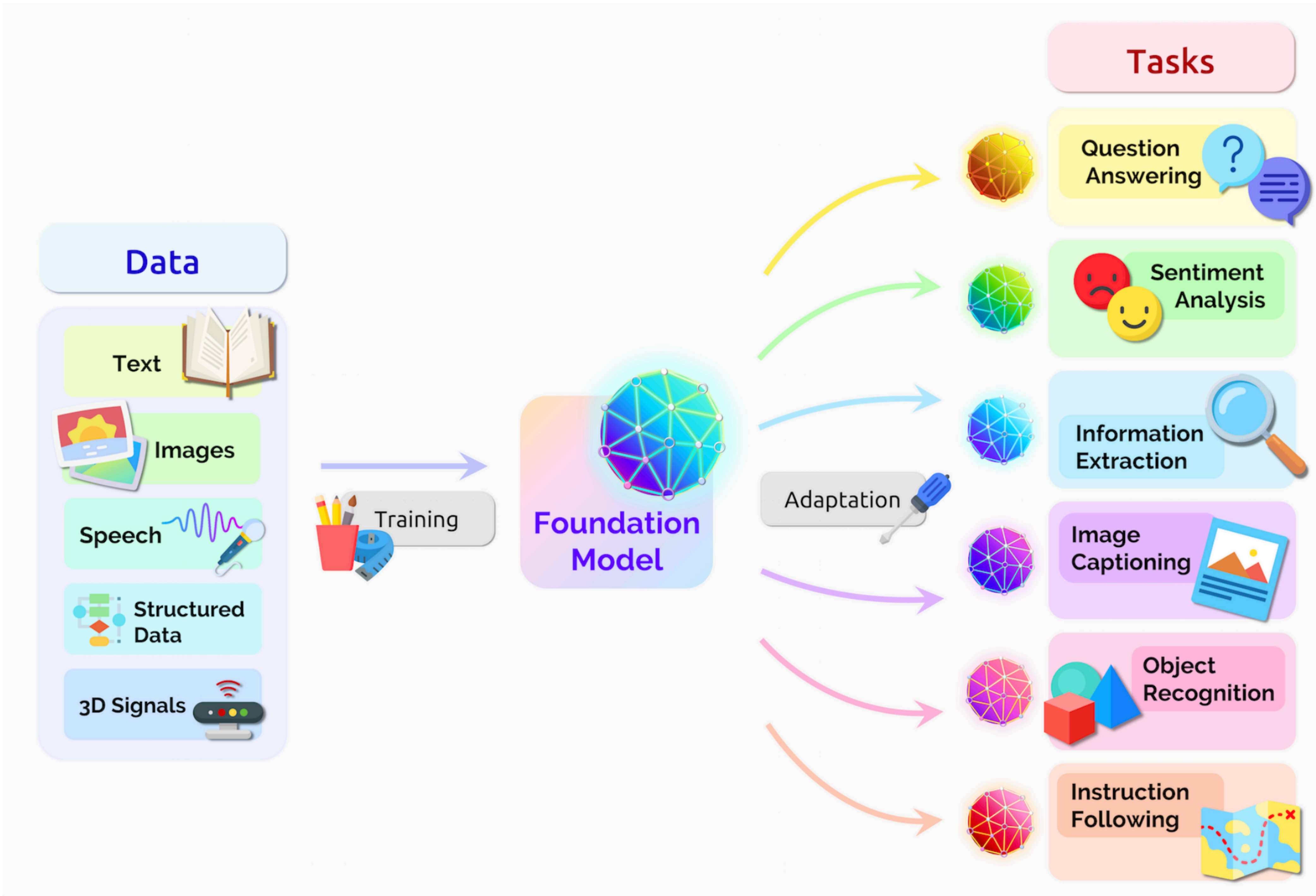


Pre-Trained Models

New quantity of interest

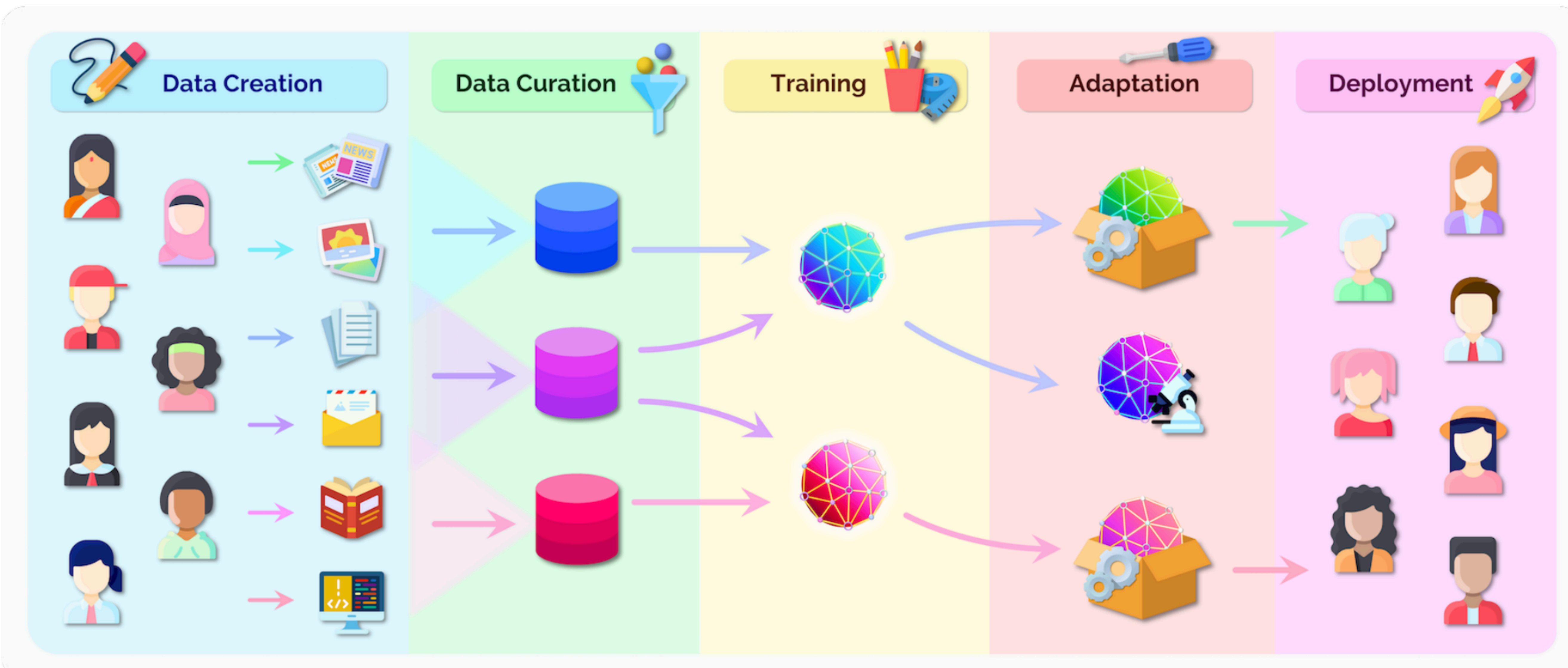


Pre-Trained Models



(Bommasani et al., 2021)

Pre-Trained Models



(Bommasani et al., 2021)

Next Steps in Meaning

Next Steps in Meaning

Benchmark Tasks

Next Steps in Meaning

Benchmark Tasks

Natural Language Inference

P[a]	A senior is waiting at the window of a restaurant that serves sandwiches.	Relationship
H[b]	A person waits to be served his food.	Entailment
	A man is looking to order a grilled cheese sandwich.	Neutral
	A man is waiting in line for the bus.	Contradiction

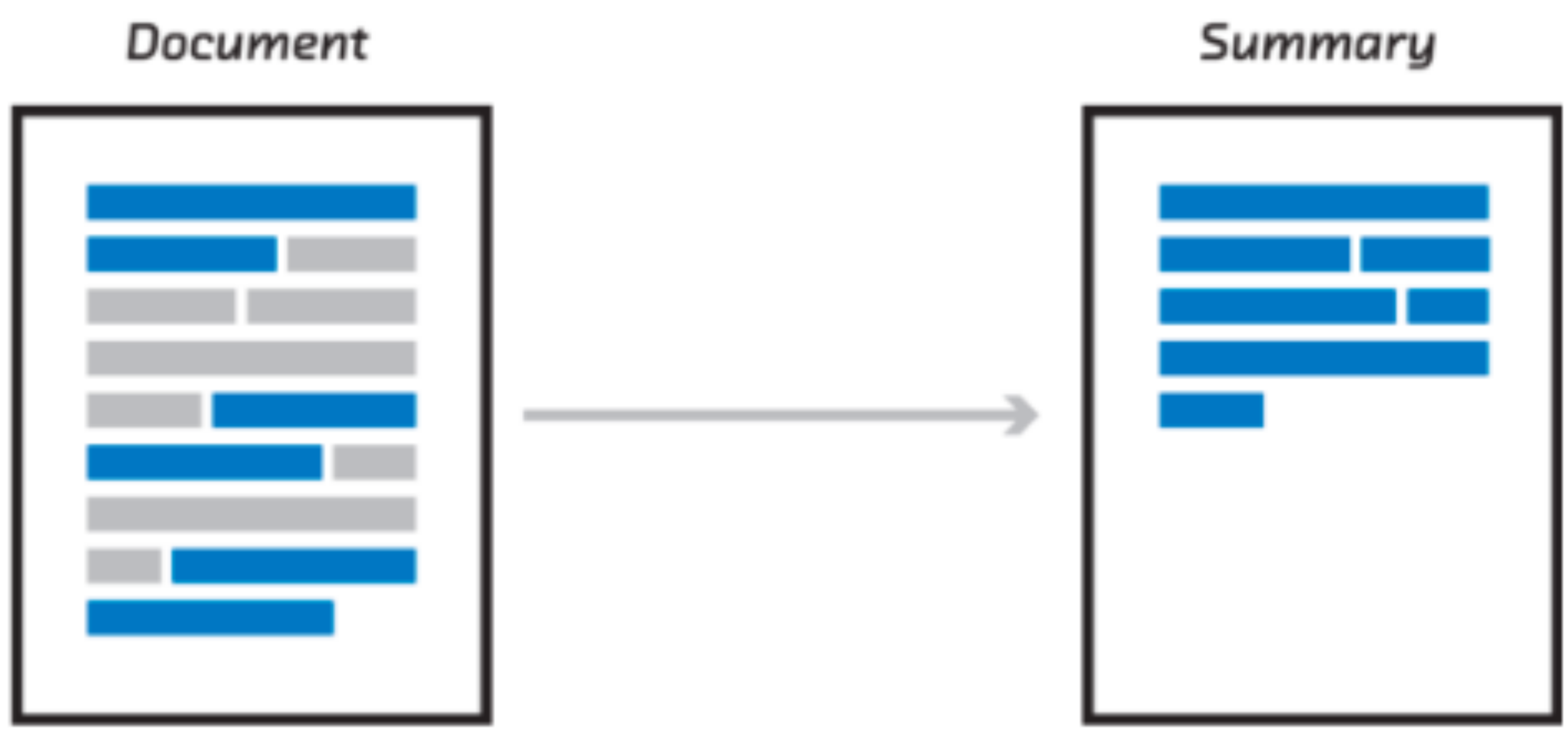
[a]**P**, Premise.
[b]**H**, Hypothesis.

