Using features inspired by psychology and linguistics to improve automatic detection of subtle information in text

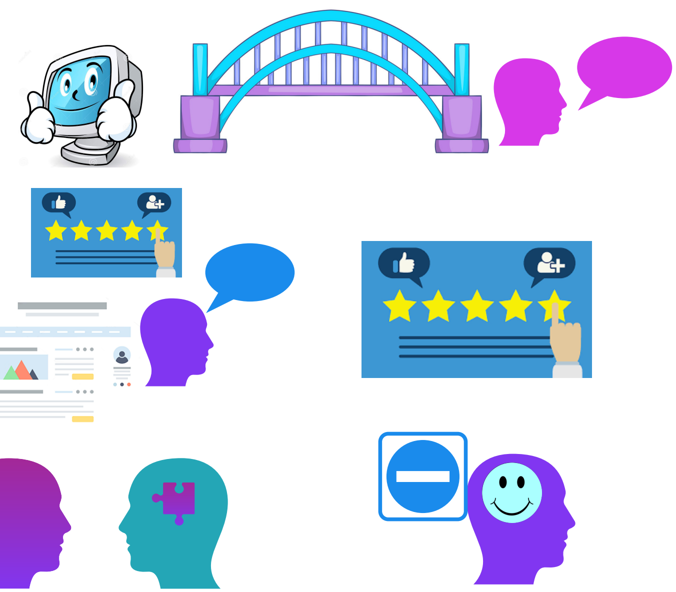
Computation of Language Laboratory

UC Irvine

CCR La Jolla

September 5, 2019

Lisa S. Pearl
Professor
Department of Language Science
Department of Cognitive Sciences
SSPB 2219
University of California, Irvine
lpearl@uci.edu



Natural language understanding:

Extract the information that humans do from natural language

"C'mon — don't you think this is awesome?"

Natural language understanding:
Extract the information that humans do from natural language

"C'mon — don't you think this is awesome?"

Contraction:
"Come on"

Natural language understanding:

Extract the information that humans do from natural language

Contraction:
"Do not"

"C'mon — don't you think this is awesome?"

Contraction:
"Come on"

Natural language understanding:
Extract the information that humans do from natural language

Exclamation

"C'mon — don't you think this is awesome?"

Natural language understanding:

Extract the information that humans do from natural language

Exclamation

Yes/no question

"C'mon — don't you think this is awesome?"

Natural language understanding:
Extract the information that humans do from natural language

Exclamation

Yes/no question

"C'mon — don't you think this is awesome?"

Natural language understanding:
Extract the information that humans do from natural language

"C'mon — don't you think this is awesome?"

addressee

mental state

Natural language understanding:

Extract the information that humans do from natural language

“C’mon — don’t you think this is awesome?”

addressee

good++++

mental state

Natural language understanding:

Extract the information that humans do from natural language

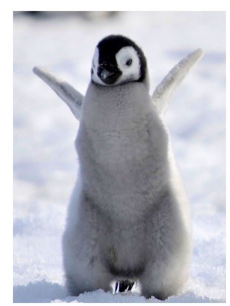
"C'mon — don't you think this is awesome?"

something salient in the discourse context

Natural language understanding:
Extract the information that humans do from natural language

"C'mon — don't you think this is awesome?"

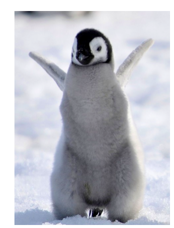
something salient in
the discourse context



Natural language understanding:
Extract the information that humans do from natural language

“C’mon — don’t you think this is awesome?”

core information

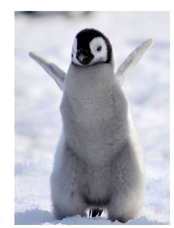


Natural language understanding:
Extract the information that humans do from natural language

But there’s more subtle information, too.

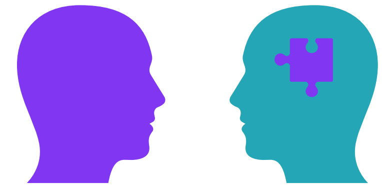
“C’mon — don’t you think this is awesome?”

core information



Natural language understanding:
Extract the information that humans do from natural language

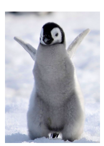
The speaker likely has a persuasive intention.



"C'mon — don't you think this is awesome?"

core information

more subtle information



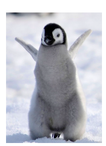
Natural language understanding:

Extract the information that humans do from natural language

If the speaker actually doesn't like penguins, he could be intending to ingratiate himself with the addressee (using deception).

"C'mon — don't you think this is awesome?"

core information

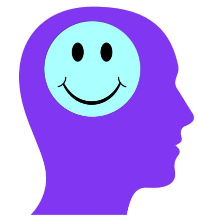


more subtle information

Natural language understanding:

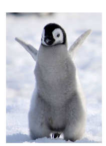
Extract the information that humans do from natural language

At face value, the speaker seems to have a good feeling about penguins (positive sentiment).



"C'mon — don't you think this is awesome?"

core information



more subtle information

intentions

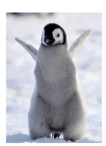
Natural language understanding:

Extract the information that humans do from natural language

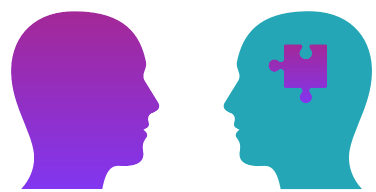
The casual style of speaking suggests familiarity with the addressee, and may indicate something about the speaker’s identity.

“C’mon — don’t you think this is awesome?”

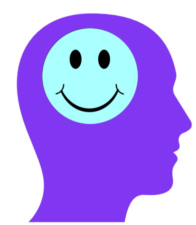
core information



more subtle information



intentions



emotions/attitudes

Natural language understanding:

Extract the information that humans do from natural language

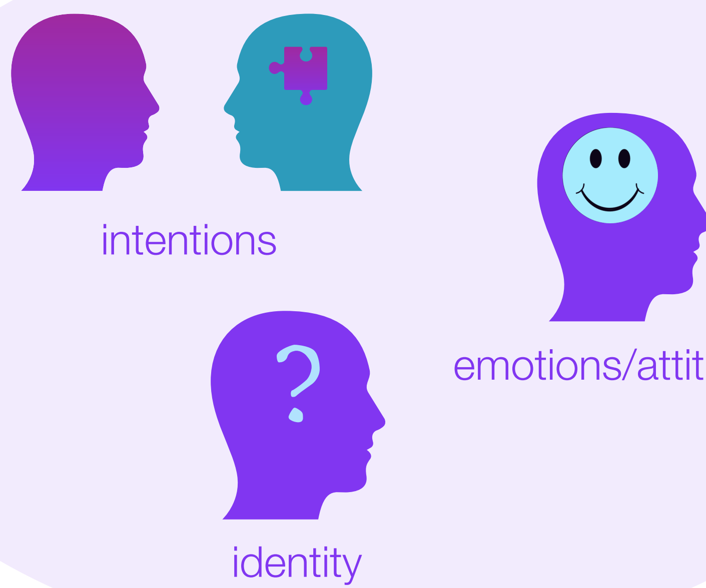
Our focus today: This more subtle information.

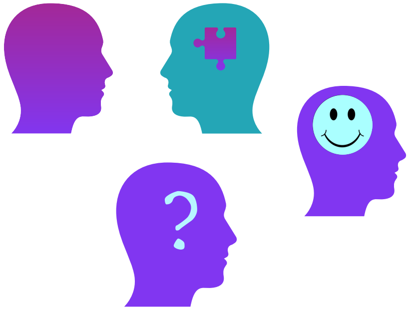
Why? Because it's currently harder to automatically extract.

"C'mon — don't you think this is awesome?"

core information

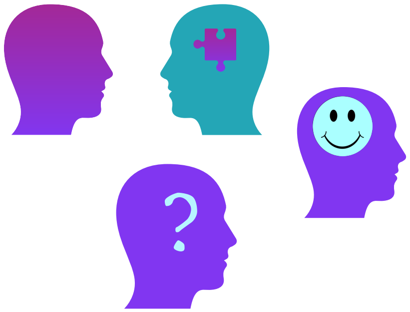
more subtle information





Extracting more subtle information from natural language

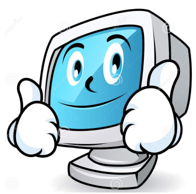
Fun fact: For people who study this, there's been an interesting divide…



Extracting more subtle information from natural language

Fun fact: For people who study this, there's been an interesting divide…

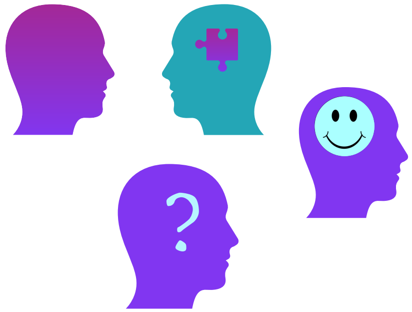
computation



mathematical knowledge

computational tools

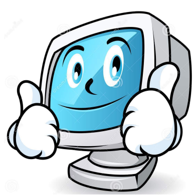
machine learning techniques



Extracting more subtle information from natural language

Fun fact: For people who study this, there's been an interesting divide…

computation



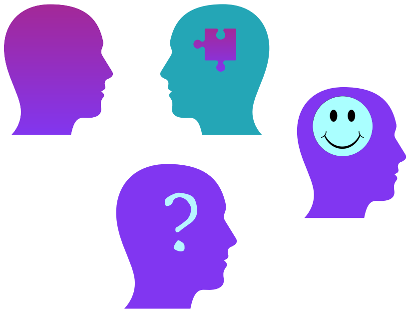
mathematical knowledge

computational tools

machine learning techniques

psychology & linguistics

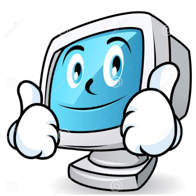
precise psychological and linguistic theoretical constructs that are hard to automatically identify

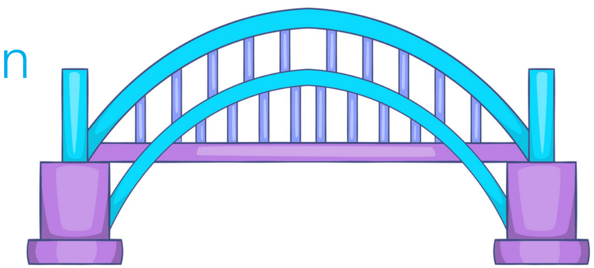


Extracting more subtle information from natural language

What I’ve been trying to do:
bridge the divide and see what we can get out of it

computation





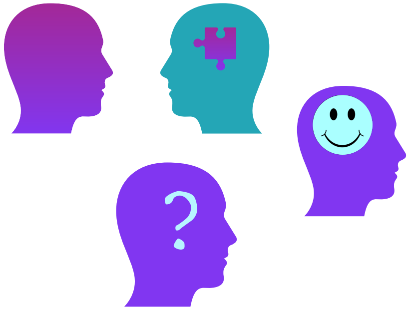
psychology & linguistics

mathematical knowledge

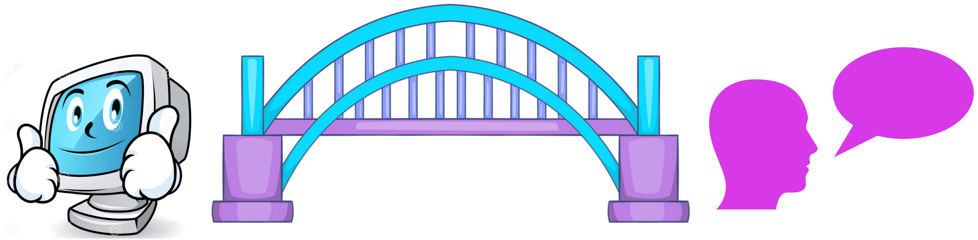
computational tools

machine learning techniques

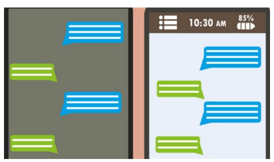
precise psychological and linguistic theoretical constructs that are hard to automatically identify



Extracting more subtle information from natural language



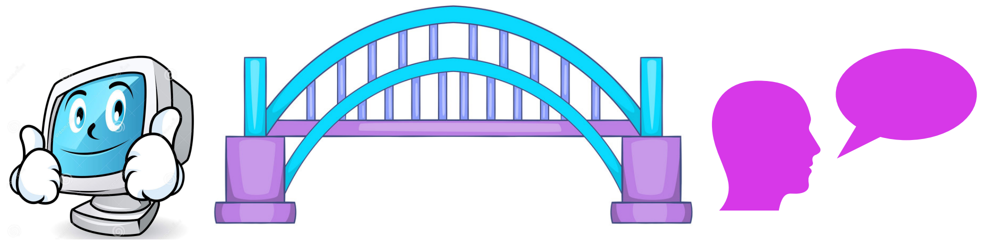
Some previous work, focusing on language text alone:



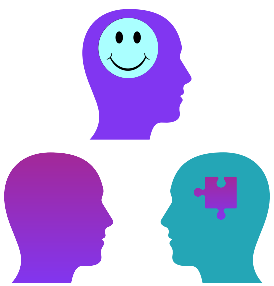
electronic
(more conversational)

written text

Extracting more subtle information from natural language

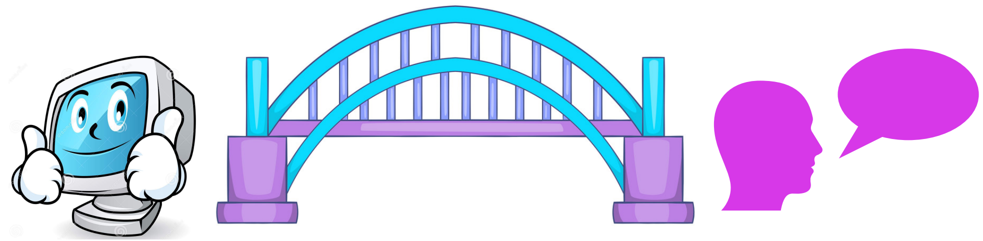


Some previous work, focusing on language text alone:

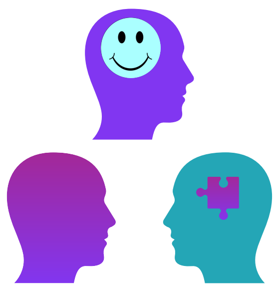


Pearl & Enverga 2015: Detecting emotions, attitudes, and intentions in short messages

Extracting more subtle information from natural language



Some previous work, focusing on language text alone:

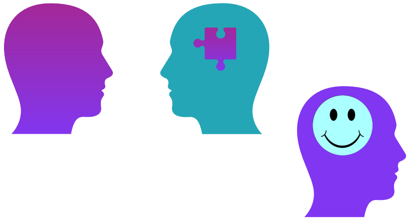


Pearl & Enverga 2015: Detecting emotions, attitudes, and intentions in short messages

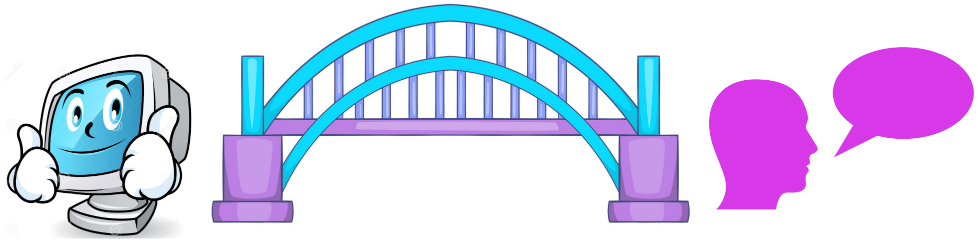
Much better accuracy when using "deeper" n-grams that were semantically & syntactically more abstract

the+best
the+brightest
the+most+fantastic
the+most+fun

→ *the+POSITIVE-ADJECTIVE-IN-THE-SUPERLATIVE*

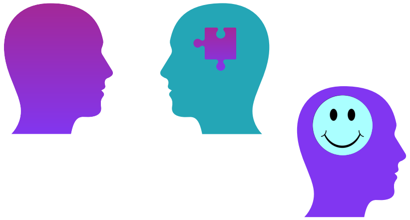


Extracting more subtle information from natural language

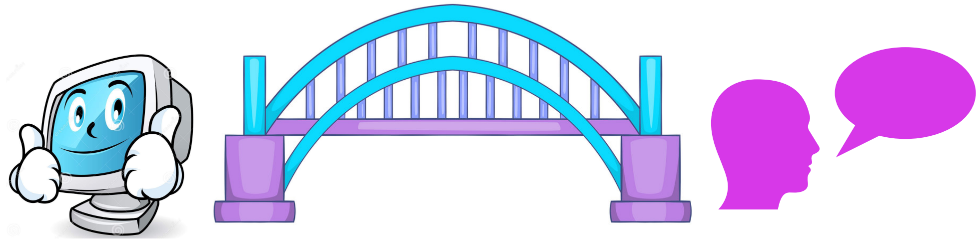


Some previous work, focusing on language text alone:

Pearl, Lu, & Haghighi 2016: Authorship in epistolary novels — can one person really write in the style of multiple characters?