Next Steps in Meaning

Benchmark Tasks

Natural Language Inference

Text Summarization



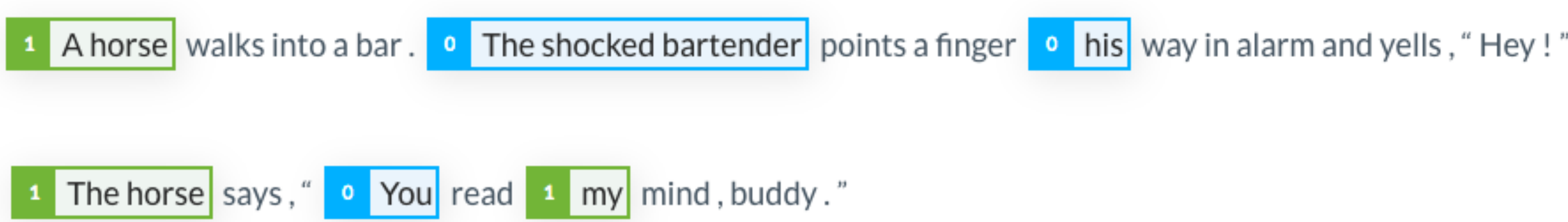
Next Steps in Meaning

Benchmark Tasks

Natural Language Inference

Text Summarization

Co-reference Resolution



Next Steps in Meaning

Benchmark Tasks

Natural Language Inference

Text Summarization

Co-reference Resolution

Question Answering

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

Next Steps in Meaning

Benchmark Tasks

System	MNLI-(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Average**
	392k	363k	108k	67k	8.5k	5.7k	3.5k	2.5k	-
Pre-OpenAI SOTA	80.6/80.1	66.1	82.3	93.2	35.0	81.0	86.0	61.7	74.0
BiLSTM+ELMo+Attn	76.4/76.1	64.8	79.8	90.4	36.0	73.3	84.9	56.8	71.0
OpenAI GPT	82.1/81.4	70.3	87.4	91.3	45.4	80.0	82.3	56.0	75.1
$\text{BERT}_{\text{BASE}}$	84.6/83.4	71.2	90.5	93.5	52.1	85.8	88.9	66.4	79.6
$\text{BERT}_{\text{LARGE}}$	**86.7/85.9**	**72.1**	**92.7**	**94.9**	**60.5**	**86.5**	**89.3**	**70.1**	**82.1**

Table 1: GLUE Test results, scored by the evaluation server (https://gluebenchmark.com/leaderboard). The number below each task denotes the number of training examples. The "Average" column is slightly different than the official GLUE score, since we exclude the problematic WNLI set.[8] BERT and OpenAI GPT are single-model, single task. F1 scores are reported for QQP and MRPC, Spearman correlations are reported for STS-B, and accuracy scores are reported for the other tasks. We exclude entries that use BERT as one of their components.

Devlin et al., 2017

Next Steps in Meaning

Benchmark Tasks

	PP	DAN	RNN	LSTM (no)	LSTM (o.g.)	skip-thought	Ours
similarity (SICK)	84.9	85.96	73.13	85.45	83.41	85.8	**86.03**
entailment (SICK)	83.1	84.5	76.4	83.2	82.0	-	**84.6**
sentiment (SST)	79.4	83.4	86.5	86.6	**89.2**	-	82.2

Table 2: Results on similarity, entailment, and sentiment tasks. The sentence embeddings are computed unsupervisedly, and then used as features in downstream supervised tasks. The row for similarity (SICK) shows Pearson's $r \times 100$ and the last two rows show accuracy. The highest score in each row is in boldface. Results in Column 2 to 6 are collected from (Wieting et al., 2016), and those in Column 7 for skip-thought are from (Lei Ba et al., 2016).

Arora et al., 2017

Embeddings Applications

Embeddings Applications

Supervised Learning

Embeddings as extracted features

Embeddings Applications

Supervised Learning

Embeddings as extracted features

$$\hat{y} = a_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \ldots + e$$

***x* =** projections into pre-trained embedding space

Embeddings Applications

Supervised Learning

Embeddings as extracted features

Similarity

Data exploration
- which document is most like this one?

Embeddings Applications

Supervised Learning

Embeddings as extracted features

Similarity

Data exploration
- which document is most like this one?

As an extracted feature
- imitation, novelty, topic switching

Accommodation

Accommodation

accommodation, mimicry, reciprocity, entrainment, mirroring...



Dimension	Canonical study
Posture	Condon and Ogston, 1967
Pause length	Jaffe and Feldstein, 1970
Utterance length	Matarazzo and Wiens, 1973
Self-disclosure	Derlenga et al., 1973
Head nodding	Hale and Burgoon, 1984
Backchannels	White, 1989
Linguistic style	Niederhoffer and Pennebaker, 2002

(review in Danescu-Niculescu-Mizil et al., 2012)

Accommodation

(Palagi et al., 2020)

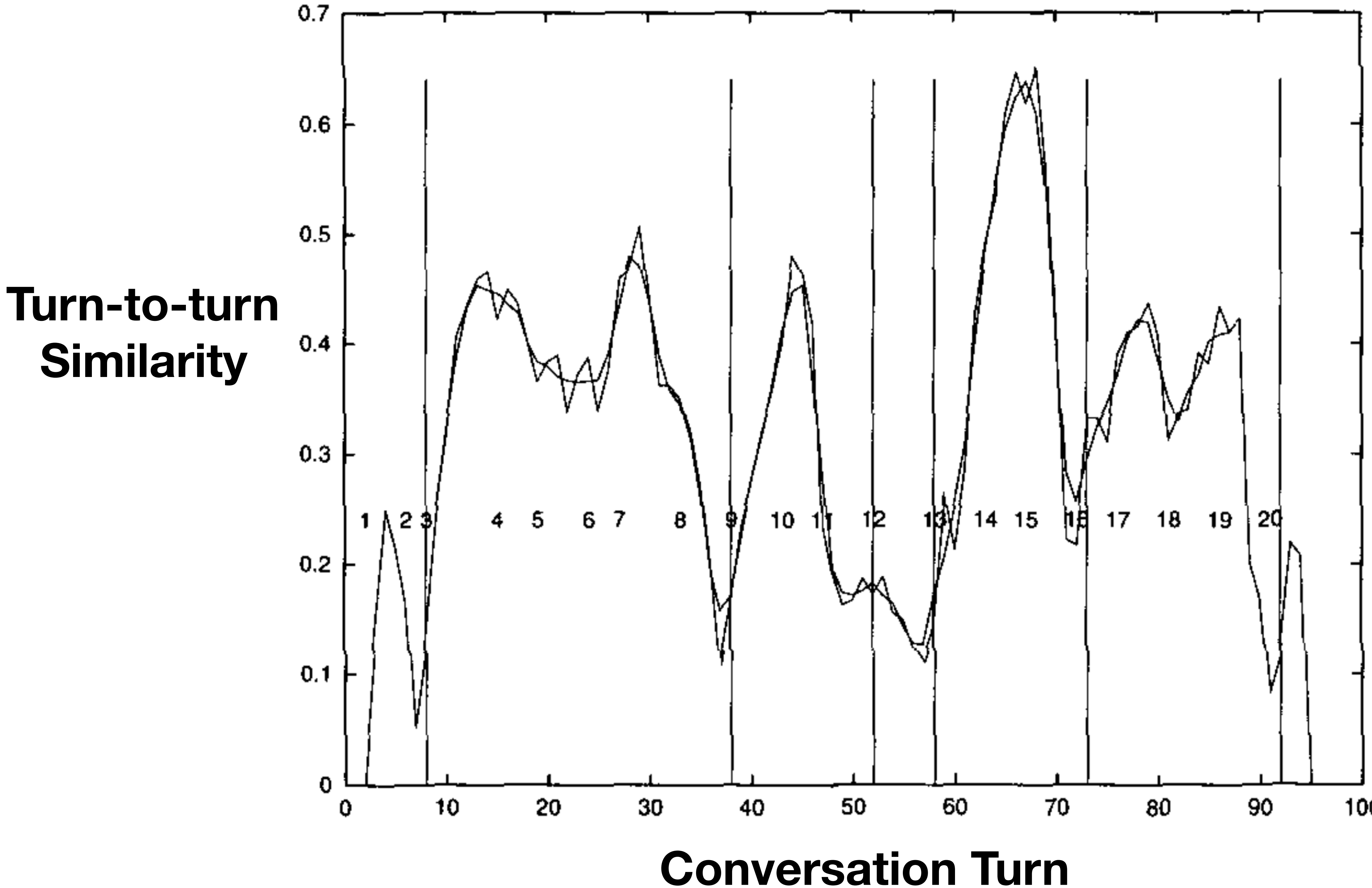
Accommodation

(Palagi et al., 2020)

Accommodation

(Palagi et al., 2020)