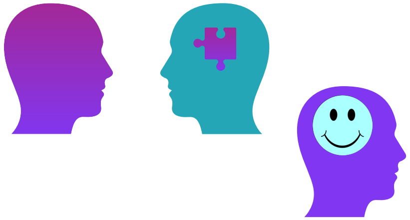
Extracting more subtle information from natural language



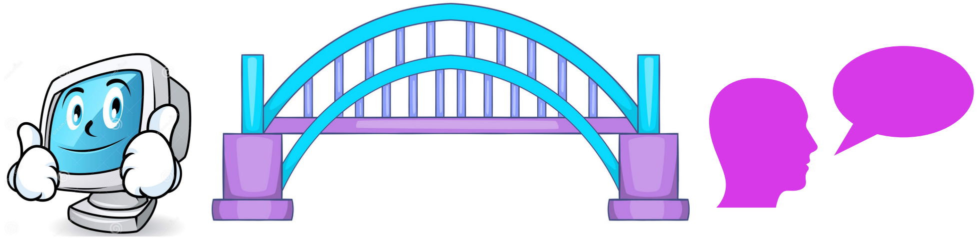
Some previous work, focusing on language text alone:

Pearl, Lu, & Haghighi 2016: Authorship in epistolary novels — can one person really write in the style of multiple characters?

Answer: Yes and no.
The features the author manipulated (which did create several fairly distinct characters) weren’t the ones that signified his own style. His own style features were still present.



Extracting more subtle information from natural language

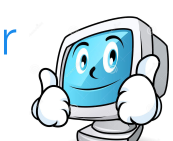


Some previous work, focusing on language text alone:

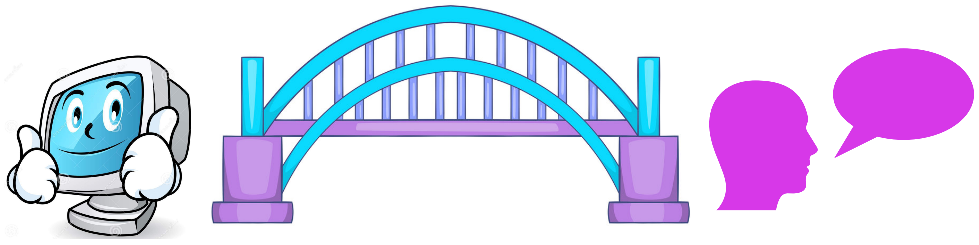
Pearl, Lu, & Haghighi 2016: Authorship in epistolary novels — can one person really write in the style of multiple characters?

Answer: Yes and no.
The features the author manipulated (which did create several fairly distinct characters) weren't the ones that signified his own style. His own style features were still present.

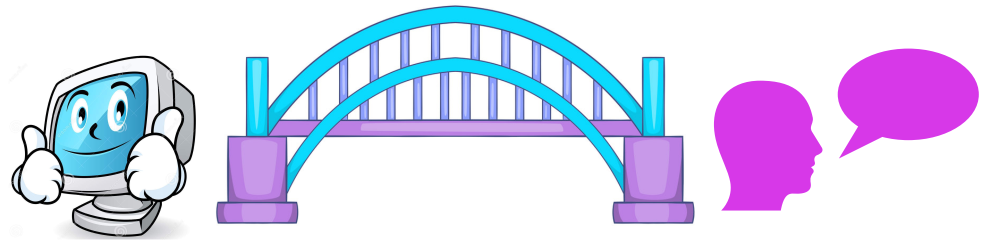
How: Using syntactically-richer and semantically-tailored features with an SMLR classifier

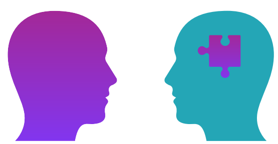


Today's plan



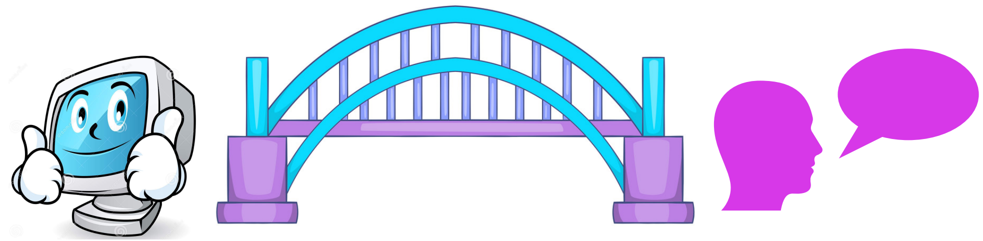
Today's plan

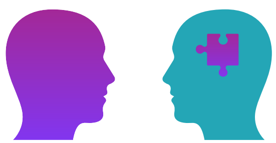




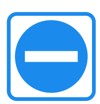
Deception detection
across content domains

Today's plan

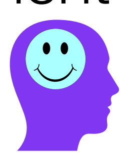




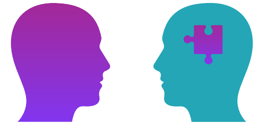
Deception detection
across content domains



Negation handling in
sentiment analysis



The detection problem

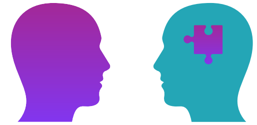


Which of these is a fake review?

from the Deceptive Opinion Spam corpus (Ott et al. 2011, 2013)

Vogler & Pearl 2019

The detection problem



Which of these is a fake review?

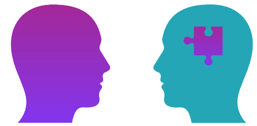
from the Deceptive Opinion Spam corpus (Ott et al. 2011, 2013)

#1

I only stayed out with my boyfriend for one night, however enjoyed my stay. The staff was friendly, the room was nice and clean, the hallways and ballrooms etc were elegant. Room service was quick and had good options to choose from that actually tasted great. The staff was able to extend our check out time for an extra 1-2 hours without an extra charge to the room. Great location too! Walking distance from the Art Museum, Millennium Park, Grant Park (right across the street) and a quick cab ride to McCormick Place. If I were in the city again I would love to stay there again.

Vogler & Pearl 2019

The detection problem



Which of these is a fake review?

from the Deceptive Opinion Spam corpus (Ott et al. 2011, 2013)

#1

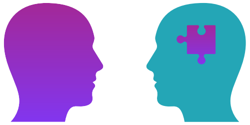
I only stayed out with my boyfriend for one night, however enjoyed my stay. The staff was friendly, the room was nice and clean, the hallways and ballrooms etc were elegant. Room service was quick and had good options to choose from that actually tasted great. The staff was able to extend our check out time for an extra 1-2 hours without an extra charge to the room. Great location too! Walking distance from the Art Museum, Millennium Park, Grant Park (right across the street) and a quick cab ride to McCormick Place. If I were in the city again I would love to stay there again.

#2

The Hilton in Chicago was awesome. The room was very clean and the hotel staff was very professional. One of the features I liked, was that in my room the internet access was wire and wireless, considering my laptop is not wireless, it help me out alot. Food was very good, quality was great. There was also a flat screen in my room...awesome. The hotel itself is locaated in the middle of alot of resturants with fin dinning. I also enjoyed the gym very much. Overall, I enjoyed myself, and I will stay again at the Hilton when I return to Chicago.

Vogler & Pearl 2019

The detection problem



Which of these is a fake review?

from the Deceptive Opinion Spam corpus (Ott et al. 2011, 2013)

#1

I only stayed out with my boyfriend for one night, however enjoyed my stay. The staff was friendly, the room was nice and clean, the hallways and ballrooms etc were elegant. Room service was quick and had good options to choose from that actually tasted great. The staff was able to extend our check out time for an extra 1-2 hours without an extra charge to the room. Great location too! Walking distance from the Art Museum, Millennium Park, Grant Park (right across the street) and a quick cab ride to McCormick Place. If I were in the city again I would love to stay there again.

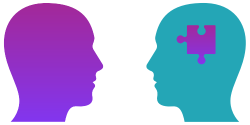
#2

The Hilton in Chicago was awesome. The room was very clean and the hotel staff was very professional. One of the features I liked, w[illegible]hat in my room the internet access was w[illegible]eless, considering my laptop is n[illegible]e out alot. Food was very goo[illegible] great. There was also a flat screen in m[illegible]om...awesome. The hotel itself is locaated in the middle of alot of resturants with fin dinning. I also enjoyed the gym very much. Overall, I enjoyed myself, and I will stay again at the Hilton when I return to Chicago.

FAKE

Vogler & Pearl 2019

The detection problem

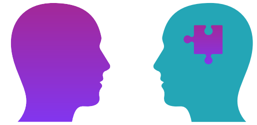


Which of these is a fake opinion?

from the Essays corpus (Mihalcea & Strapparava 2009)

Vogler & Pearl 2019

The detection problem



Which of these is a fake opinion?

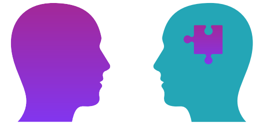
from the Essays corpus (Mihalcea & Strapparava 2009)

#1

Death penalty is very important as a deterrent against crime. We live in a society, not as individuals. This imposes some restrictions on our actions. If a person doesn’t adhere to these restrictions, he or she forfeits her life. Why should taxpayers’ money be spent on feeding murderers?

Vogler & Pearl 2019

The detection problem



Which of these is a fake opinion?

from the Essays corpus (Mihalcea & Strapparava 2009)

#1

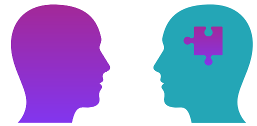
Death penalty is very important as a deterrent against crime. We live in a society, not as individuals. This imposes some restrictions on our actions. If a person doesn't adhere to these restrictions, he or she forfeits her life. Why should taxpayers' money be spent on feeding murderers?

#2

I stand against death penalty. It is pompous of anyone to think that they have the right to take life. No court of law can eliminate all possibilities of doubt. Also, some circumstances may have pushed a person to commit a crime that would otherwise merit severe punishment.

Vogler & Pearl 2019

The detection problem



Which of these is a fake opinion?

from the Essays corpus (Mihalcea & Strapparava 2009)

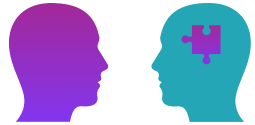
#1

Death penalty is very importa[illegible] as a deterrent against crime. We live i[illegible]ety, not as individuals. This imposes s[illegible]n our actions. If a person does[illegible]ese restrictions, he or she forfeits h[illegible]hy should taxpayers' money be spent on feeding murderers?

#2

I stand against death penalty. It is pompous of anyone to think that they have the right to take life. No court of law can eliminate all possibilities of doubt. Also, some circumstances may have pushed a person to commit a crime that would otherwise merit severe punishment.

The detection problem



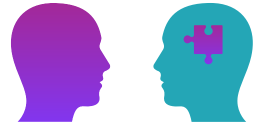
Which of these is a fake interview answer?

(to *Please describe your educational background)*

from the Deceptive Interview corpus (Burgoon et al. 1999)

Vogler & Pearl 2019

The detection problem



Which of these is a fake interview answer?

(to *Please describe your educational background)*

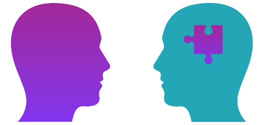
from the Deceptive Interview corpus (Burgoon et al. 1999)

#1

Well, I am a, I completed my masters degree in business administration. And I am hopefully going to be completing one for my doctorate, depending on time and money. In December of 1990. U of A. As I say that depends on money and the family situation. When I have time and money and work allows and everything else. Where did I complete that, I did that in ’87, and I took some time off and went back. Here in Tucson.

Vogler & Pearl 2019

The detection problem



Which of these is a fake interview answer?

(to *Please describe your educational background)*

from the Deceptive Interview corpus (Burgoon et al. 1999)

#1

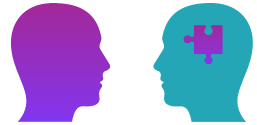
Well, I am a, I completed my masters degree in business administration. And I am hopefully going to be completing one for my doctorate, depending on time and money. In December of 1990. U of A. As I say that depends on money and the family situation. When I have time and money and work allows and everything else. Where did I complete that, I did that in ’87, and I took some time off and went back. Here in Tucson.

#2

I have a bachelors of arts in education. I have an associates degree in accounting and computerized, eh um, bookkeeping and I have an artisans training in crafts. About eighteen years of formal school and about 45 years of practice. Oh yes, very much so. Um, not necessarily, I think a person who wants to be a teacher has to be very much dedicated, now more than ever. And as for accounting, that is just wisdom in these economic times. And I happen to be a creative fidget when it comes to crafts.

Vogler & Pearl 2019

The detection problem



Which of these is a fake interview answer?

(to *Please describe your educational background)*

from the Deceptive Interview corpus (Burgoon et al. 1999)

#1

Well, I am a, I completed my masters degree in business administration. And I [illegible] hopefully going to be completing one for [illegible]te, depending on time and mone[illegible] [illegible]90. U of A. As I say that depen[illegible] [illegible]y and the family situation. When I have tim[illegible] [illegible]d money and work allows and everything else. Where did I complete that, I did that in '87, and I took some time off and went back. Here in Tucson.

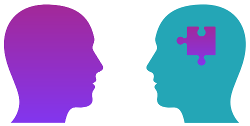
FAKE

#2

I have a bachelors of arts in education. I have an associates degree in accounting and computerized, eh um, bookkeeping and I have an artisans training in crafts. About eighteen years of formal school and about 45 years of practice. Oh yes, very much so. Um, not necessarily, I think a person who wants to be a teacher has to be very much dedicated, now more than ever. And as for accounting, that is just wisdom in these economic times. And I happen to be a creative fidget when it comes to crafts.

Vogler & Pearl 2019

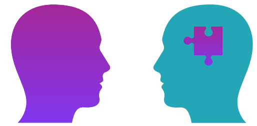
The cross-domain detection problem



We can get reasonable detection performance when we train and test in the same content domain.

Vogler & Pearl 2019

The cross-domain detection problem



We can get reasonable detection performance when we train and test in the same content domain.

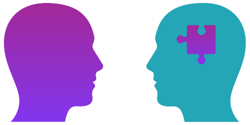
train = test

Feng et al. 2012: **.912** F-score for hotel reviews

Vogler & Pearl 2019

The cross-domain detection problem



We can get reasonable detection performance when we train and test in the same content domain.

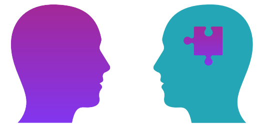
train = test

Feng et al. 2012: **.912** F-score for hotel reviews

Fornaciari & Poesio 2014: **.752** F-score for fake positive book reviews

Vogler & Pearl 2019

The cross-domain detection problem



We can get reasonable detection performance when we train and test in the same content domain.

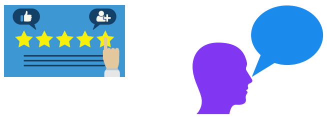
train = test

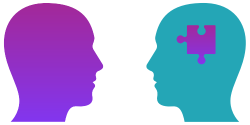
Feng et al. 2012: **.912** F-score for hotel reviews

Fornaciari & Poesio 2014: **.752** F-score for fake positive book reviews

Vogler & Pearl 2019



The cross-domain detection problem



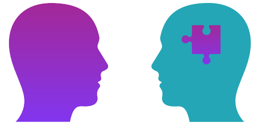
We can get reasonable detection performance when we train and test in the same content domain.

train = test

Feng et al. 2012: **.715-.850** F-score for fake opinions about abortion, the death penalty, and best friends

Vogler & Pearl 2019


The cross-domain detection problem

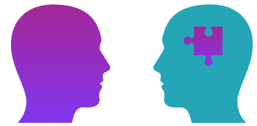


We can get reasonable detection performance when we train and test in the same content domain.

train = test


Feng et al. 2012: **.715-.850** F-score for fake opinions about abortion, the death penalty, and best friends

The cross-domain detection problem

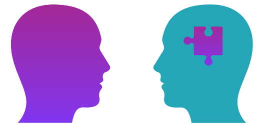


We can get reasonable detection performance when we train and test in the same content domain.

train = test

Yancheva & Rudzicz 2013: **.917** accuracy on children's deceptive interviews about a minor transgression

The cross-domain detection problem



But performance drops a lot when testing on a different content domain.

✓ train = test

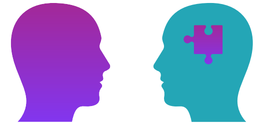
Feng et al. 2012: **.912** F-score for hotel reviews

train ≠ test

Ott et al. 2013: **.703-.830** for hotel reviews where the valence changed

Vogler & Pearl 2019

The cross-domain detection problem



But performance drops a lot when testing on a different content domain.

train = test

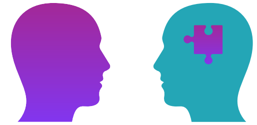
Feng et al. 2012: **.912** F-score for hotel reviews

train ≠ test

Li et al. 2014: **.679-.784** F-score for restaurant and doctor reviews (service reviewed changed)

Vogler & Pearl 2019

The cross-domain detection problem



But performance drops a lot when testing on a different content domain.

train $\neq$ test

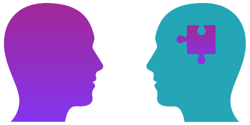
✓ train = test

Feng et al. 2012: **.715-.850** F-score for fake opinions about abortion, the death penalty, and best friends

Feng et al. 2012: **.668-.709** F-score for opinions about different content

Vogler & Pearl 2019

The cross-domain detection problem



But performance drops a lot when testing on a different content domain.

✓ train = test

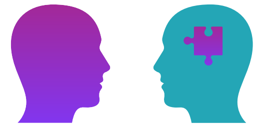
Yancheva & Rudzicz 2013: **.917** accuracy on children's deceptive interviews about a minor transgression

train ≠ test

Fornaciari & Poesio 2011, 2013: **.630** F-score for detecting false court testimony where content is quite variable

Vogler & Pearl 2019

The cross-domain detection problem



The goal: Try to find something that works better at deception detection when we don't have similar content to train on.

train ≠ test

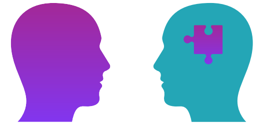
✓ train = test

If we have similar training data, it seems like existing techniques are probably pretty good.

Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test

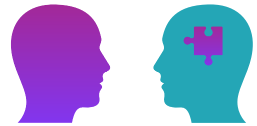


So what features might generalize better across content domains?

Vogler & Pearl 2019

The cross-domain detection problem

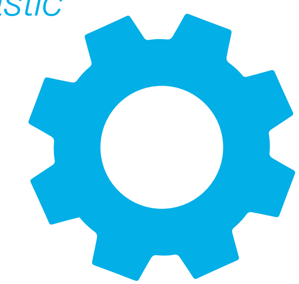
train ≠ test



It turns out that general-purpose linguistic features often used in NLP like word-based n-grams and rules based on syntactic structure have done really well within domain (Ott et al. 2011, Feng et al. 2012).

the+best
the+brightest
the+most+fantastic
the+most+fun

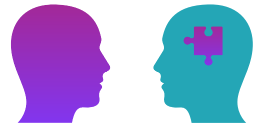
NP^NP —> NNS



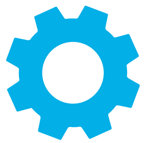
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



general-purpose linguistic features



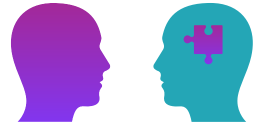
In the psychology of deception, the amount of specific detail is thought to correlate with psychological mechanisms underlying the generation of deceptive language in any domain.

(information manipulation theory: McCornack 1992, information management theory: Burgoon, et al. 1996, Criteria-Based Statement Analysis: Steller and Koehnken 1989, Reality Monitoring: Johnson and Raye 1981)

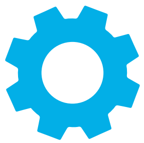
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



general-purpose linguistic features



In the psychology of deception, the amount of specific detail is thought to correlate with psychological mechanisms underlying the generation of deceptive language in any domain.

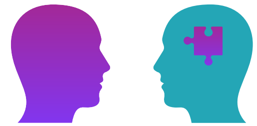
(information manipulation theory: McCornack 1992, information management theory: Burgoon, et al. 1996, Criteria-Based Statement Analysis: Steller and Koehnken 1989, Reality Monitoring: Johnson and Raye 1981)

In particular, less specific detail is correlated with deception.

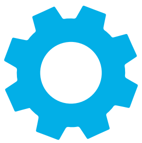
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



general-purpose linguistic features



In the psychology of deception, the amount of specific detail is thought to correlate with psychological mechanisms underlying the generation of deceptive language in any domain.

(information manipulation theory: McCornack 1992, information management theory: Burgoon, et al. 1996, Criteria-Based Statement Analysis: Steller and Koehnken 1989, Reality Monitoring: Johnson and Raye 1981)

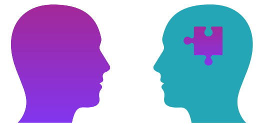
The problem: "specific detail" is a squishy concept that humans can be trained to recognize, but which is hard to automatically identify.

SQUISHY

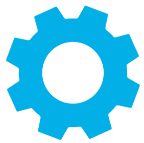
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



general-purpose linguistic features



specific detail

SQUISHY

Let's try…

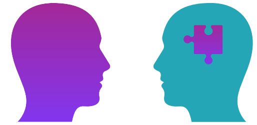
I only stayed out with my boyfriend for one night, however enjoyed my stay. The staff was friendly, the room was nice and clean, the hallways and ballrooms etc were elegant. Room service was quick and had good options to choose from that actually tasted great. The staff was able to extend our check out time for an extra 1-2 hours without an extra charge to the room. Great location too! Walking distance from the Art Museum, Millennium Park, Grant Park (right across the street) and a quick cab ride to McCormick Place. If I were in the city again I would love to stay there again.

Vogler & Pearl 2019

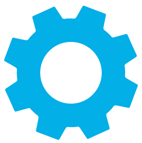


The cross-domain detection problem

train ≠ test



general-purpose linguistic features



specific detail

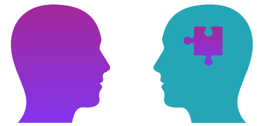
SQUISHY

I only stayed out with my boyfriend for one night, however enjoyed my stay. The staff was friendly, the room was nice and clean, the hallways and ballrooms etc were elegant. Room service was quick and had good options to choose from that actually tasted great. The staff was able to extend our check out time for an extra 1-2 hours without an extra charge to the room. Great location too! Walking distance from the Art Museum, Millennium Park, Grant Park (right across the street) and a quick cab ride to McCormick Place. If I were in the city again I would love to stay there again.

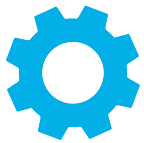
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



general-purpose linguistic features



specific detail

SQUISHY

What we did:

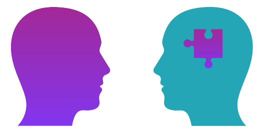
Use human powers to get squishy samples: Look through many text samples and manually identify specific detail examples.

Try to leverage general-purpose linguistic features: Come up with some linguistic structural heuristics that do a “good enough” job of capturing these bits of language.

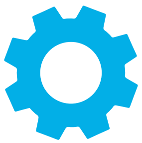
Vogler & Pearl 2019

The cross-domain detection problem

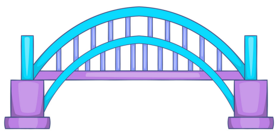
train ≠ test



general-purpose linguistic features



specific detail



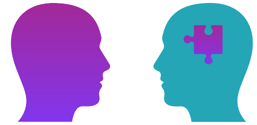
The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

This place is a haven of cool, uncluttered comfort in one of the greatest cities in North America, just two minutes from the nearest airport shuttle.

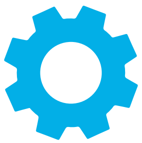
Vogler & Pearl 2019

train ≠ test

The cross-domain detection problem

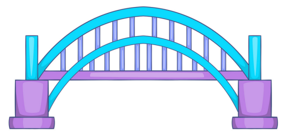


general-purpose linguistic features



specific detail

SQUISHY



The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

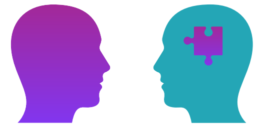
PP modifiers: # and length

This place is a haven ***of*** *cool, uncluttered comfort in one* ***of*** *the greatest cities* ***in*** *North America, just two minutes* ***from*** *the nearest airport shuttle.*

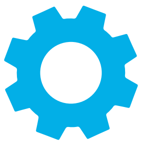
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test

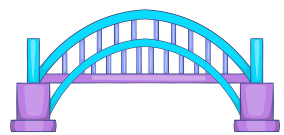


general-purpose linguistic features



specific detail

SQUISHY



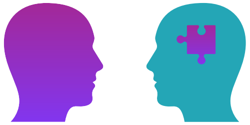
The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

AdjP modifiers: # and length

This place is a haven of ***cool, uncluttered*** *comfort in one of the* ***greatest*** *cities in North America, just two minutes from the* ***nearest*** *airport shuttle.*

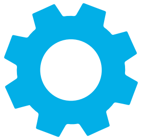
Vogler & Pearl 2019

The cross-domain detection problem



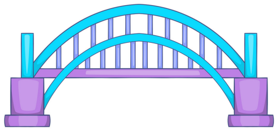
train ≠ test

general-purpose linguistic features



specific detail

SQUISHY



The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

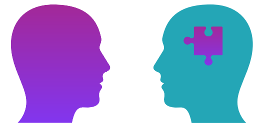
Numbers

This place is a haven of cool, uncluttered comfort in ***one*** *of the greatest cities in North America, just* ***two*** *minutes from the nearest airport shuttle.*

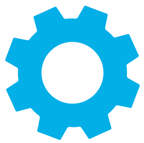
Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test

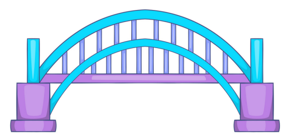


general-purpose linguistic features



specific detail

SQUISHY



The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

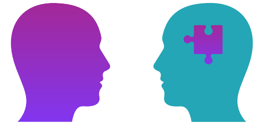
Proper nouns

*This place is a haven of cool, uncluttered comfort in one of the greatest cities in **North America**, just two minutes from the nearest airport shuttle.*

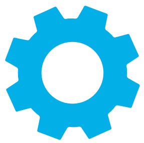
Vogler & Pearl 2019

train ≠ test

The cross-domain detection problem

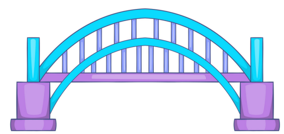


general-purpose linguistic features



specific detail

SQUISHY



The result: Seven linguistically-defined specific detail features that can be incorporated into a classifier

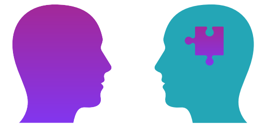
Consecutive nouns

This place is a haven of cool, uncluttered comfort in one of the greatest cities in North America, just two minutes from the nearest ***airport shuttle****.*

Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



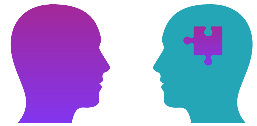
linguistically-defined specific detail features

Sanity check: Most of these appear significantly more frequently in truthful language samples across all three content domains (product reviews, opinions, interview answers).

Vogler & Pearl 2019

The cross-domain detection problem

train ≠ test



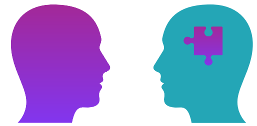
specific
detail

linguistically-defined specific detail features

Sanity check: Most of these appear significantly more frequently in truthful language samples across all three content domains (product reviews, opinions, interview answers).