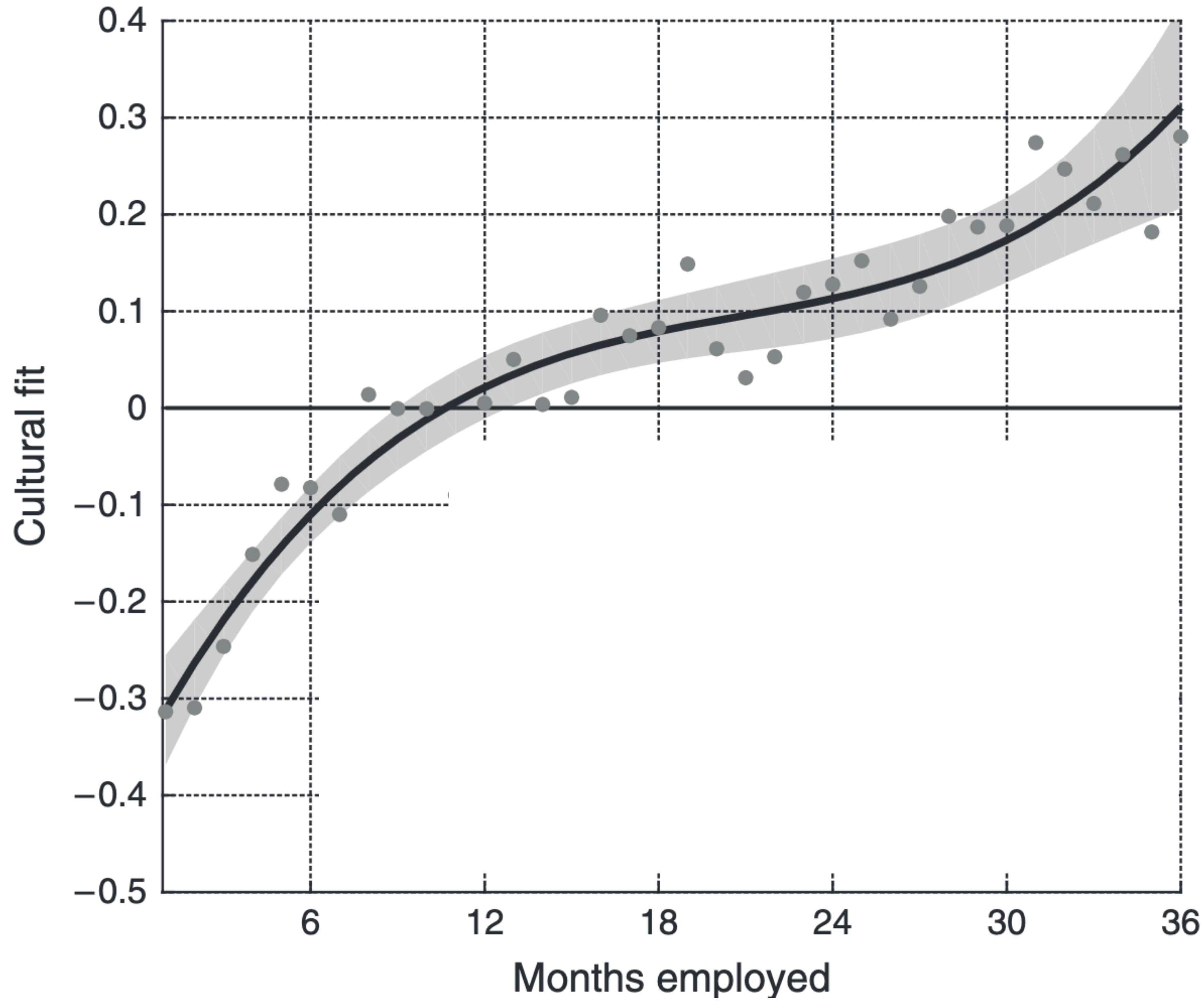
Accommodation



(Hearst, 1997)

Accommodation

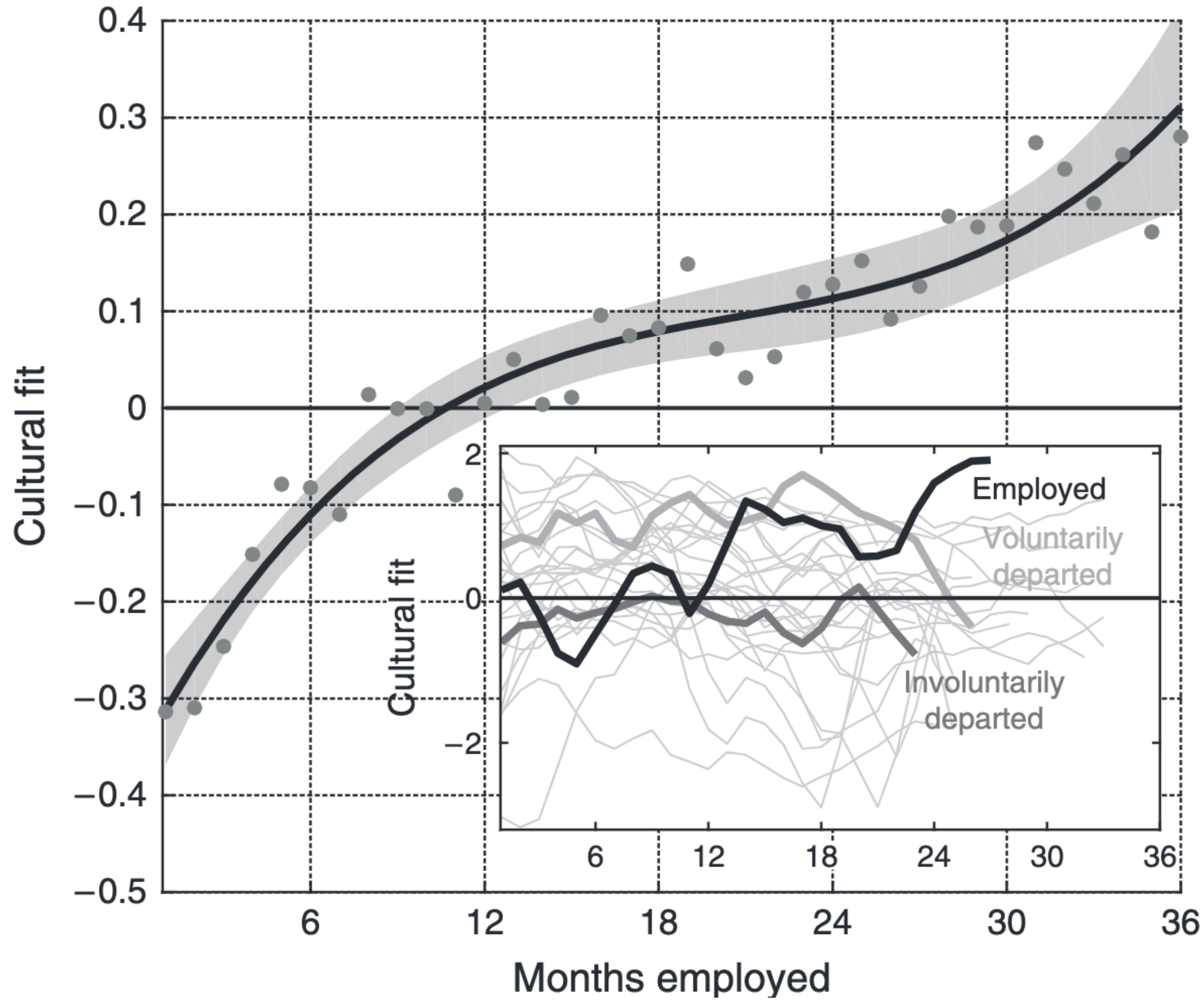
Figure 3. Cultural Fit (Standardized) as a Function of Number of Months Employed



(Srivastava et al., 2018)

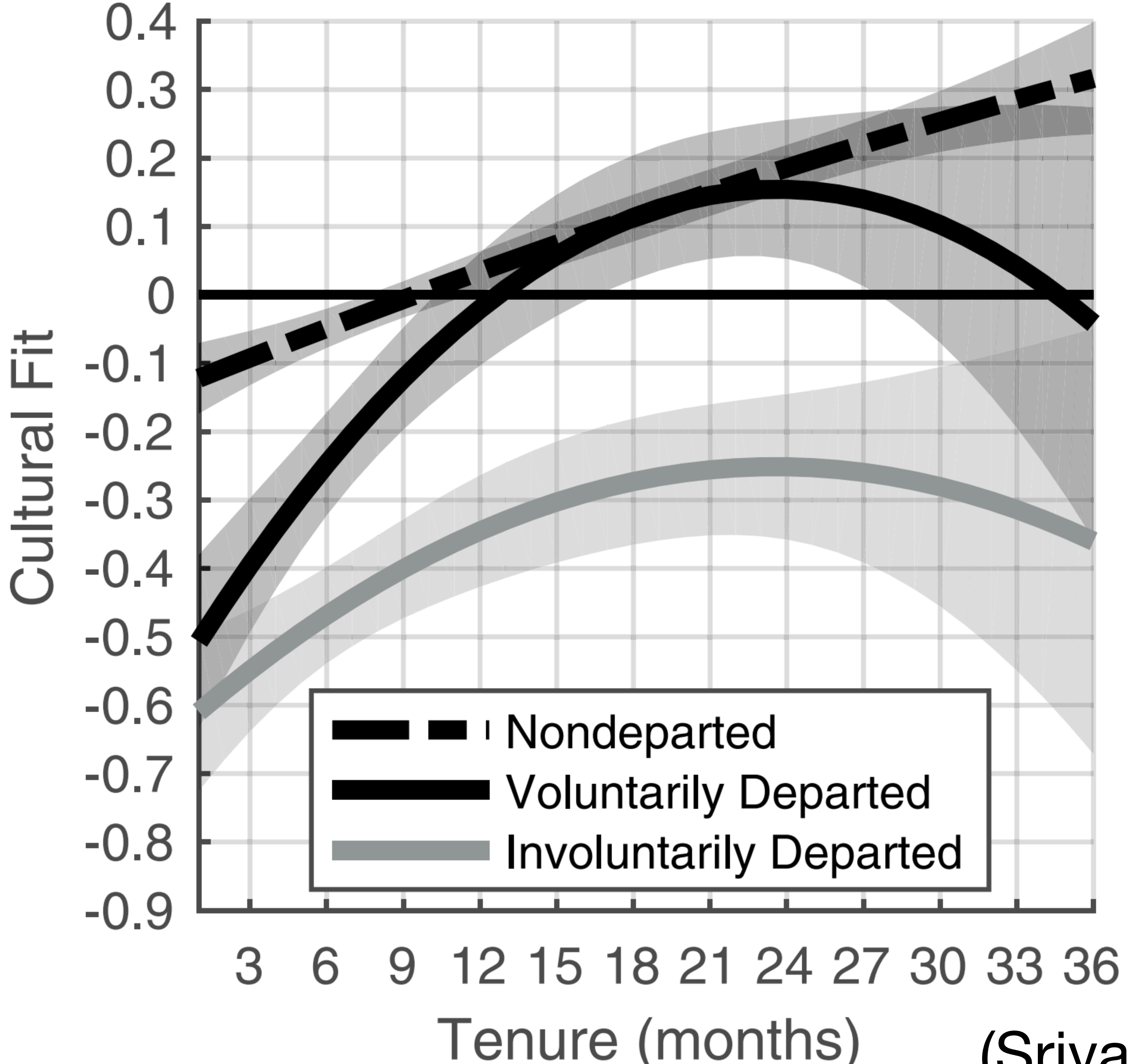
Accommodation

Figure 3. Cultural Fit (Standardized) as a Function of Number of Months Employed



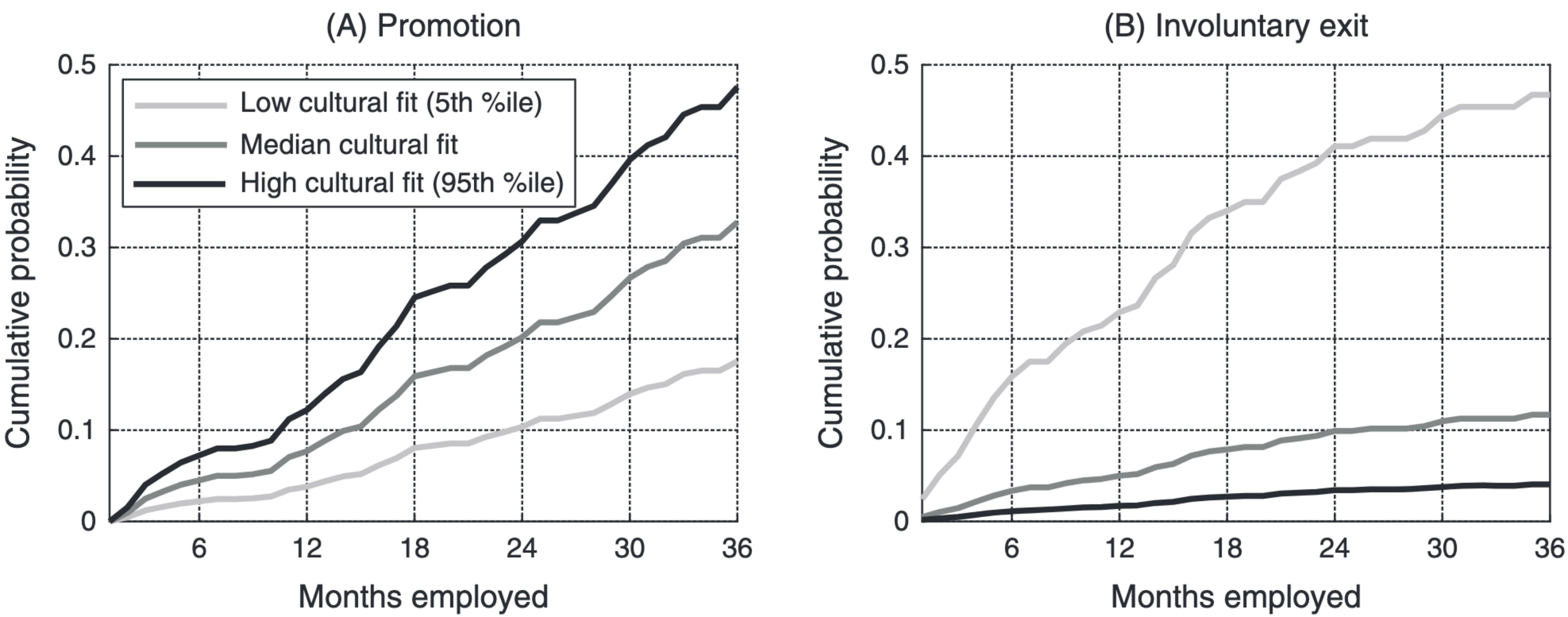
(Srivastava et al., 2018)

Accommodation



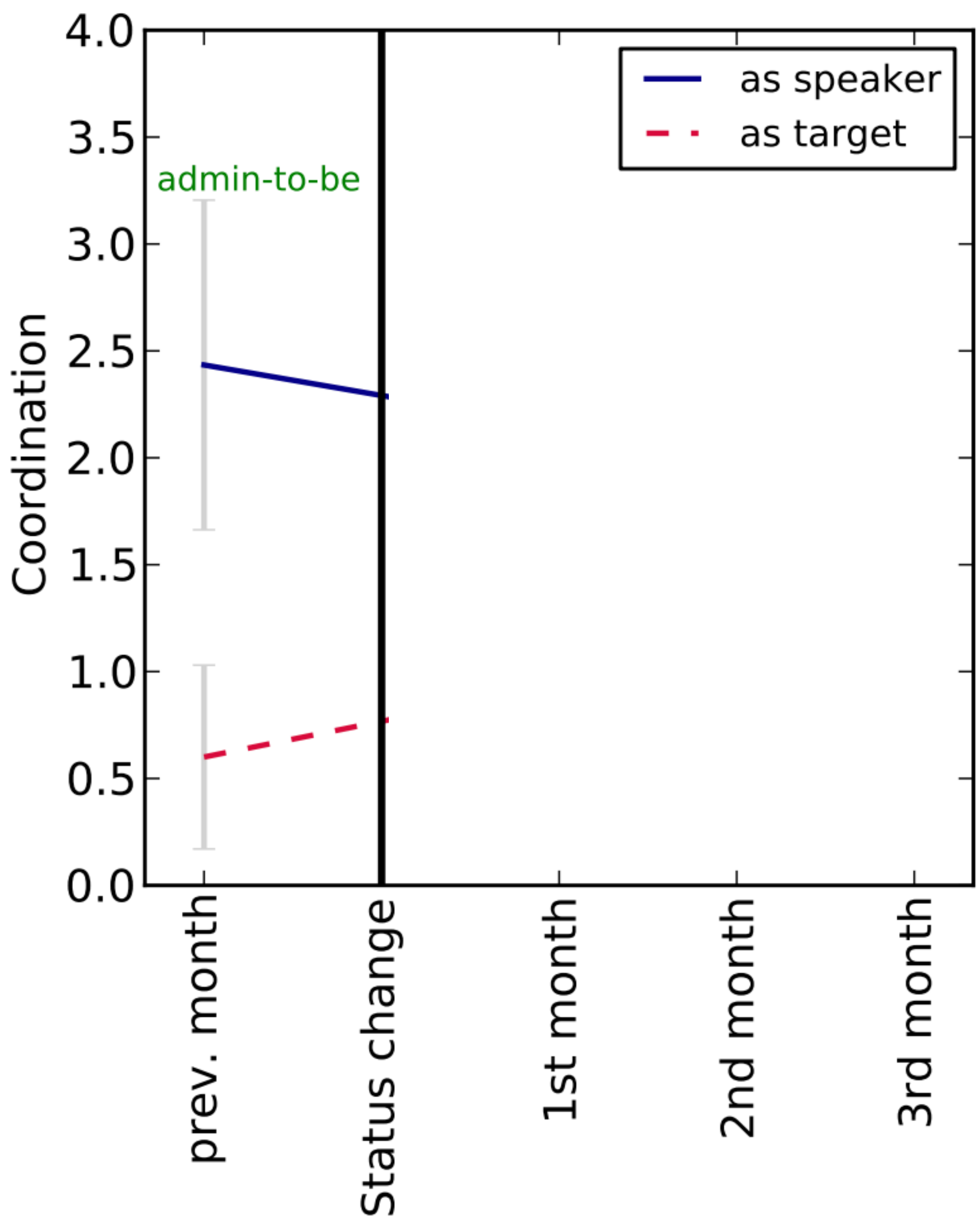
(Srivastava et al., 2018)

Accommodation



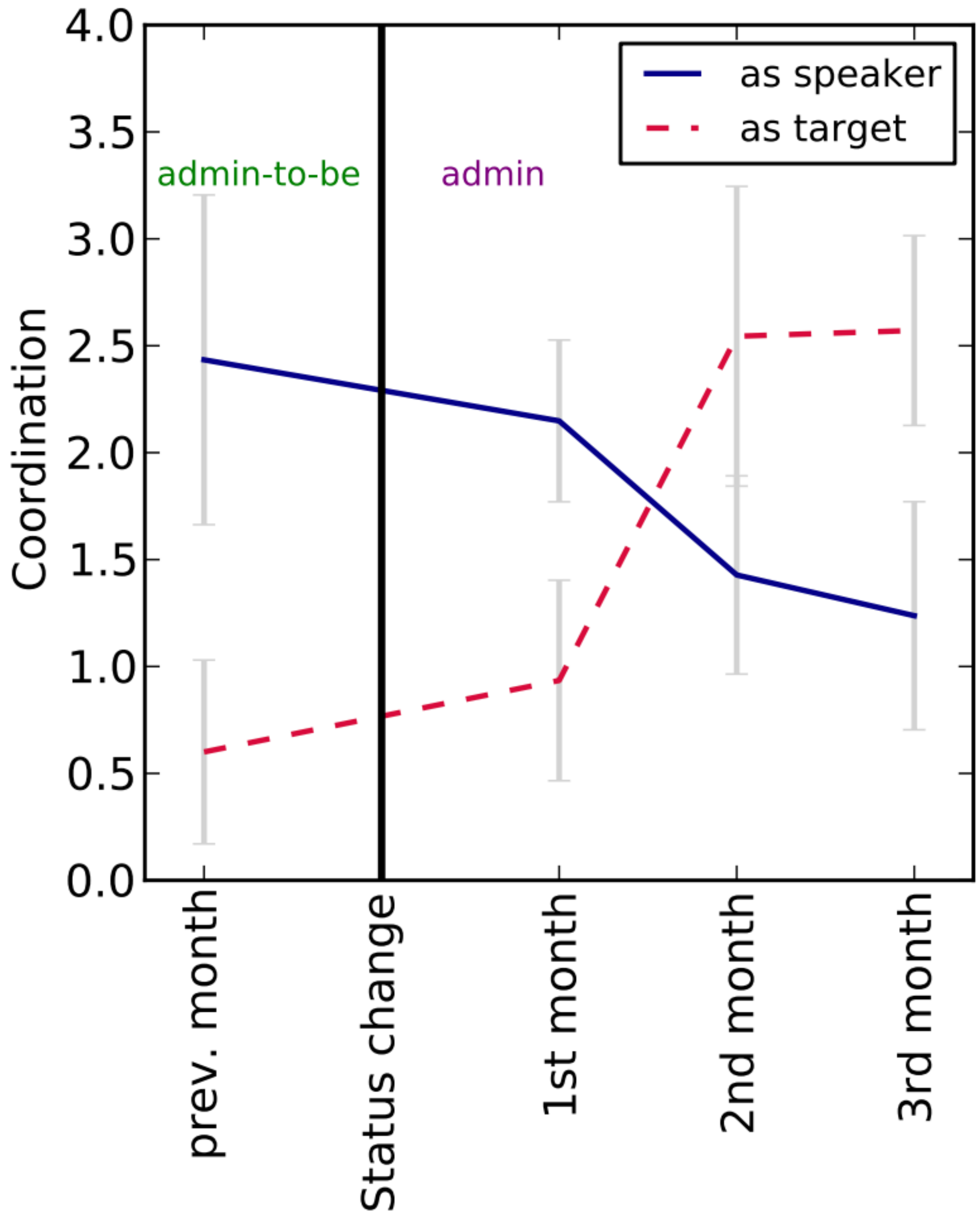
(Srivastava et al., 2018)