Vogler & Pearl 2019

The cross-domain detection problem



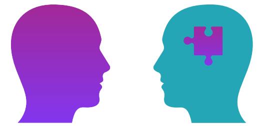
linguistically-defined specific detail features

But will they be effective when cross-domain generalization is required?

train $\neq$ test

Vogler & Pearl 2019

The cross-domain detection problem



linguistically-defined specific detail features

But will they be effective when cross-domain generalization is required?

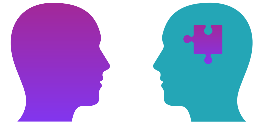
train ≠ test

Let’s find out by incorporating them into an SVM.

Vogler & Pearl 2019

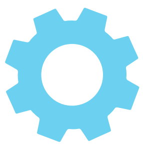
train ≠ test

The cross-domain detection problem



We also want to have something to compare against. So, we'll compare SVMs using only our linguistically-defined specific details against SVMs using

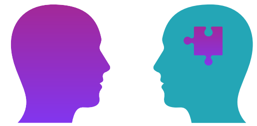
- n-grams (which have done really well in previous within-domain work)

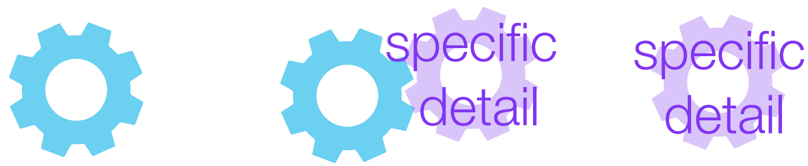


- both n-grams and our linguistically-defined specific details

Vogler & Pearl 2019

Within-domain baselines





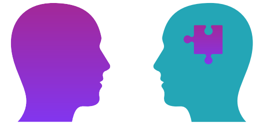
How do they do within-domain? $\boxed{\text{train = test}}$

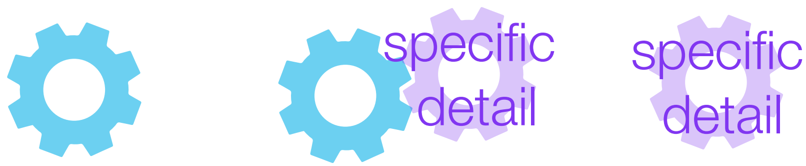
(using 5-fold cross-validation)

Deceptive Opinion Spam
(Ott et al. 2011, 2013)
positive & negative
valence hotel reviews

Vogler & Pearl 2019

Within-domain baselines

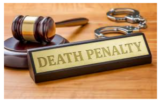




How do they do within-domain? train = test

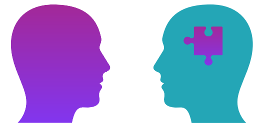
(using 5-fold cross-validation)

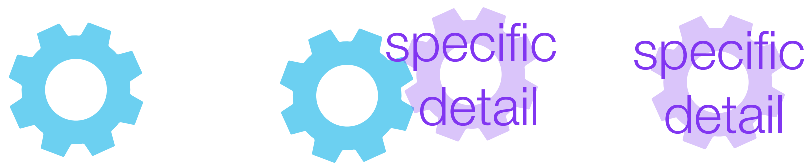
Essays
(Mihalcea & Strapparava 2009)
short opinion essays on
abortion, the death penalty, and
best friends



Vogler & Pearl 2019

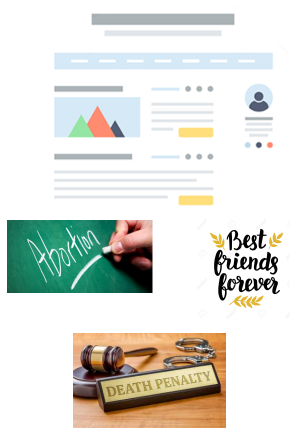
Within-domain baselines





How do they do within-domain? train = test

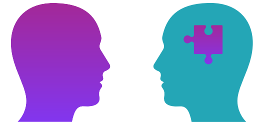
(using 5-fold cross-validation)

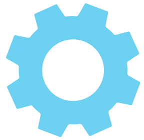


Deceptive Interview
(Burgoon et al. 1999)
real-time answers to 12
job interview questions

Vogler & Pearl 2019

Within-domain baselines



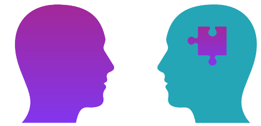


train = test

We'll also separate these out by F-score performance on truthful vs. deceptive samples because different patterns appear (especially once we go cross-domain).

Vogler & Pearl 2019

Within-domain baselines



train = test

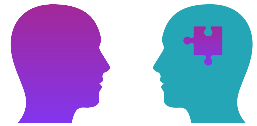
0.87 0.86 0.87 0.87

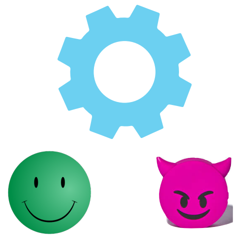
For product reviews, n-grams on their own work pretty well, and adding in linguistically-defined specific details doesn't do much.

There's also **no difference** between performance on truthful vs. deceptive reviews.

Vogler & Pearl 2019

Within-domain baselines





train = test

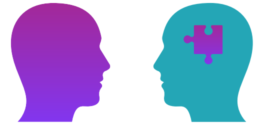
0.87	0.86	0.87	0.87	0.55	0.66

In comparison, specific details on their own don't do so great.

There seems to be worse performance on truthful reviews.

Vogler & Pearl 2019

Within-domain baselines



train = test			specific detail		specific detail	
	0.87	0.86	0.87	0.87	0.55	0.66
	0.71	0.71	0.70	0.70	0.52	0.64

We see the same pattern for the opinion essays, though overall performance for SVMs relying on n-grams drops quite a bit in comparison to product reviews.

Vogler & Pearl 2019

Within-domain baselines

train = test

	0.87	0.86	0.87	0.87	0.55	0.66
	0.71	0.71	0.70	0.70	0.52	0.64
	0.58	0.55	0.56	0.53		

We see the same pattern again for the interview answers, and overall performance for SVMs relying on n-grams drops quite a bit in comparison to opinion essays.

Vogler & Pearl 2019

Within-domain baselines

train = test			specific detail		specific detail	
	0.87	0.86	0.87	0.87	0.55	0.66
	0.71	0.71	0.70	0.70	0.52	0.64
	0.58	0.55	0.56	0.53	0.32	0.61

The SVM relying only specific details also drops performance drastically on truthful answers…though deceptive answer performance remains about the same.

Vogler & Pearl 2019

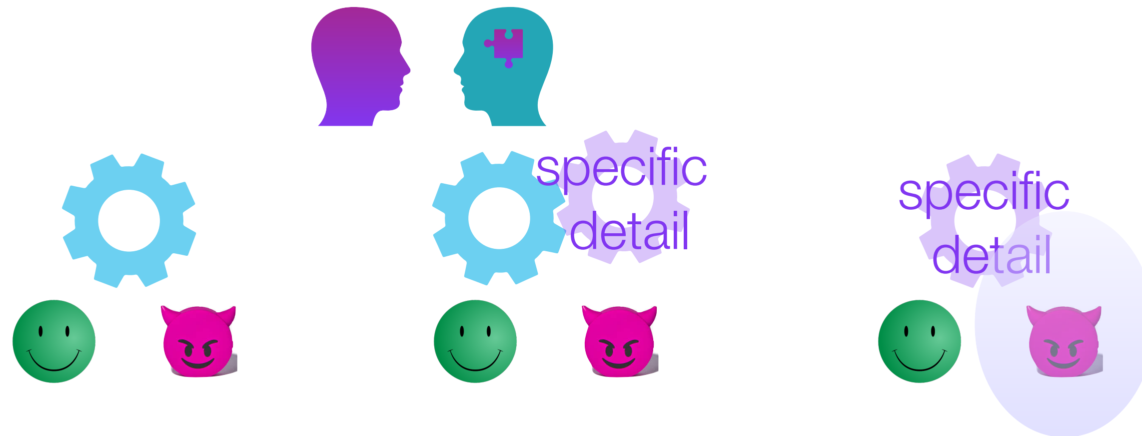
Within-domain baselines

train = test

Takeaway: If you have training data in the same content domain, n-grams will work well enough on their own… though performance does depend on the content domain.

Vogler & Pearl 2019

Within-domain baselines



train = test

Takeaway 2: …but specific details on their own may get you a boost on deceptive performance when the content domain is “hard” for n-grams .

Vogler & Pearl 2019

Narrow-change of content

train ≠ test

But now let's see what happens if we have some narrow changes in content between training and test…

Vogler & Pearl 2019

Narrow-change of content

train ≠ test

But now let's see what happens if we have some narrow changes in content between training and test…

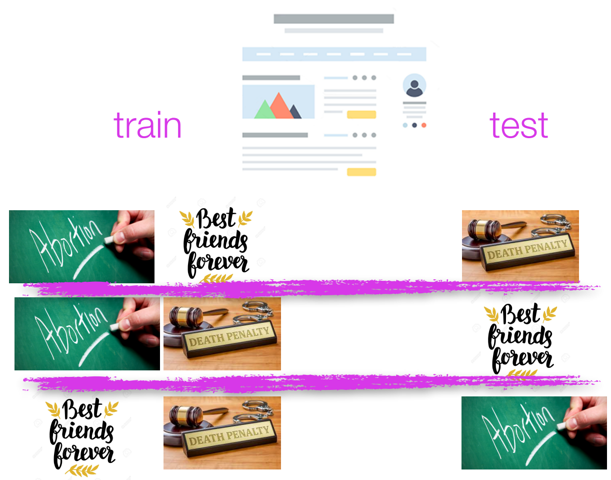
changes in product review valence

Vogler & Pearl 2019

Narrow-change of content

train ≠ test

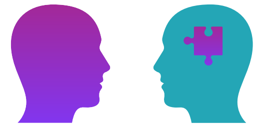
But now let's see what happens if we have some narrow changes in content between training and test…



changes in opinion essay topic

Vogler & Pearl 2019

Narrow-change of content



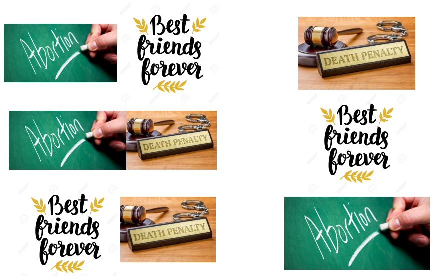
train ≠ test

specific detail

specific detail

	0.78	0.70	0.77	0.71		
	0.82	0.75	0.82	0.76		

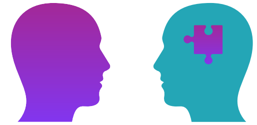
When only valence of the review changes, SVMs incorporating n-grams still do okay — though there's a performance drop compared to the within-domain performance.



There's no obvious gain when incorporating specific details.

Vogler & Pearl 2019

Narrow-change of content



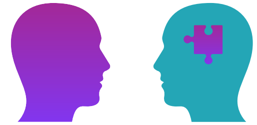
train ≠ test

0.78	0.70	0.77	0.71
0.82	0.75	0.82	0.76

We also see better performance for truthful reviews, compared with deceptive reviews.

Vogler & Pearl 2019

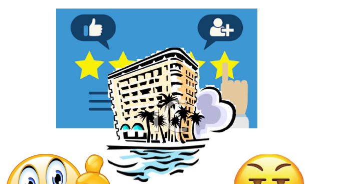
Narrow-change of content



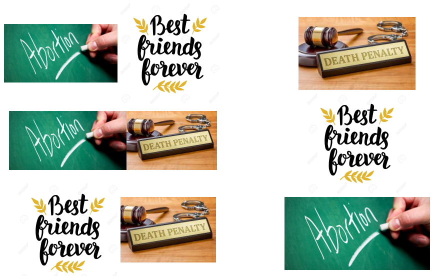
train ≠ test

specific detail

specific detail



	0.78	0.70	0.77	0.71	0.58	0.63
	0.82	0.75	0.82	0.76	0.50	0.68

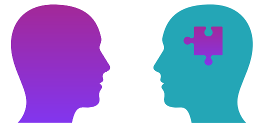


With specific details alone, we again see a comparative performance drop. But, we see better performance on deceptive reviews.

(And it was about the same as the within-domain performance.)

Vogler & Pearl 2019

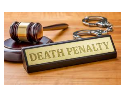
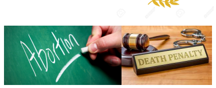
Narrow-change of content



train ≠ test

				specific detail		specific detail	
		0.78	0.70	0.77	0.71	0.58	0.63
		0.82	0.75	0.82	0.76	0.50	0.68

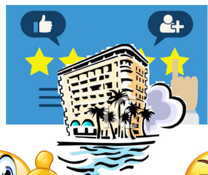
For more substantial content change, n-gram approaches have more significant drops in performance.

		0.56	0.65	0.60	0.66
		0.55	0.61	0.54	0.63
		0.64	0.71	0.64	0.72

Vogler & Pearl 2019

Narrow-change of content

train ≠ test



	specific detail	specific detail

0.78	0.70	0.77	0.71	0.58	0.63
0.82	0.75	0.82	0.76	0.50	0.68

Though interestingly, they now perform better with deceptive essays than truthful ones.

0.56	0.65	0.60	0.66
0.55	0.61	0.54	0.63
0.64	0.71	0.64	0.72

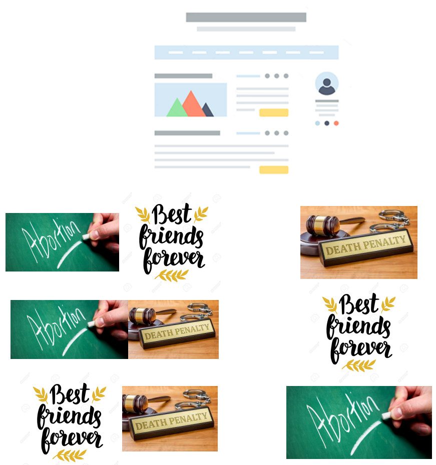
Vogler & Pearl 2019

Narrow-change of content

train ≠ test

0.78	0.70	0.77	0.71	0.58	0.63
0.82	0.75	0.82	0.76	0.50	0.68

Specific details alone can drop their performance some (or a lot).



0.56	0.65	0.60	0.66	0.64	0.54
0.55	0.61	0.54	0.63	0.55	0.66
0.64	0.71	0.64	0.72	0.39	0.64

Vogler & Pearl 2019

Narrow-change of content

train ≠ test

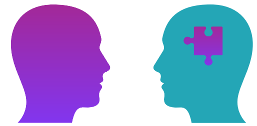
		specific detail		specific detail	
0.78	0.70	0.77	0.71	0.58	0.63
0.82	0.75	0.82	0.76	0.50	0.68

And usually perform better on deceptive essays (though sometimes truthful ones).