Accommodation



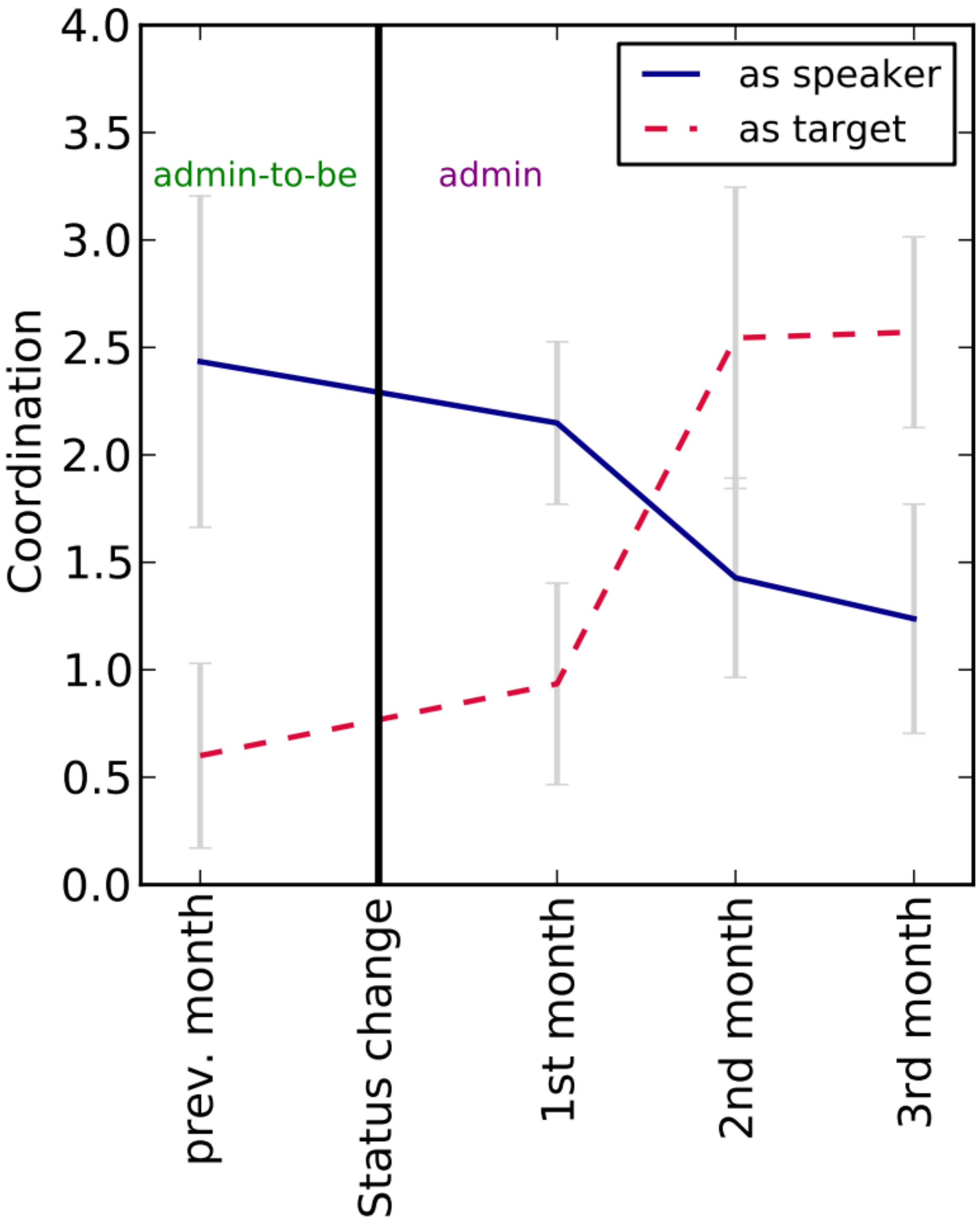
(Danescu-Niculescu-Mizil et al., 2012)

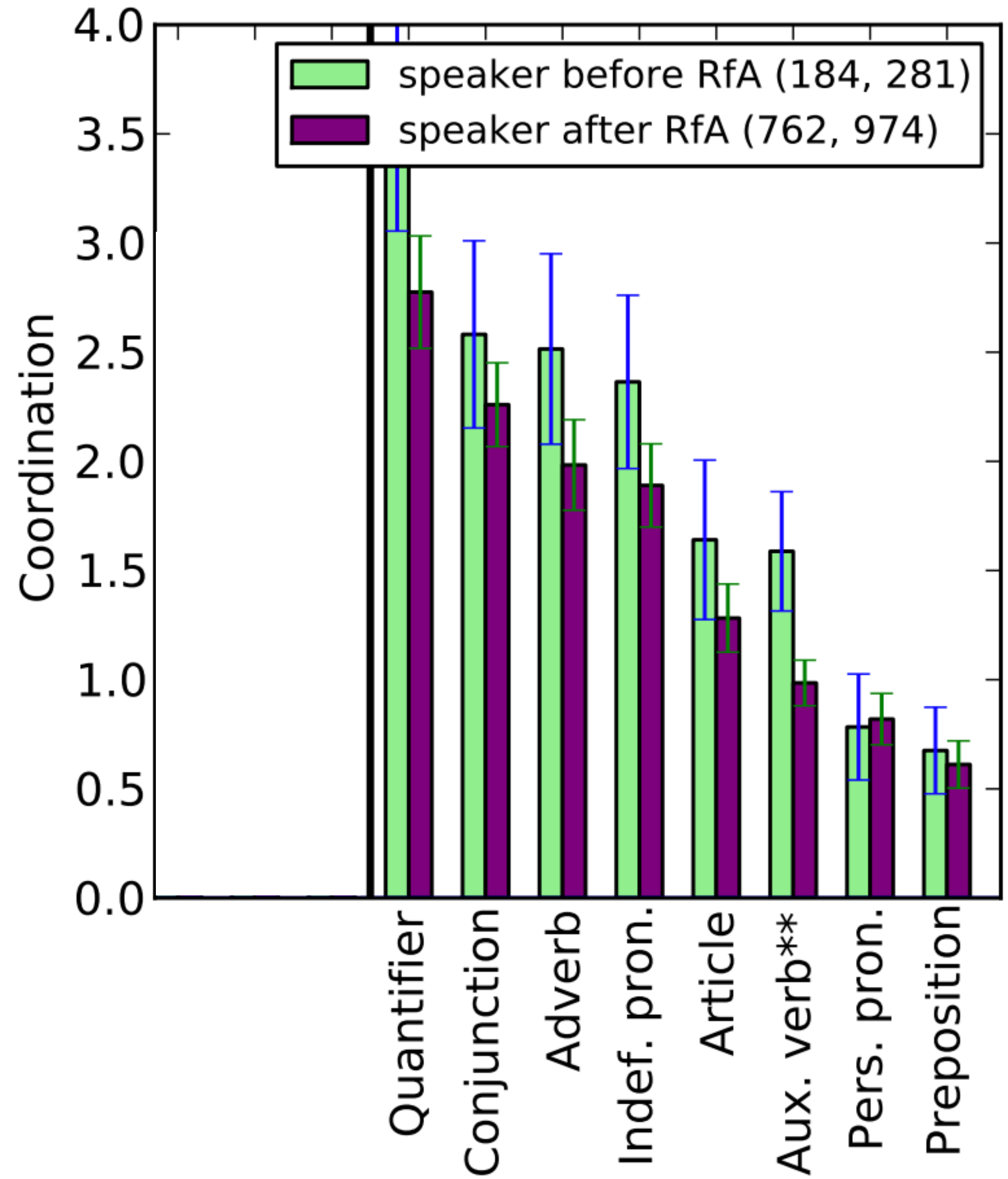
Accommodation



(Danescu-Niculescu-Mizil et al., 2012)

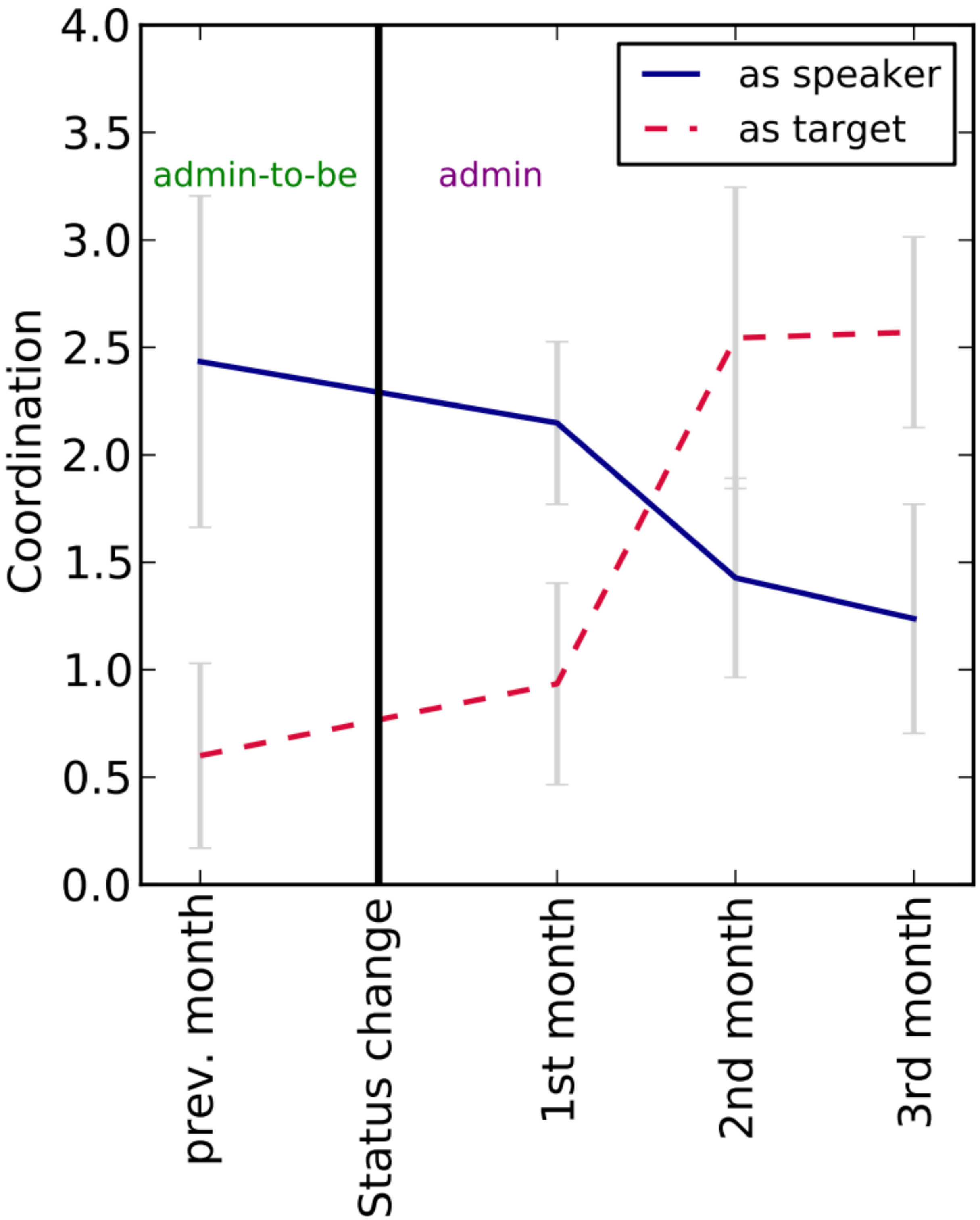
Accommodation

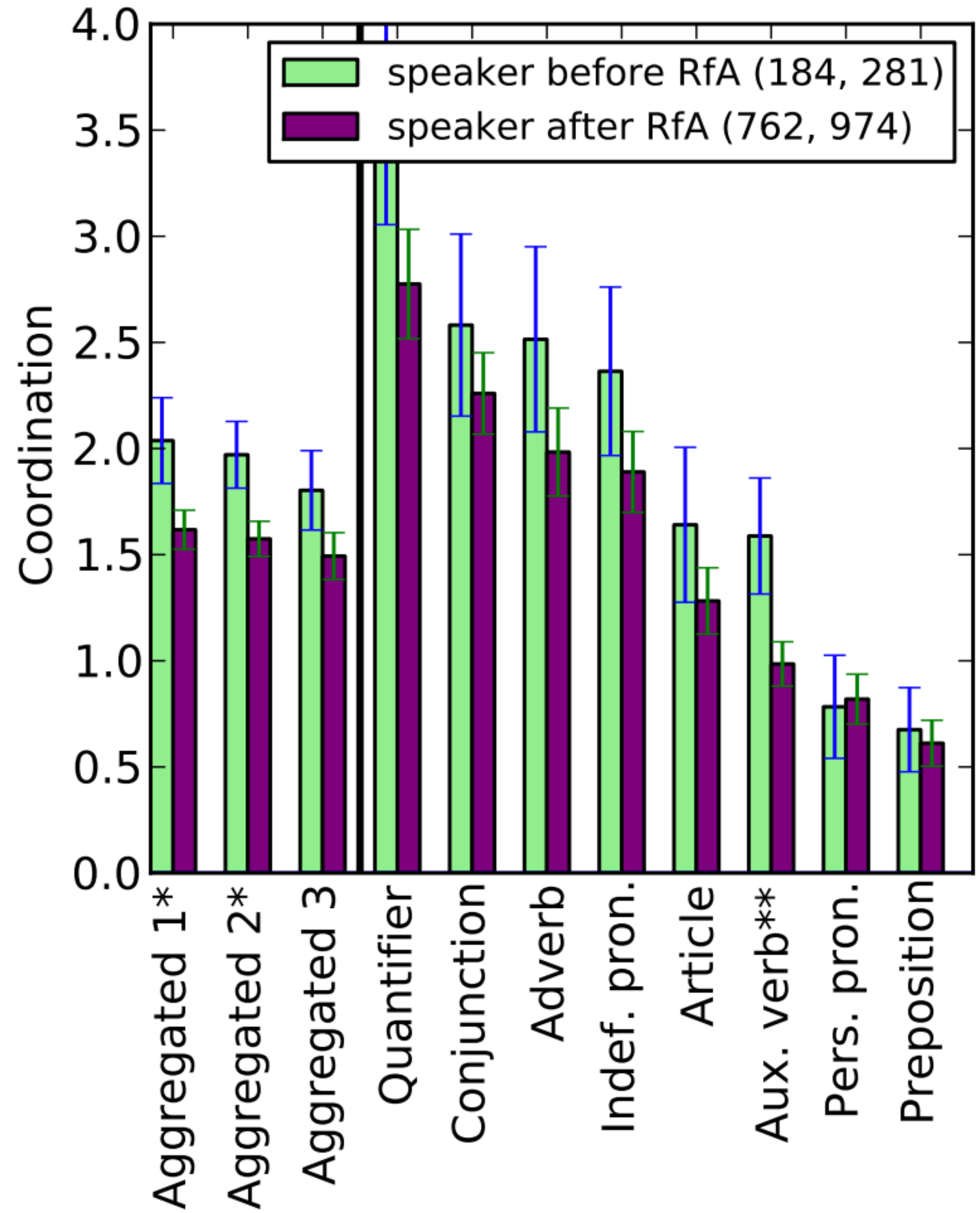




(Danescu-Niculescu-Mizil et al., 2012)

Accommodation





(Danescu-Niculescu-Mizil et al., 2012)