0.56	0.65	0.60	0.66	0.64	0.54
0.55	0.61	0.54	0.63	0.55	0.66
0.64	0.71	0.64	0.72	0.39	0.64

Vogler & Pearl 2019

Narrow-change of content

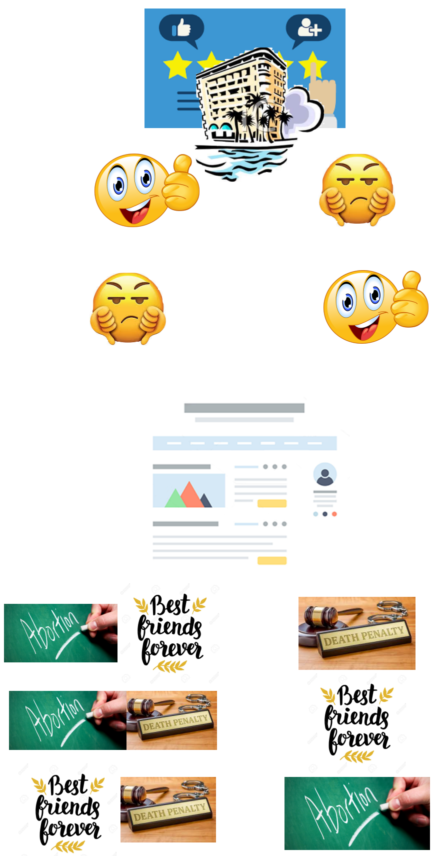


Takeaway: If the change in content is fairly minimal, n-grams alone will still do well enough (though not as well as when there's no change).

Vogler & Pearl 2019

Narrow-change of content

train $\neq$ test



Takeaway 2: If the change in content is more substantial, there can be some benefit to incorporating specific details (or using them alone).

Vogler & Pearl 2019

Broad-change of content

train ≠ test

But now let's see what happens if we have some more substantial changes in content between training and test…

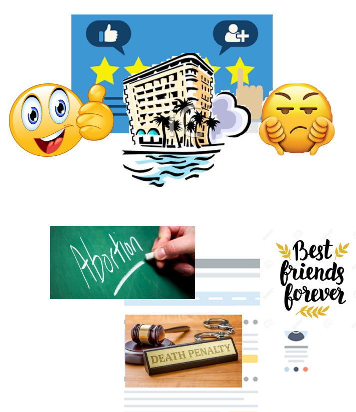
Vogler & Pearl 2019

Broad-change of content

$\boxed{\text{train} \neq \text{test}}$

But now let's see what happens if we have some more substantial changes in content between training and test…

train



test

changes in content and form

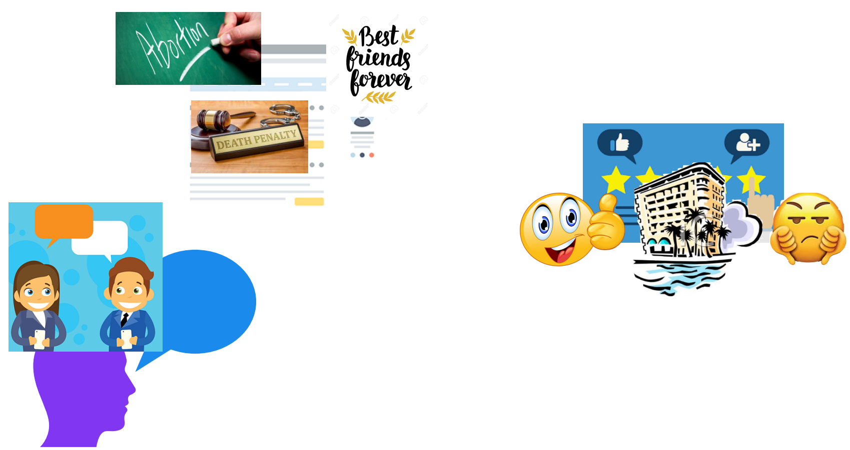
Vogler & Pearl 2019

Broad-change of content

train ≠ test

But now let's see what happens if we have some more substantial changes in content between training and test…

train test



changes in content and form

Vogler & Pearl 2019

Broad-change of content

train ≠ test

But now let's see what happens if we have some more substantial changes in content between training and test…

train

test

changes in content and form

Vogler & Pearl 2019

Broad-change of content

specific detail

specific detail

train ≠ test

0.60	0.44	0.60	0.48	0.32	0.63
0.60	0.39	0.64	0.28	0.57	0.52
0.63	0.29	0.63	0.29	0.37	0.64

Performance really plummets overall, but approaches using n-grams take a real hit for deceptive data.

Vogler & Pearl 2019

Broad-change of content

train ≠ test				specific detail		specific detail	
	0.60	0.44	0.60	0.48	0.32	0.63
	0.60	0.39	0.64	0.28	0.57	0.52
	0.63	0.29	0.63	0.29	0.37	0.64

In contrast, specific details alone do much better for deceptive data, even though performance on truthful data suffers in comparison.

Vogler & Pearl 2019

Broad-change of content

train ≠ test				
	0.37	0.42	0.81	recall
	0.31	0.18	0.49	recall
	0.19	0.20	0.82	recall

These differences are really due to recall performance — specific details alone have relatively great deceptive recall.

Vogler & Pearl 2019

Broad-change of content

train ≠ test			specific detail		specific detail		
	0.70	*0.37*	0.67	*0.42*	0.22	*0.81*	recall
	0.72	*0.31*	0.85	*0.18*	0.61	*0.49*	recall
	0.84	*0.19*	0.84	*0.20*	0.27	*0.82*	recall

...though this comes at the expense of truthful recall.

Vogler & Pearl 2019

Broad-change of content

train ≠ test

Takeaway: If the change in content is really substantial, there can be some benefit in detecting truthful data when incorporating specific details….but n-grams alone can be just as good.

Vogler & Pearl 2019

Broad-change of content

train ≠ test

Takeaway 2: For detecting deceptive data, if the change in content is really substantial, there's significant benefit to relying just on specific details alone.

Vogler & Pearl 2019

Broad-change of content

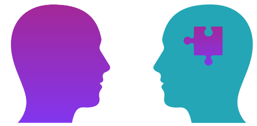
train ≠ test

Takeaway 3: When we dig into which specific details the classifiers relied on the most to produce this behavior (when these features were available features), the Number feature was far and away the most used.

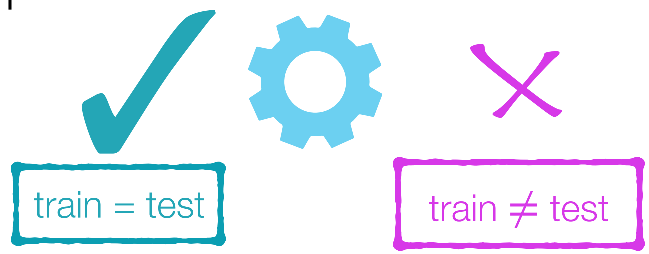
Number = "*two* minutes”

Vogler & Pearl 2019

Things we learned

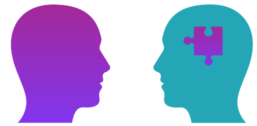


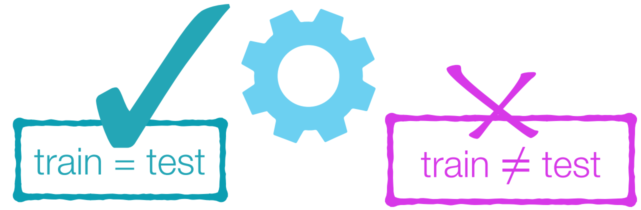
General-purpose language model features like n-grams are great for detecting deception within-domain, but classifier performance drops precipitously the more the content changes between training and test.



Vogler & Pearl 2019

Things we learned

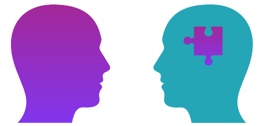


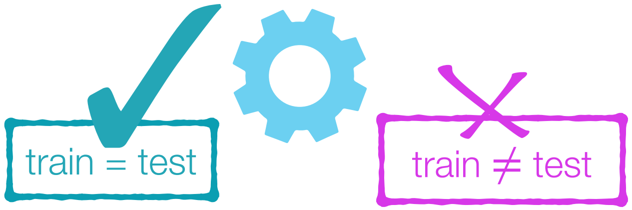


Linguistically-defined specific detail features (especially exact numbers) shine when there are dramatic content changes between training and test…

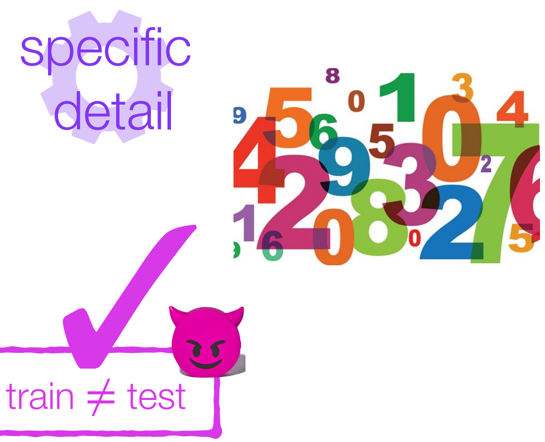
Vogler & Pearl 2019

Things we learned



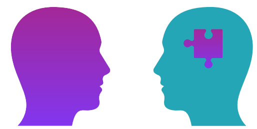


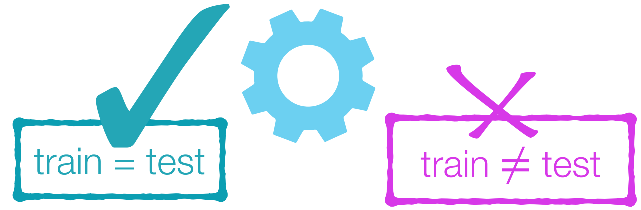
Linguistically-defined specific detail features (especially exact numbers) shine when there are dramatic content changes between training and test, especially if it's more important to make sure no deceptive data slip through undetected (deceptive recall). But this comes at the expense of marking truthful data as deceptive.

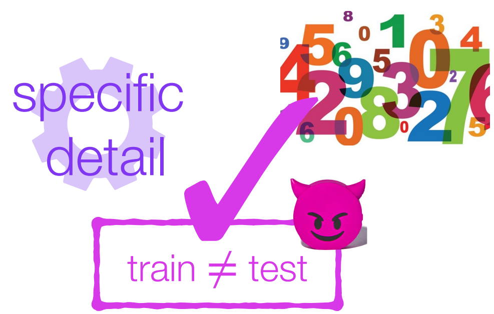


Vogler & Pearl 2019

Things we learned







So…if your training data are pretty different from the test data you have

and

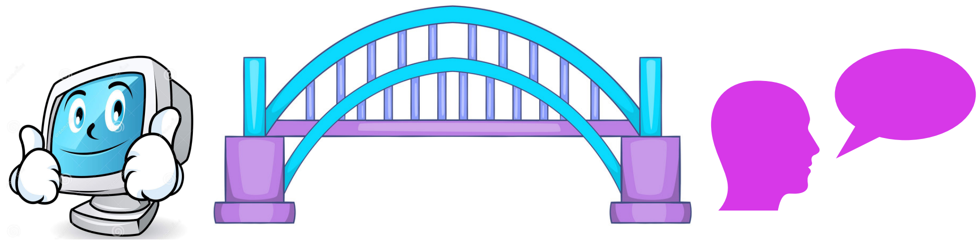
if it's more important to you not to let a false statement slip through (without further monitoring by humans, for example)

then

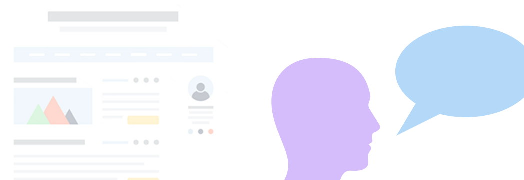
incorporating linguistically-defined specific details into your features is probably worthwhile.

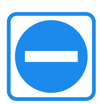
Vogler & Pearl 2019

Today's plan



Deception detection
across content domains





Negation handling in
sentiment analysis

The problem with negation in sentiment analysis

"This product truly did not live up to the expectations; or advertised results! Will not repurchase. Do not recommend it”

(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)

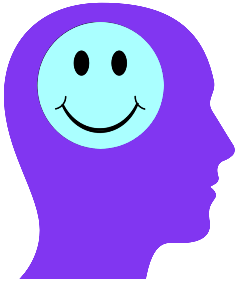
Hii, Yuen, & Pearl in prep.

The problem with negation in sentiment analysis

"This product truly did not live up to the expectations; or advertised results! Will not repurchase. Do not recommend it"

(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)

Sentiment analysis



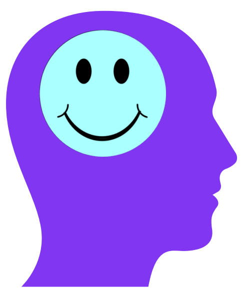
Do you think this is more likely to be a 5-star (positive) review, a 3-star (neutral) review or a 1-star (negative) review?

Hii, Yuen, & Pearl in prep.

The problem with negation in sentiment analysis

"This product truly did not live up to the expectations; or advertised results! Will not repurchase. Do not recommend it”

(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)



Sentiment analysis

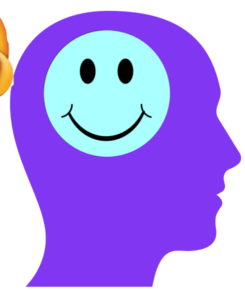
Most people say negative.

Hii, Yuen, & Pearl in prep.

The problem with negation in sentiment analysis

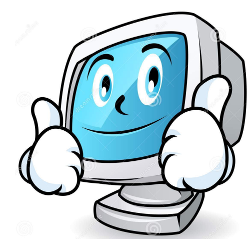
"This product truly did not live up to the expectations; or advertised results! Will not repurchase. Do not recommend it"

(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)



Sentiment analysis

The problem: Many state-of-the-art sentiment analyzers say it's positive.

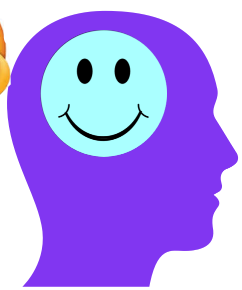


Hii, Yuen, & Pearl in prep.

The problem with negation in sentiment analysis

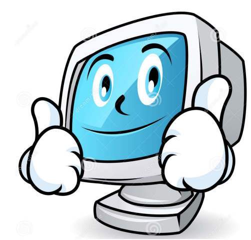
"This product truly did not live up to the expectations; or advertised results! Will not repurchase. Do not recommend it”

(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)



Sentiment analysis

The problem: Many state-of-the-art sentiment analyzers say it’s positive.



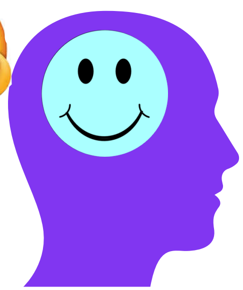
Why???

Hii, Yuen, & Pearl in prep.

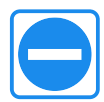
The problem with negation in sentiment analysis

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

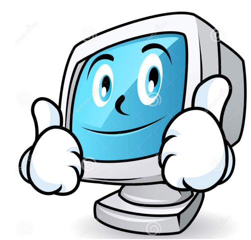
(actual product review from the Amazon product review corpus: He and McAuley, 2016; McAuley et al., 2015)



Negation

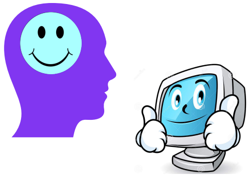


The problem: Inability to handle negation, which can drastically alter the sentiment expressed.



Why???

Hii, Yuen, & Pearl in prep.



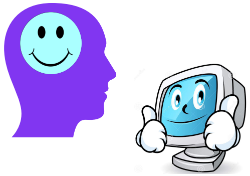
So how do we handle negation?

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"

Two components:

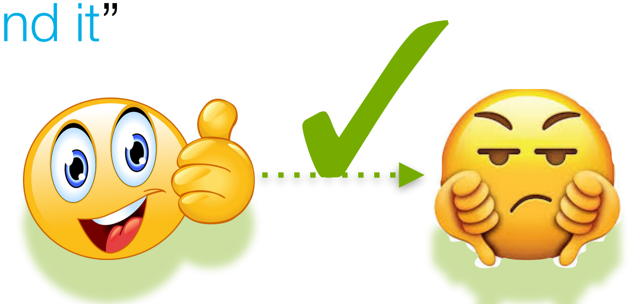
(1) Detect the scope of negation = what parts of the message get their sentiment altered by negation

Hii, Yuen, & Pearl in prep.



So how do we handle negation?

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"



Two components:

(1) Detect the scope of negation

(2) Resolve negation = update the sentiment of the language within the scope of negation

Hii, Yuen, & Pearl in prep.

So how do we handle negation?

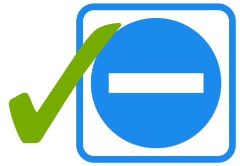
We're going to focus on negation resolution, since there seem to be some pretty good approaches out there for scope detection.

The issue is more what to do about it once you've identified something needs to be done.

(1) Detect the scope of negation

(2) Resolve negation = update the sentiment of the language within the scope of negation

Hii, Yuen, & Pearl in prep.



Negation resolution

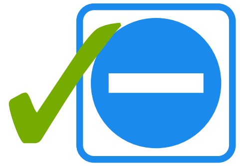
"This is **not** good"

Many current symbolic approaches rely on a sentiment lexicon that provides the "base sentiment" for words and phrases (SemEval2015-English-Twitter-Lexicon, SCL-NMA, SCL-OPP, NRC-Hashtag-Sentiment-Lexicon-v1.0, NRC-Emoticon-Lexicon-v1.0, NRC-Hashtag-Sentiment- AffLexNegLex-v1.0, NRC-Emoticon- AffLexNegLex-v1.0). This sentiment is what gets altered if it's in the scope of negation.

$-1 \leq \text{sentiment} \leq 1$

good = 0.66

Hii, Yuen, & Pearl in prep.



Negation resolution

"This is **not** good"

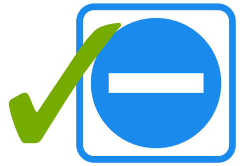
Several existing approaches to altering the base sentiment:

good = 0.66

Just invert the score.

not good = -0.66

Hii, Yuen, & Pearl in prep.



Negation resolution

"This is **not** good"

Several existing approaches to altering the base sentiment:

good = 0.66

Just invert the score.

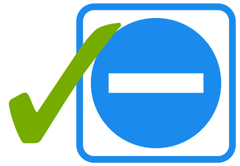
not good = -0.66

The problem: Relative sentiment scores get messed up.

terrible = -0.7

not terrible = 0.7

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

Several existing approaches to altering the base sentiment:

Just invert the score.

not good = -0.66

The problem: Relative sentiment scores get messed up.

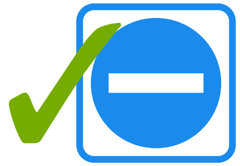
terrible = -0.7

not terrible = 0.7

good = 0.66

Is “not terrible” more positive than “good”?
It shouldn’t be…

Hii, Yuen, & Pearl in prep.



Negation resolution

invert not good = -0.66

“This is **not** good”

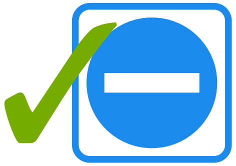
Several existing approaches to altering the base sentiment: good = 0.66

An observation: Negating positive terms seems to involve a different amount of sentiment shifting than negating negative terms.

terrible

good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

“This is **not** good”

Several existing approaches to altering the base sentiment:

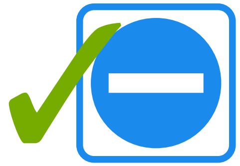
good = 0.66

An observation: Positive terms get more of a shift (Kiritchenko et al., 2014).

terrible

good

Hii, Yuen, & Pearl in prep.

Negation resolution

invert | not good = -0.66

"This is **not** good"

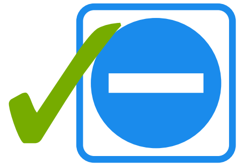
Several existing approaches to altering the base sentiment: good = 0.66

An observation: Positive terms get more of a shift (Kiritchenko et al., 2014).

terrible   not good ◄··································· good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

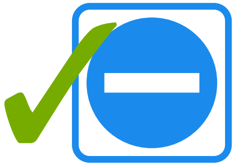
“This is **not** good”

Several existing approaches to altering the base sentiment: good = 0.66

An observation: Negative terms get less of a shift (Kiritchenko et al., 2014).

terrible   not good   good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert $\boxed{\text{not good = -0.66}}$

“This is **not** good”

Several existing approaches to altering the base sentiment:

good = 0.66

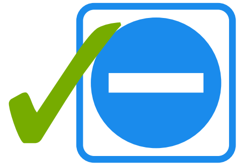
An observation: Negative terms get less of a shift (Kiritchenko et al., 2014).

terrible ⋯⋯▶ not terrible

not good

good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

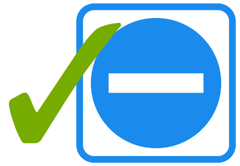
“This is **not** good”

Several existing approaches to altering the base sentiment: good = 0.66

Solution: Implement an asymmetrical shift (Socher et al. 2013), where positive terms shift one amount and negative terms shift a different (lesser) amount.

terrible | not good | not terrible | good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

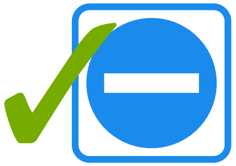
“This is **not** good”

good = 0.66

Several existing approaches to altering the base sentiment:

Another observation: A term’s base sentiment score may not capture all the components necessary to accurately compute its negated sentiment score.

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

“This is **not** good”

Several existing approaches to altering the base sentiment:

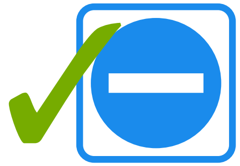
good = 0.66

Another observation: A term’s base sentiment score may not capture all the components necessary to accurately compute its negated sentiment score.

One solution: Leverage a term’s antonym, which is more closely connected to the nuances of its meaning. Use the antonym’s base sentiment score as the negated score (Carrillo-de Albornoz and Plaza, 2013).

good
bad = -0.5
not good = -0.5

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

“This is **not** good”

Several existing approaches to altering the base sentiment:

good = 0.66

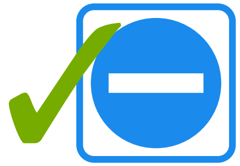
One solution: Leverage a term’s antonym.

good
bad = -0.5
not good = -0.5

A problem: Reliably finding a term’s antonym.

recommend
antonym not in WordNet

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

"This is **not** good"

Several existing approaches to altering the base sentiment:

good = 0.66

One solution: Leverage a term's antonym.

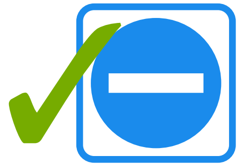
good
bad = -0.5
not good = -0.5

A problem: Reliably finding a term's antonym.

recommend
antonym not in WordNet

synonyms = urge, advocate
antonyms not in WordNet

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

“This is **not** good”

Several existing approaches to altering the base sentiment: good = 0.66

One solution: Leverage a term’s antonym.

good
bad = -0.5
not good = -0.5

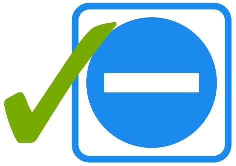
A problem: Reliably finding a term’s antonym.

recommend
antonym not in WordNet

synonyms = urge, advocate
antonyms not in WordNet

related forms = recommendation, urgency
antonyms not in WordNet

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

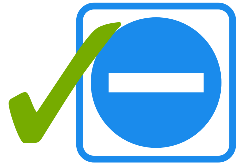
antonym —
good
bad = -0.5
not good = -0.5

“This is **not** good”

good = 0.66

But the idea that nuances of meaning may matter seems right.

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

But the idea that nuances of meaning may matter seems right. ✓

One linguistic intuition: how specific a term's meaning is may impact how much it gets shifted

beautiful
nice
good

Shifted scores from Kiritchenko et al., 2014

Hii, Yuen, & Pearl in prep.

Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

But the idea that nuances of meaning may matter seems right.

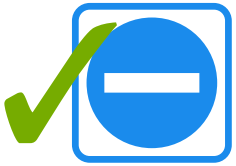
One linguistic intuition: how specific a term's meaning is may impact how much it gets shifted

more specific

beautiful
nice
good

Shifted scores from Kiritchenko et al., 2014

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

But the idea that nuances of meaning may matter seems right. ✓

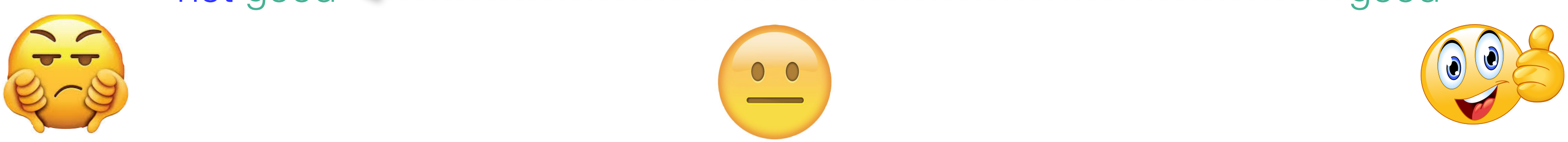
One linguistic intuition: how specific a term's meaning is may impact how much it gets shifted

more specific

beautiful

nice

not good ◄ good



Shifted scores from Kiritchenko et al., 2014

Hii, Yuen, & Pearl in prep.

Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

But the idea that nuances of meaning may matter seems right. ✓

One linguistic intuition: how specific a term's meaning is may impact how much it gets shifted

more specific

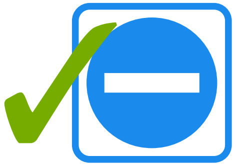
beautiful

not nice ◀······ nice

not good ◀······ good

Shifted scores from Kiritchenko et al., 2014

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

But the idea that nuances of meaning may matter seems right. ✓

One linguistic intuition: how specific a term's meaning is may impact how much it gets shifted

more specific

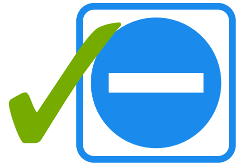
not beautiful ← beautiful

not nice ← nice

not good ← good

Shifted scores from Kiritchenko et al., 2014

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

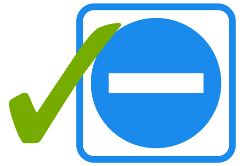
more specific

beautiful

nice

good

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

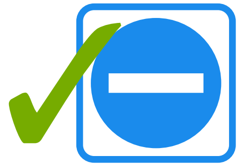
How do we tell how specific a term's meaning is?

more specific

beautiful
nice
good

Some ideas:

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

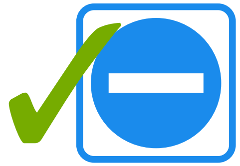
How do we tell how specific a term's meaning is?

more specific

beautiful
nice
good

Some ideas:
Frequency (less frequent terms may be more specific — that could be why they appear less frequently)

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

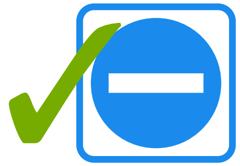
more specific

beautiful
nice
good

Some ideas:
Frequency (less frequent terms may be more specific — that could be why they appear less frequently)

Variety of contexts (terms that appear in fewer contexts may be more specific — they're only appropriate in certain contexts)

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

more specific

beautiful

nice

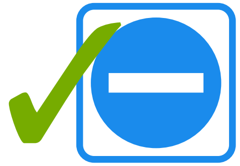
good

Some heuristics that we might use to approximate meaning specificity:

Frequency — Calculated by using the 82.8 million Amazon product reviews corpus (McAuley et al. 2015, He and McAuley 2016)

Inverse dispersion (Gries 2008): $0 \leq$ InvDisp ≤ 1, 0 = uniform distribution across contexts while 1 = only in a single context

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

more specific

beautiful
nice
good

Some heuristics that we might use to approximate meaning specificity:

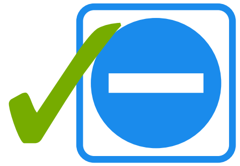
Frequency Calculated by using the 82.8 million Amazon product reviews corpus (McAuley et al. 2015, He and McAuley 2016)

Inverse dispersion (Gries 2008): 0 ≤ InvDisp ≤ 1, 0 = uniform distribution across contexts while 1 = only in a single context

Sums difference of observed relative frequency vs. expected relative frequency if there were a uniform distribution across contexts (divide by 2 so range ∈ [0,1])

$$\sum_{i=1}^{contexts} \frac{|observed_{term_i} - expected_{term_i}|}{2}$$

Hii, Yuen, & Pearl in prep.



Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

more specific

beautiful
nice
good

Some heuristics that we might use to approximate meaning specificity:

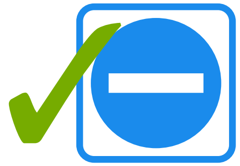
Frequency

Inverse dispersion (Gries 2008): 0 $\leq$ InvDisp $\leq$ 1, 0 = uniform distribution across contexts while 1 = only in a single context

Calculated by using the 82.8 million Amazon product reviews corpus (McAuley et al. 2015, He and McAuley 2016), and dividing it into 10 equal-size sections as contexts.

$$\sum_{i=1}^{contexts} \frac{|observed_{term_i} - expected_{term_i}|}{2}$$

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

How do we tell how specific a term's meaning is?

more specific

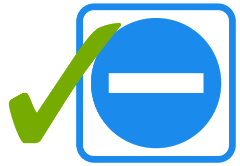
beautiful
nice
good

Goal: Combine these two heuristics to approximate a term's meaning specificity
Freq, InvDisp

Note: Similar in spirit to tf-idf, which involves term frequency and inverse document frequency. But we can't easily use standard tf-idf, because product reviews are so short that term frequency is 1 or 0 in a product review — therefore, the frequency part isn't useful.