Accommodation

Feature 1

Feature 2

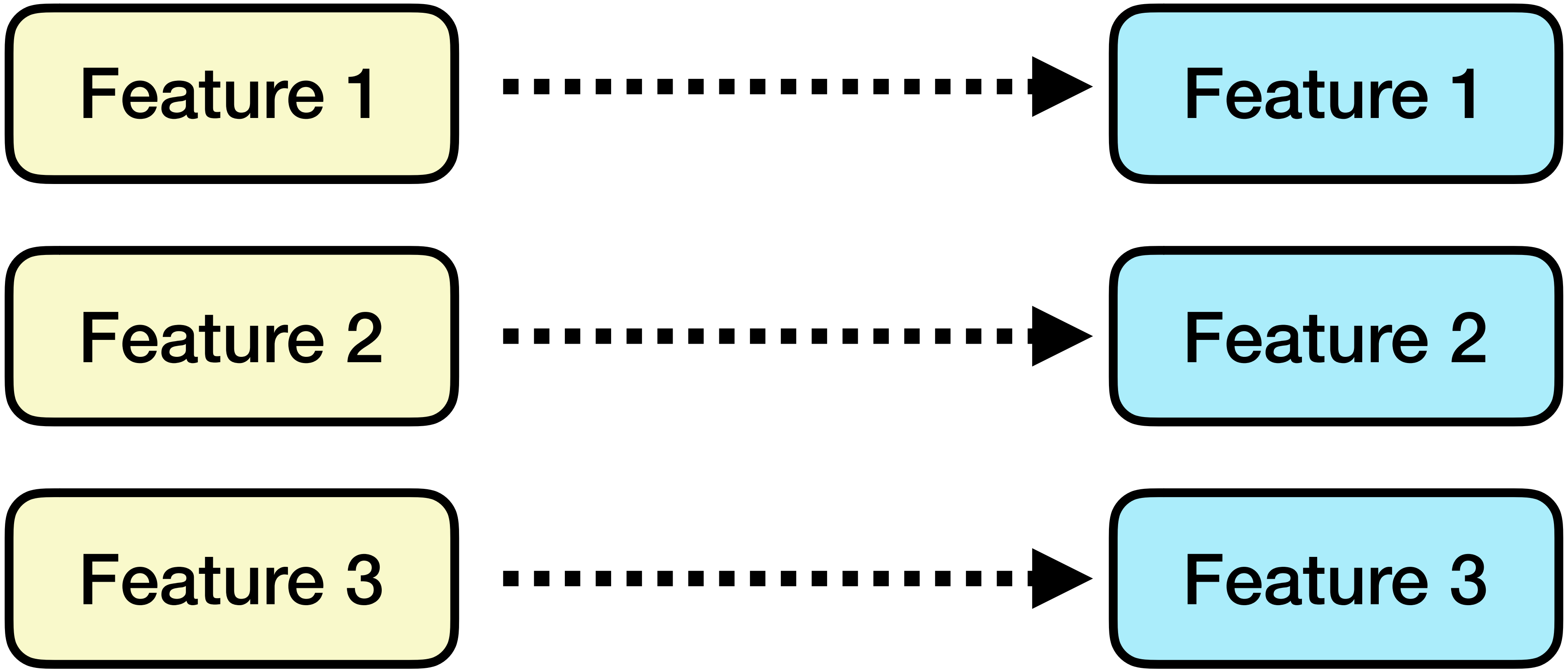
Feature 3

Feature 1

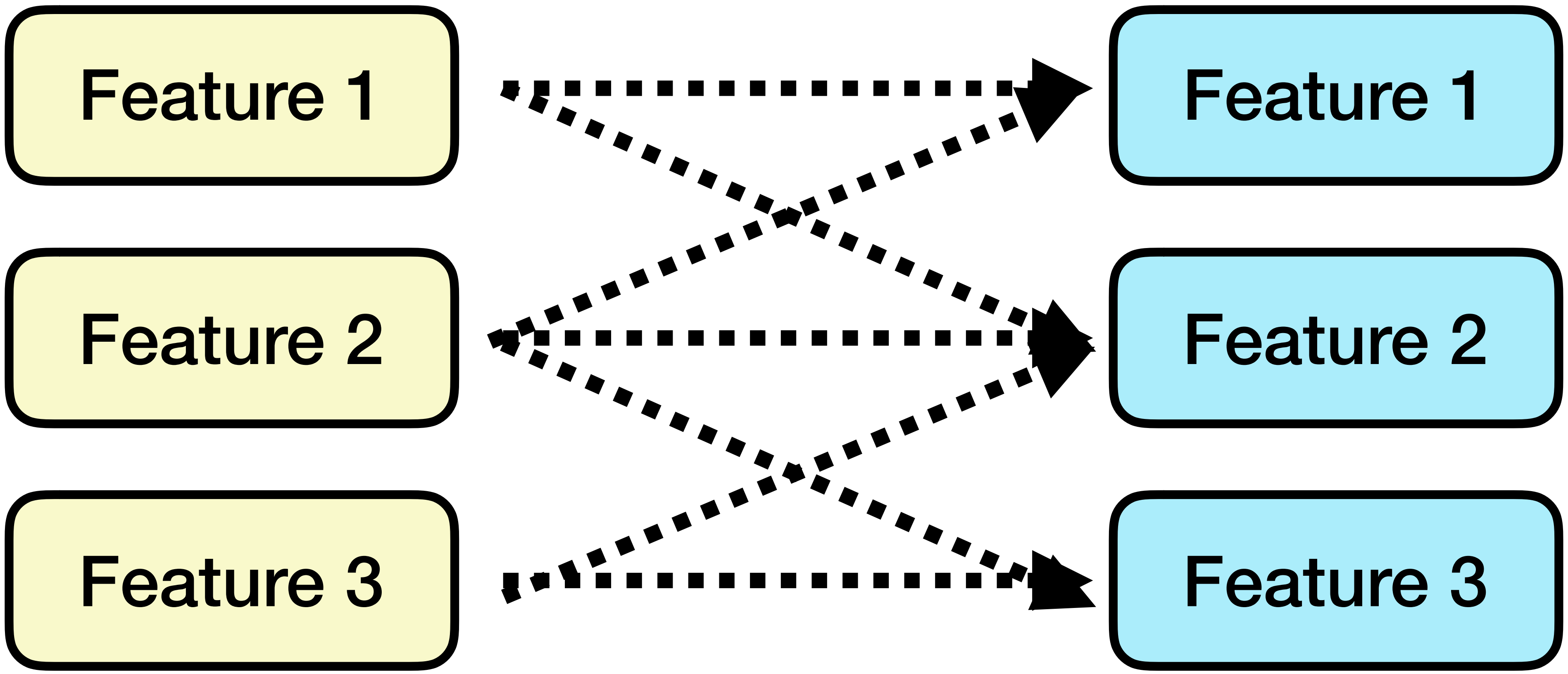
Feature 2

Feature 3

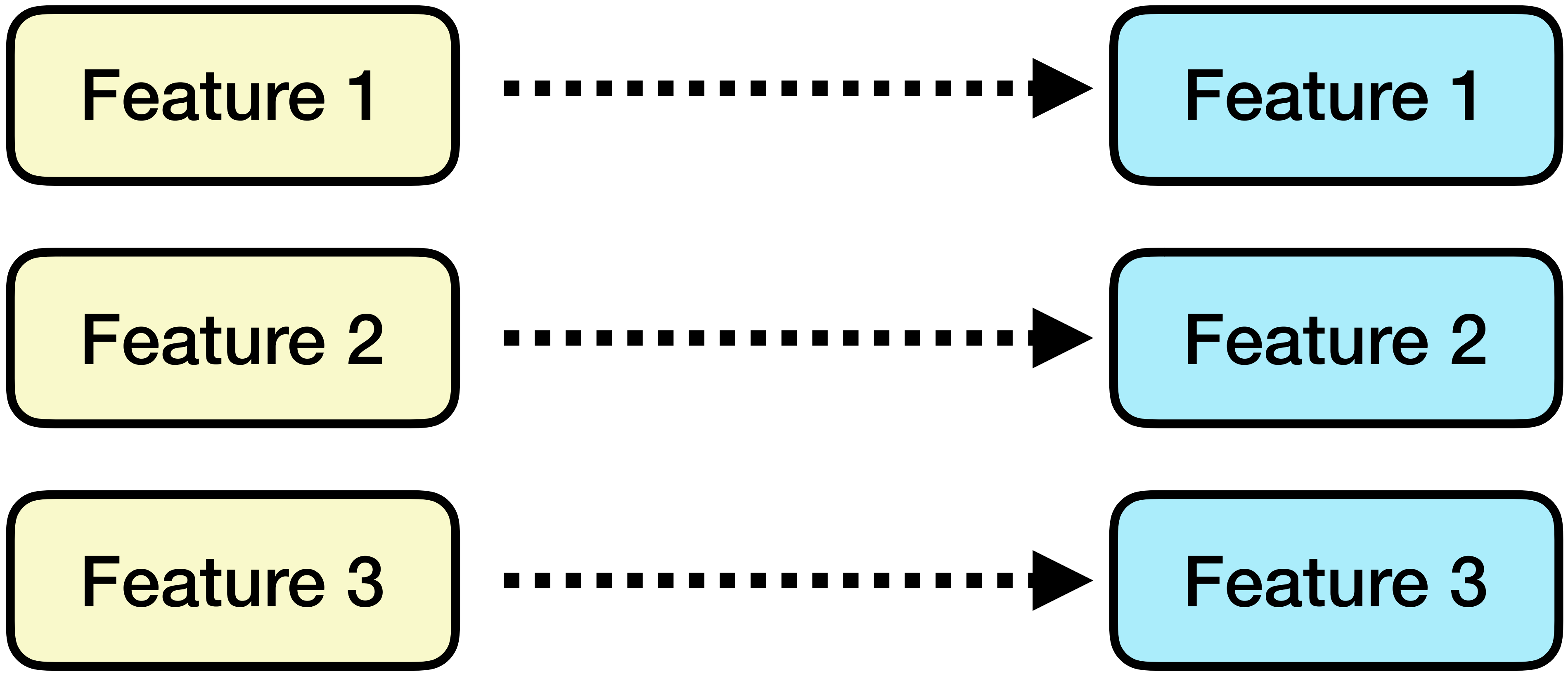
Accommodation



Accommodation



Accommodation

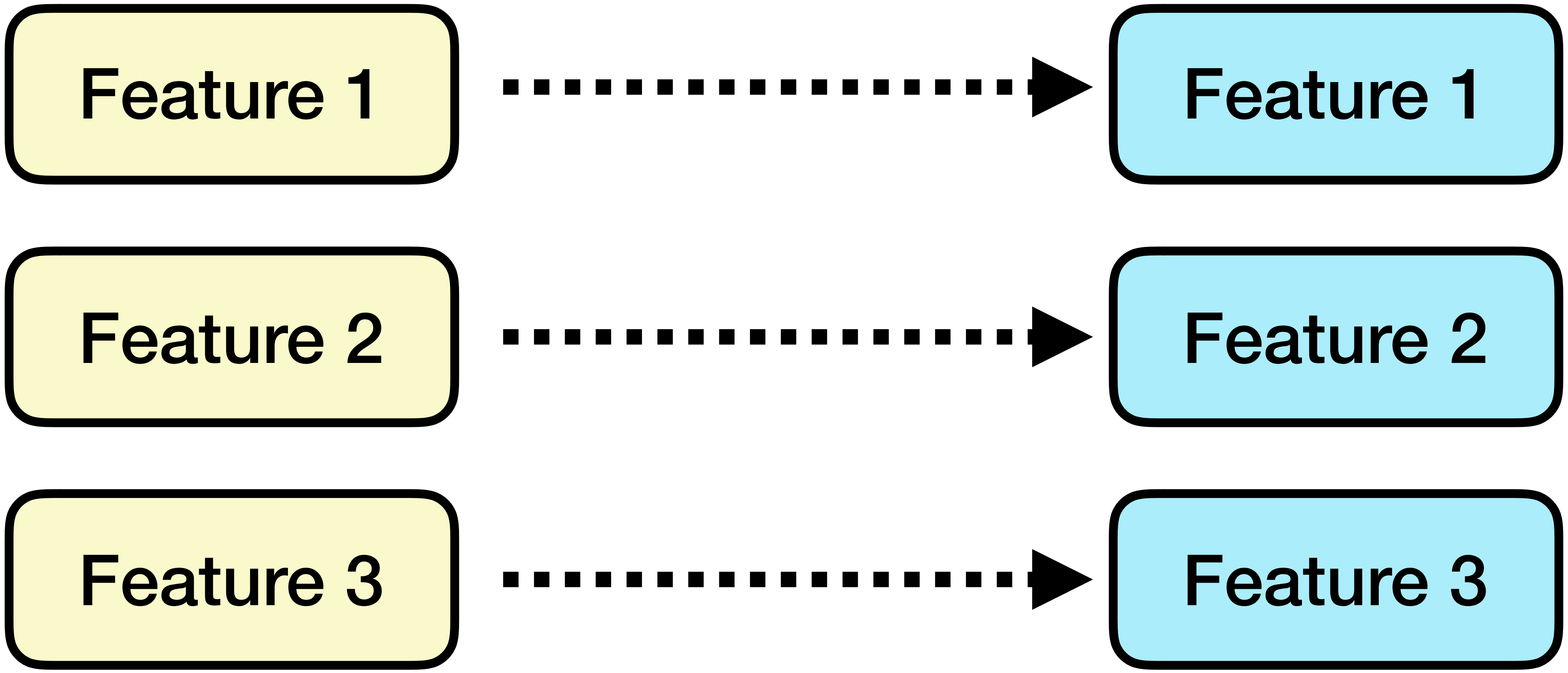


Feature Sets

Function word categories (pronouns, quantifiers, etc.)

(Ireland et al., 2011; DNM et al., 2012)

Accommodation

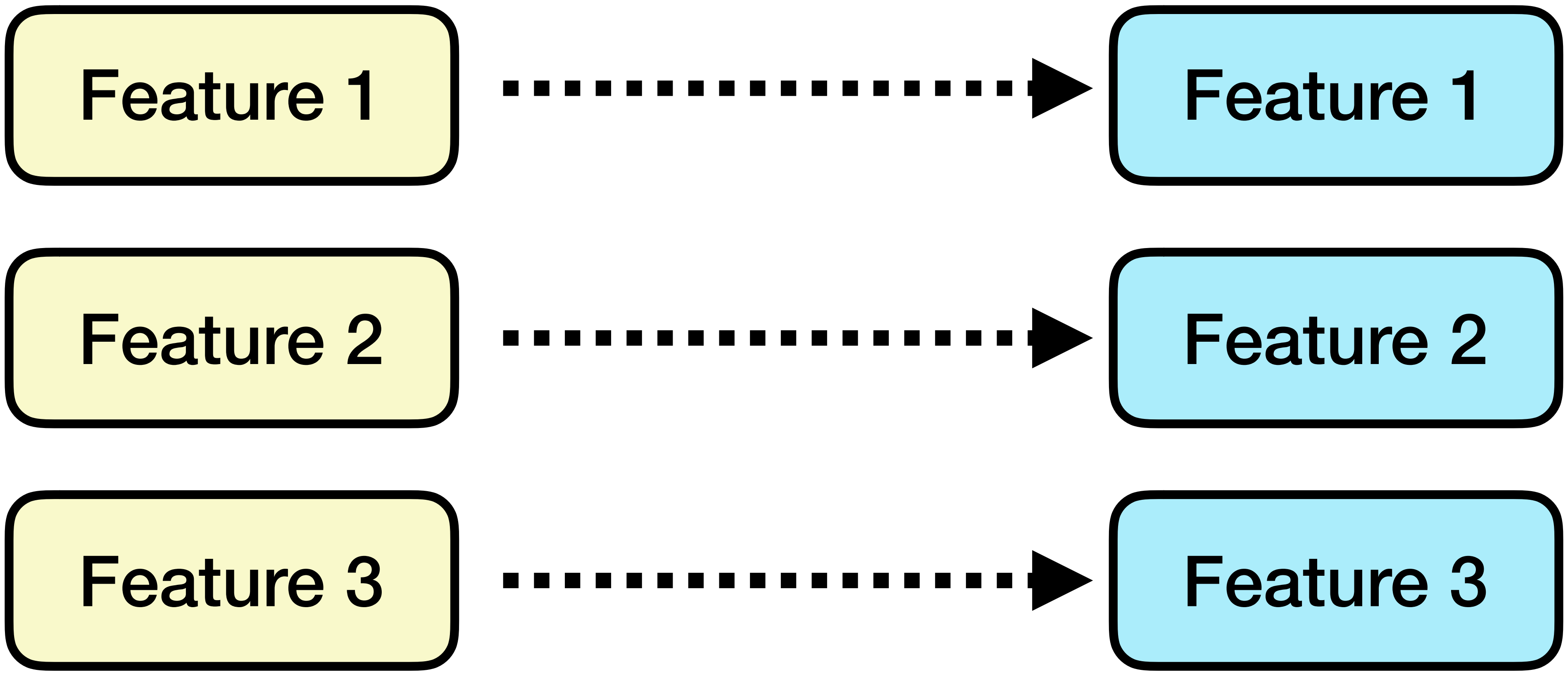


Feature Sets

Function word categories (pronouns, quantifiers, etc.)
(Ireland et al., 2011; DNM et al., 2012)