Instead: Here term frequency is calculated over the entire corpus so we don't have that problem.

Hii, Yuen, & Pearl in prep.



Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

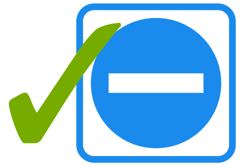
meaning specificity ≈ Freq, InvDisp

more specific

beautiful
nice
good

But how do we combine these two quantities in a sensible way?

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

“This is **not** good”

good = 0.66

meaning specificity ≈ Freq, InvDisp

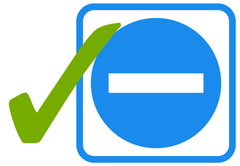
more specific

beautiful
nice
good

But how do we combine these two quantities in a sensible way?

One answer: Do a multiple regression, with the data coming from a set of terms whose ground truth (both base score and negated score) we feel pretty certain about.

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

meaning specificity ≈ Freq, InvDisp

more specific

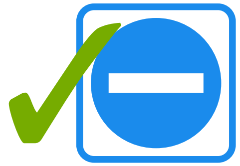
beautiful
nice
good

But how do we combine these two quantities in a sensible way?

One answer: Do a multiple regression, with the data coming from a set of terms whose ground truth (both base score and negated score) we feel pretty certain about.

Data like this: 42 terms extracted and manually checked from Kiritchenko et al. 2014

Hii, Yuen, & Pearl in prep.



Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

“This is **not** good”

good = 0.66

meaning specificity ≈ Freq, InvDisp

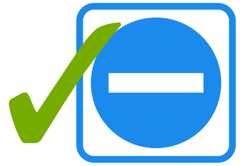
more specific

beautiful
nice
good

The resulting negation calculation equation, using these terms.

$$Negated = -0.061 - 0.39 * base + 2.77 * Freq - 2.26 * InvDisp - 705.61 * Freq * InvDisp$$

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

“This is **not** good”

good = 0.66

meaning specificity ≈ Freq, InvDisp

more specific

beautiful
nice
good

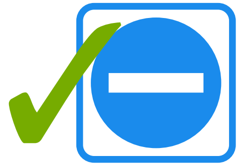
The resulting negation calculation equation, using these terms.

$$Negated = -0.061 - 0.39 * base + 2.77 * Freq - 2.26 * InvDisp - 705.61 * Freq * InvDisp$$

not good

The negated score …

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

meaning specificity ≈ Freq, InvDisp

more specific

beautiful
nice
good

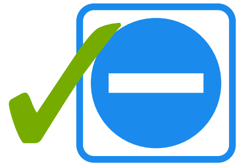
The resulting negation calculation equation, using these terms.

$$Negated = -0.061 - 0.39 * base + 2.77 * Freq - 2.26 * InvDisp - 705.61 * Freq * InvDisp$$

not good — good = 0.66

The negated score depends some on the base score …

Hii, Yuen, & Pearl in prep.



Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym: good, bad = -0.5, not good = -0.5

"This is **not** good"

good = 0.66

meaning specificity ≈ Freq, InvDisp

more specific

beautiful

nice

good

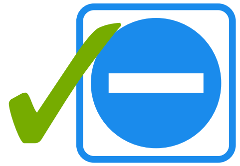
The resulting negation calculation equation, using these terms.

$$Negated = -0.061 - 0.39 * base + 2.77 * Freq - 2.26 * InvDisp - 705.61 * Freq * InvDisp$$

not good   good = 0.66

The negated score depends some on the base score, more on frequency and inverse dispersion individually …

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

good = 0.66

meaning specificity ≈ Freq, InvDisp

more specific

beautiful
nice
good

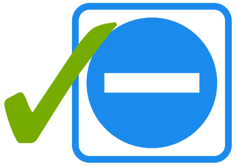
The resulting negation calculation equation, using these terms.

$$Negated = -0.061 - 0.39 * base + 2.77 * Freq - 2.26 * InvDisp - 705.61 * Freq * InvDisp$$

not good | good = 0.66

The negated score depends some on the base score, more on frequency and inverse dispersion individually, and a heck of a lot more on the interaction of frequency and inverse dispersion.

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

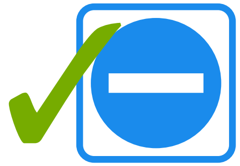
"This is **not** good"

meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

good = 0.66

Sanity check: Does adding in this heuristic meaning specificity information help at all?

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

“This is **not** good”

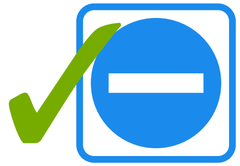
meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

good = 0.66

Information theory says …
if we use this meaning specificity approach when trying to calculate the negated score, given the base score …

not good = ???   good = 0.66

Hii, Yuen, & Pearl in prep.



Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good; bad = -0.5; not good = -0.5

"This is **not** good"

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

good = 0.66

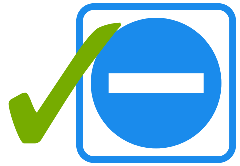
Information theory says …
if we use this meaning specificity approach when trying to calculate the negated score, given the base score …

not good = ??? good = 0.66

… we find information gain, compared to not using meaning specificity information.

$$I[meaning\ specificity : negated \mid base] = H[negated \mid base] - H[negated \mid base, meaning\ specificity]$$

Hii, Yuen, & Pearl in prep.



Negation resolution

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

"This is **not** good"

meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

good = 0.66

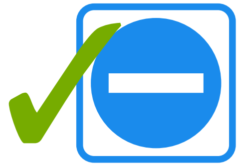
Information theory says …
if we use this meaning specificity approach when trying to calculate the negated score, given the base score …

not good = ??? good = 0.66

… and we find 4.2 times the information gain, compared with using random meaning specificity values.

$$I[meaning\ specificity : negated \mid base] = H[negated \mid base, random\ meaning] - H[negated \mid base, meaning\ specificity]$$

Hii, Yuen, & Pearl in prep.



Negation resolution

invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

"This is **not** good"

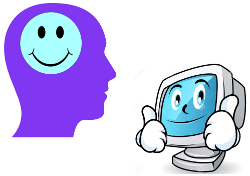
meaning specificity | not good ≈
Freq, InvDisp,
Freq*InvDisp

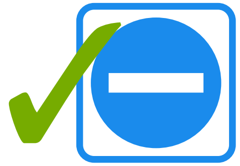
good = 0.66

✓ Information theory says yes.

✓ Exploratory decision tree analysis (forced maximum depth of 3) also showed information gain when relying on the meaning specificity features (frequency, inverse dispersion, and their interaction), as opposed to just base score.

Hii, Yuen, & Pearl in prep.





Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

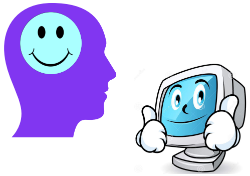
asym shift — not good vs. not terrible

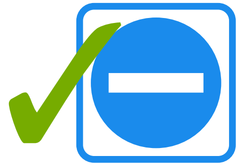
antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈
Freq, InvDisp,
Freq*InvDisp

So let’s try our meaning specificity approach in a negation resolution evaluation pipeline, where the goal is to classify a product review involving negation as either positive, neutral, or negative.

Hii, Yuen, & Pearl in prep.





Negation resolution

"This is **not** good"

good = 0.66

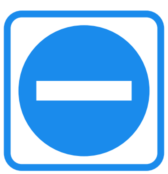
invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

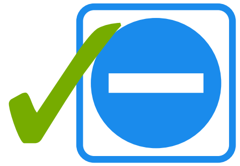
So let's try our meaning specificity approach in a negation resolution evaluation pipeline, where the goal is to classify a product review involving negation as either positive, neutral, or negative.



nothing — not good = 0.66

We can compare it against all the other approaches, as well as a baseline of doing nothing when encountering negation.

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

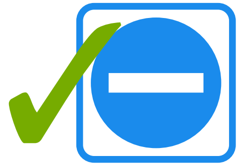
invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection

Hii, Yuen, & Pearl in prep.



Negation resolution

"This is **not** good"

good = 0.66

invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66

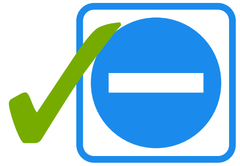
"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"

Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection

Several state-of-the-art methods (4-grams: Blair- Goldensohn et al. 2008; Taboada et al. 2011; Thelwall et al. 2012; parse trees: Carrillo-de Albornoz and Plaza 2013; Socher et al. 2013; NegTool: Enger et al. 2017)

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

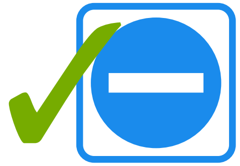
Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection

Several state-of-the-art methods (4-grams: Blair- Goldensohn et al. 2008; Taboada et al. 2011; Thelwall et al. 2012; parse trees: Carrillo-de Albornoz and Plaza 2013; Socher et al. 2013; NegTool: Enger et al. 2017)

We’ll try all of these out, since it’s unclear a priori which will work best in the final sentiment analysis result.

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good; bad = -0.5; not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

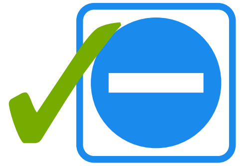
Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection

(2) Negation scope resolution

Which will be one of these options.

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good; bad = -0.5; not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

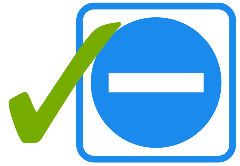
Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection 

(2) Negation scope resolution 

Which will be one of these options.

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

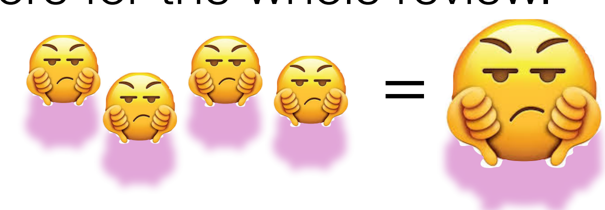
"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

Remember that there are two key parts of a sentiment analysis pipeline

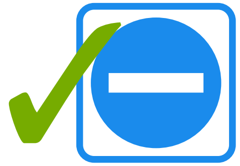
(1) Negation scope detection 

(2) Negation scope resolution 

…and then we have to **aggregate** the different sentiment scores into one score for the whole review.

$\sum$ 

Hii, Yuen, & Pearl in prep.



Negation resolution

“This is **not** good”

good = 0.66

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good; bad = -0.5; not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it”

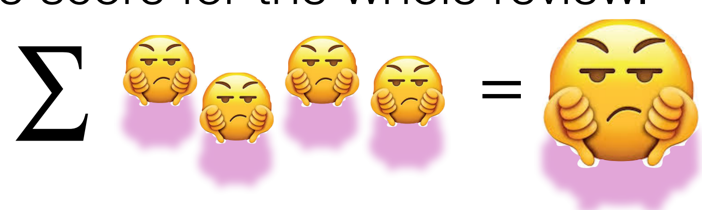
Remember that there are two key parts of a sentiment analysis pipeline

(1) Negation scope detection

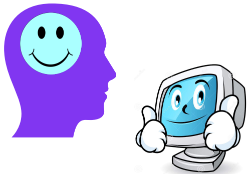
(2) Negation scope resolution

…and then we have to **aggregate** the different sentiment scores into one score for the whole review.

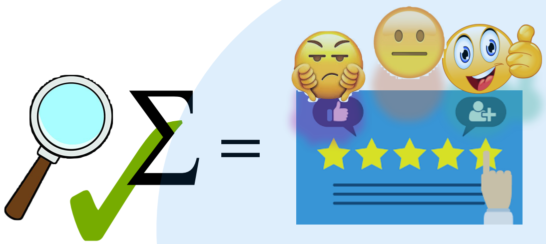
We’ll try both flat averaging and aggregating structurally using a parse tree.



Hii, Yuen, & Pearl in prep.



Negation resolution



So what kind of reviews do we want to evaluate these approaches on?

invert: not good = -0.66

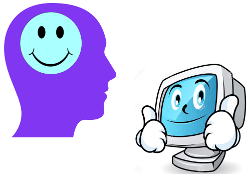
asym shift: not good vs. not terrible

antonym:
good
bad = -0.5
not good = -0.5

meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

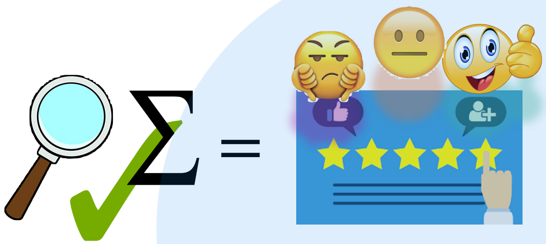
nothing: not good = 0.66

Hii, Yuen, & Pearl in prep.



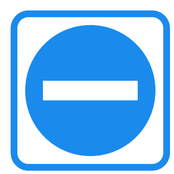
Negation resolution

invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66



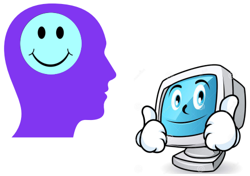
So what kind of reviews do we want to evaluate these approaches on?

Basic data: a collection of reviews that have negation in them



10,000 reviews from the Amazon product reviews corpus (McAuley et al 2015, He & McAuley 2016)

Hii, Yuen, & Pearl in prep.



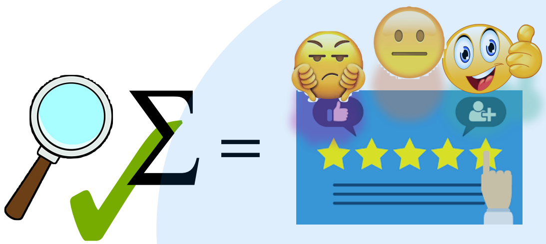
Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

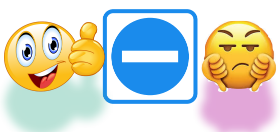
meaning specificity — not good ≈
Freq, InvDisp,
Freq*InvDisp



So what kind of reviews do we want to evaluate these approaches on?

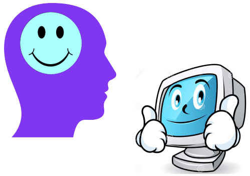
Basic 

Hard data: a collection of reviews that have negation in them, and the presence of negation changes the valence (from positive to negative or from negative to positive).

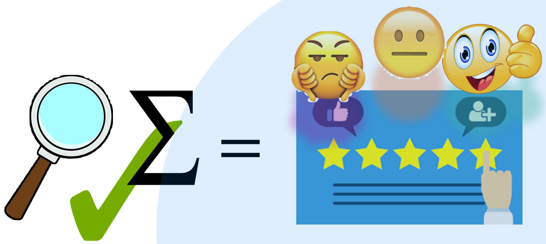


10,000 reviews from the Amazon product reviews corpus (McAuley et al 2015, He & McAuley 2016)

Hii, Yuen, & Pearl in prep.



Negation resolution



So what kind of reviews do we want to evaluate these approaches on?

invert | not good = -0.66

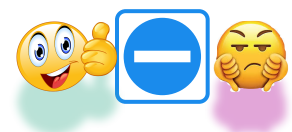
asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

Basic 

Hard data: a collection of reviews that have negation in them, and the presence of negation changes the valence (from positive to negative or from negative to positive).