Latent Semantic Similarity - PCA-transformed unigrams
(Hearst, 1997; Babcock, Ta & Ickes, 2014)

Accommodation



Feature Sets

Function word categories (pronouns, quantifiers, etc.)
(Ireland et al., 2011; DNM et al., 2012)

Latent Semantic Similarity - PCA-transformed unigrams
(Hearst, 1997; Babcock, Ta & Ickes, 2014)

All 64 LIWC dictionaries
(Srivastava et al., 2018)

Accommodation

Tools

Accommodation

Tools

Linguistic Style Matching (Ireland et al., 2011)

Total counts across convos (too simple)

Danescu-Niculescu-Mizil et al (2012)

- One feature at a time, and aggregated
- Separate into turns (“adjacency pairs”)
- Adjust for baseline usage

WHAM (Goodman & Frank, 2016)

Many texts per person/topic

-Adjust for person-level usage

Is accommodation intentional/stylistic/unconscious?

Embeddings Applications

Supervised Learning

Embeddings as extracted features

Similarity

Data exploration
- which document is most like this one?

As an extracted feature
- imitation, novelty, topic switching

Embeddings Applications

Supervised Learning

Embeddings as extracted features

Similarity

Data exploration
- which document is most like this one?

As an extracted feature
- imitation, novelty, topic switching

Distributed dictionaries
- which document is most similar to these words?

Distributed Dictionaries

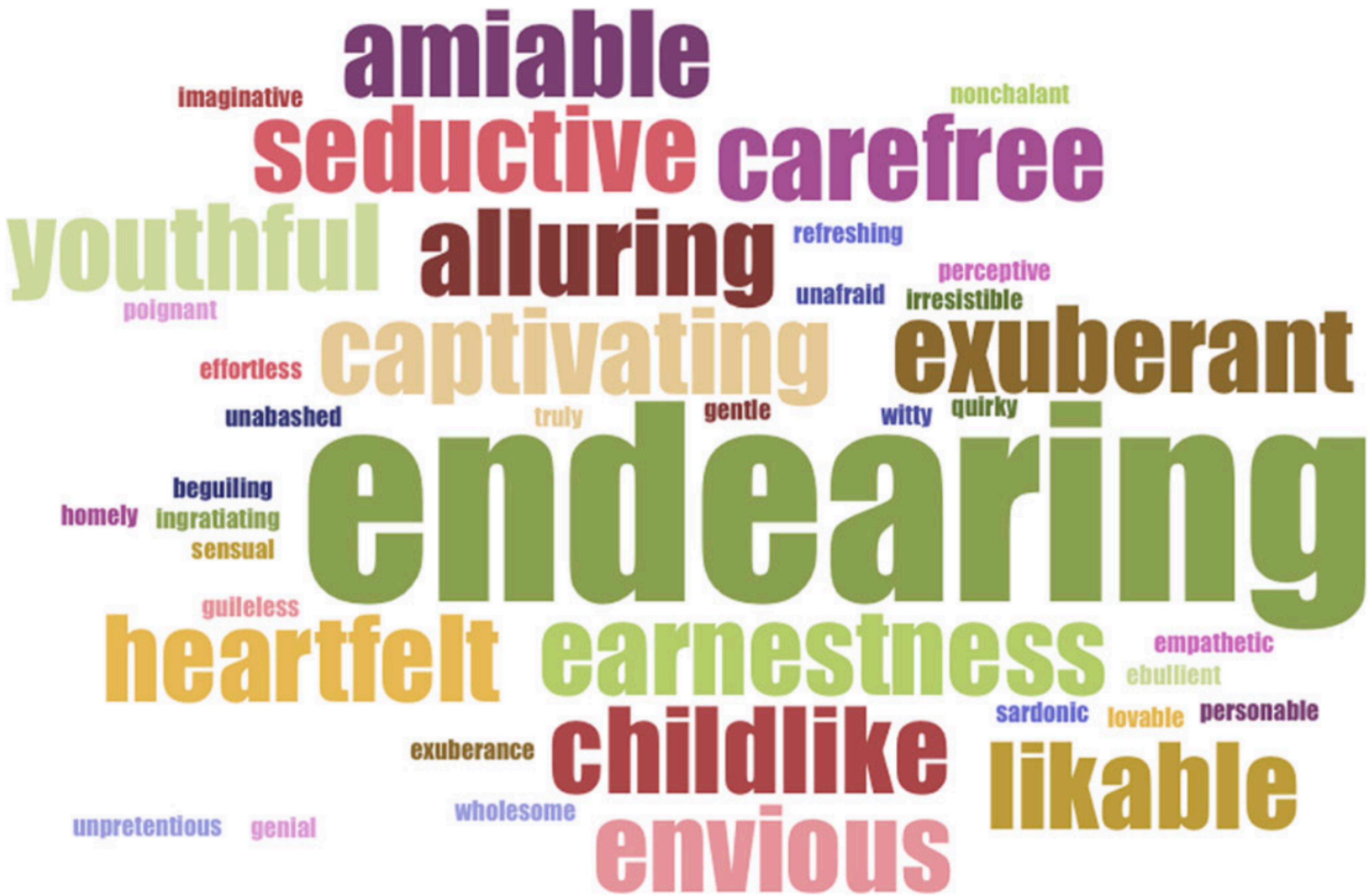


Fig. 4 Nearest neighbors of the LIWC positive emotions dictionary

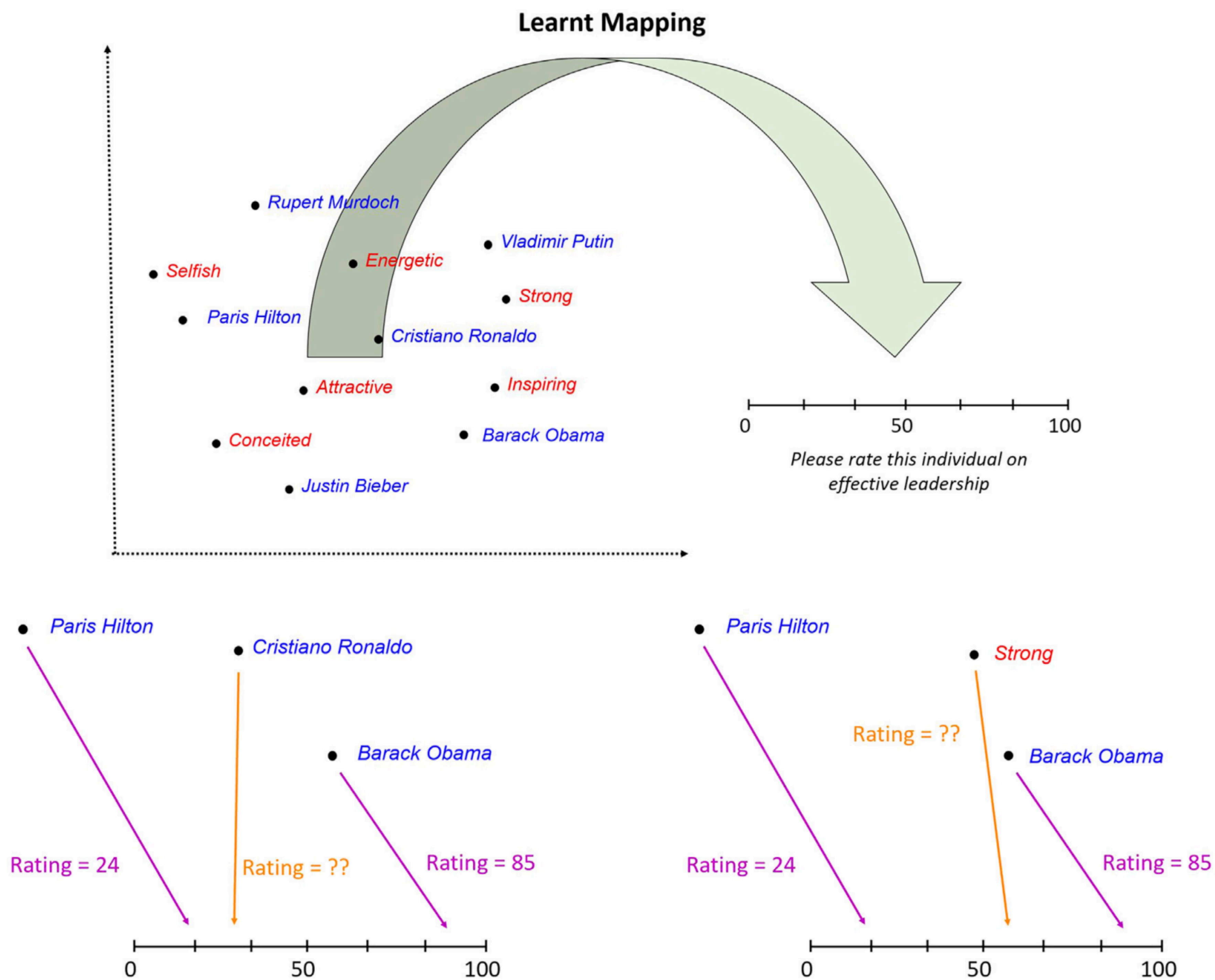
(Garten et al., 2018)

Distributed Dictionaries

Fig. 3 Nearest neighbors of the word "respect"

(Garten et al., 2018)

Distributed Dictionaries



(Bhatia et al., 2021)

Embedding Models

Advantages

Captures common synonymy

Embedding Models

Advantages

Captures common synonymy

Excellent for estimation

Embedding Models

Advantages

Captures common synonymy

Excellent for estimation

Natural for similarity-based tasks

Embedding Models

Advantages

Captures common synonymy

Excellent for estimation

Natural for similarity-based tasks

Limitations

Hard to interpret results

Computationally intensive