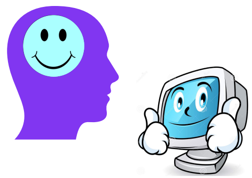


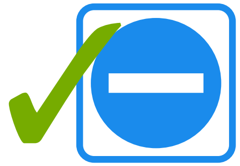
10,000 reviews from the Amazon product reviews corpus (McAuley et al 2015, He & McAuley 2016)

nothing | not good = 0.66

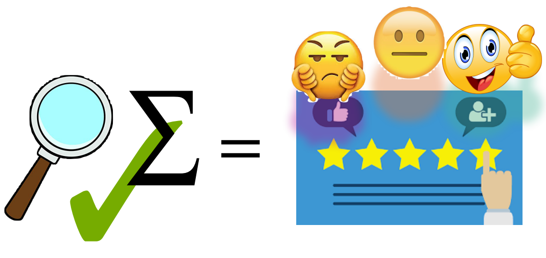
Upshot: if you do nothing, you definitely get the wrong answer.

Hii, Yuen, & Pearl in prep.

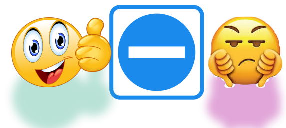




Negation resolution



Basic 

Hard 

invert: not good = -0.66

asym shift: not good vs. not terrible

antonym: good, bad = -0.5, not good = -0.5

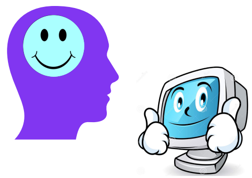
meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

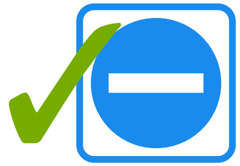
nothing: not good = 0.66

Evaluation metric (RI+partial): a version of the Rand Index (Rand 1971), aka "accuracy", that gives partial credit.

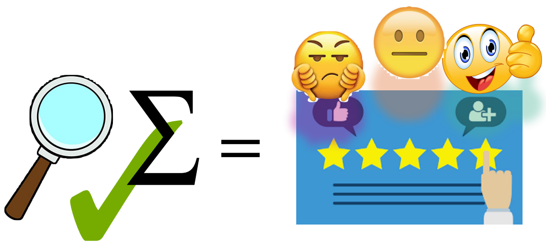
Intuition, part 1: Get full credit for correctly classifying negative reviews as negative, neutral reviews as neutral, and positive reviews as positive.

Hii, Yuen, & Pearl in prep.

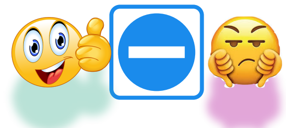




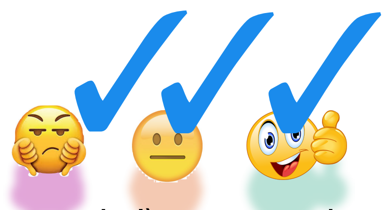
Negation resolution



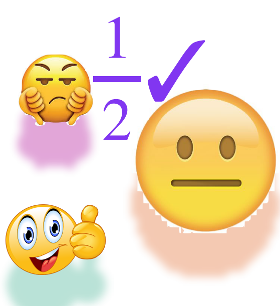
Basic 

Hard 

invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66



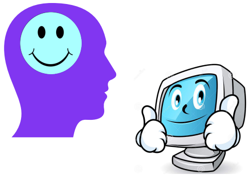
Evaluation metric (RI+partial): a version of the Rand Index (Rand 1971), aka "accuracy", that gives partial credit.

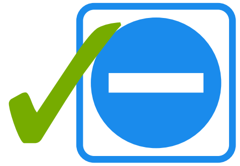


Intuition, part 2: Get half credit for classifying negative reviews as neutral or positive reviews as neutral.

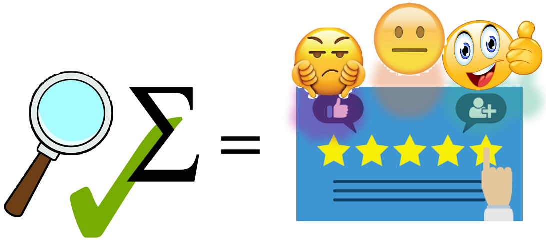
Why? Because this isn't as egregious as classifying positive reviews as negative or negative reviews as positive.

Hii, Yuen, & Pearl in prep.



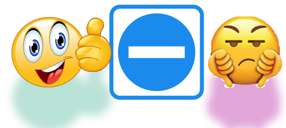


Negation resolution



Basic

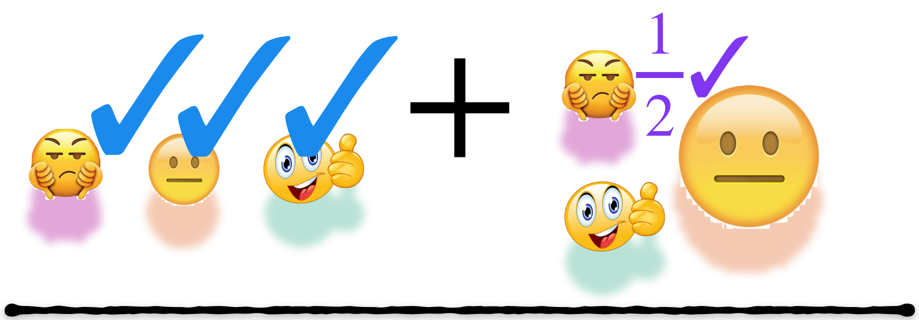
Hard



invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66

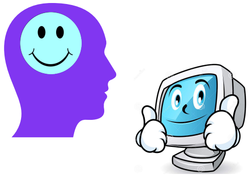
$$\boxed{0 \leq \text{RI+partial} \leq 1}$$

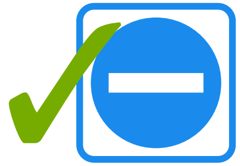
Evaluation metric (RI+partial): a version of the Rand Index (Rand 1971), aka “accuracy”, that gives partial credit.



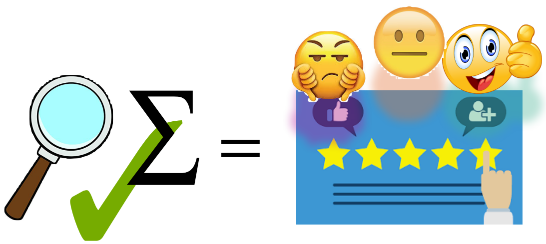
all 10,000 classifications

Hii, Yuen, & Pearl in prep.



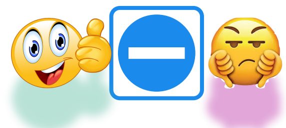


Negation resolution



What we found

Basic 

Hard 

invert: not good = -0.66

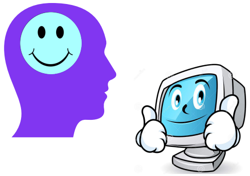
asym shift: not good vs. not terrible

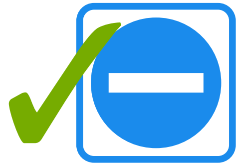
antonym:
good
bad = -0.5
not good = -0.5

meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

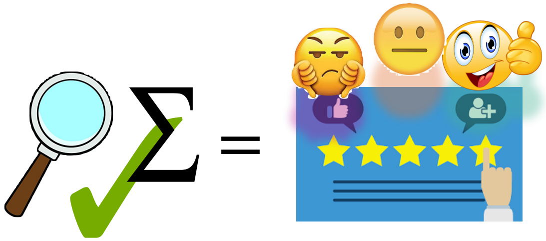
nothing: not good = 0.66

Hii, Yuen, & Pearl in prep.





Negation resolution



invert: not good = -0.66

asym shift: not good vs. not terrible

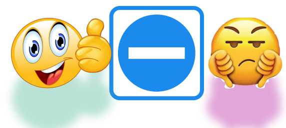
antonym:
good
bad = -0.5
not good = -0.5

meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

nothing: not good = 0.66

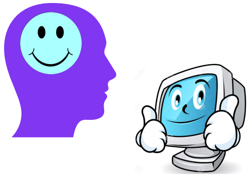
What we found

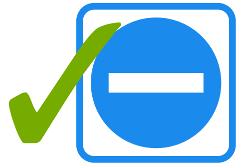
Basic 

Hard 

RI+partial:
Range = .557-.638

Hii, Yuen, & Pearl in prep.





Negation resolution

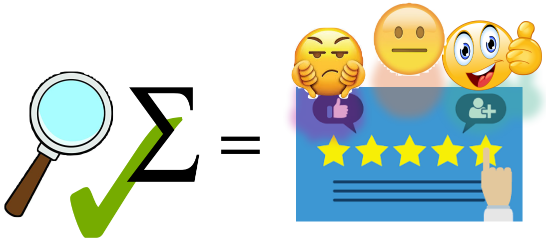
invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

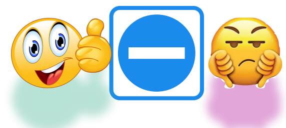
meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

nothing | not good = 0.66



What we found

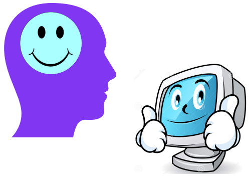
Basic 

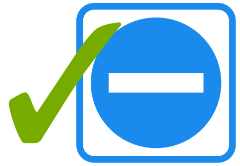
Hard 

RI+partial:
Range = .557-.638

inverting

Hii, Yuen, & Pearl in prep.





Negation resolution

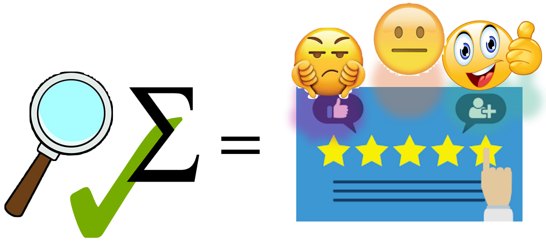
invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

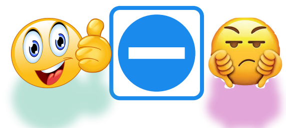
meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66



What we found

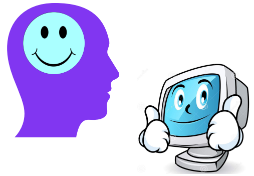
Basic 

Hard 

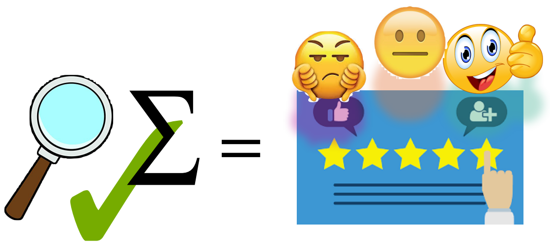
RI+partial:
Range = .557-.638
inverting

Note: Doing nothing already gets you to .629

Hii, Yuen, & Pearl in prep.



Negation resolution



invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66

What we found

Basic

Hard

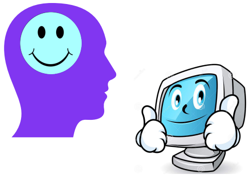
RI+partial:
Range = .557-.638

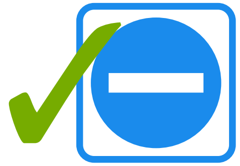
inverting

Note: Doing nothing already gets you to .629

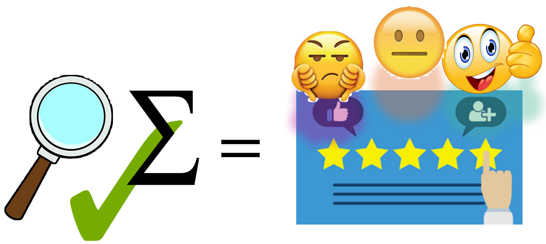
…and that may explain why the overly-simplistic inverting approach does so well. Handling negation cleverly doesn't get you much mileage in the basic cases.

Hii, Yuen, & Pearl in prep.





Negation resolution



invert | not good = -0.66

asym shift | not good vs. not terrible

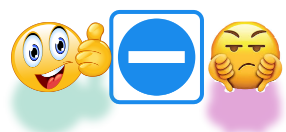
antonym | good
bad = -0.5
not good = -0.5

meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

nothing | not good = 0.66

What we found

Basic 

Hard 

RI+partial:
Range = .557-.638

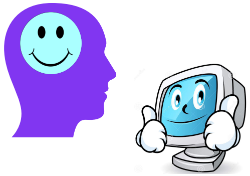
inverting

Note: Doing nothing already gets you to .629

Handling negation cleverly doesn't get you much mileage

("This case is as cute as it is durable. Your phone sits in a rubber casing that fits very snug. Your phone won't be falling out.")

Hii, Yuen, & Pearl in prep.



Negation resolution

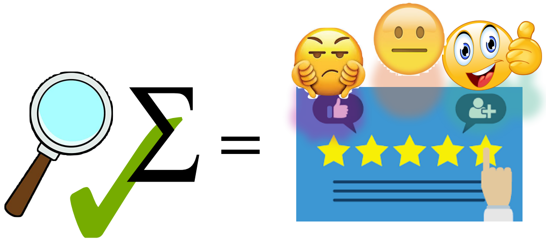
invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

meaning specificity | not good ≈ Freq, InvDisp, Freq*InvDisp

nothing | not good = 0.66



What we found

Basic

Hard

RI+partial:
Range = .557-.638

inverting

Note: Doing nothing already gets you to .629

Handling negation cleverly doesn't get you much mileage

("This case is as cute as it is durable. Your phone sits in a rubber casing that fits very snug. Your phone won't be falling out.")

Hii, Yuen, & Pearl in prep.

Negation resolution

invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

What we found

Basic

Hard

RI+partial:
Range = .557-.638

inverting

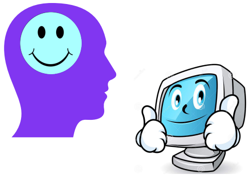
Note: Doing nothing already gets you to .629

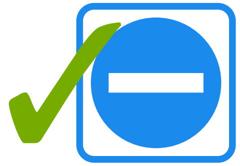
Handling negation cleverly doesn't get you much mileage

("This case is as cute as it is durable. Your phone sits in a rubber casing that fits very snug. Your phone won't be falling out.")

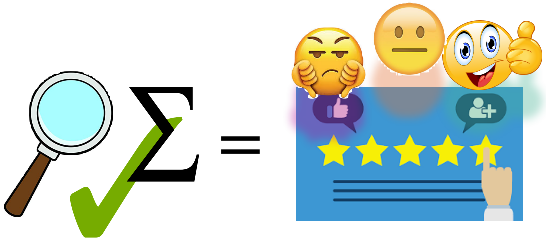
$\sum$ =

Hii, Yuen, & Pearl in prep.





Negation resolution



invert — not good = -0.66

asym shift — not good vs. not terrible

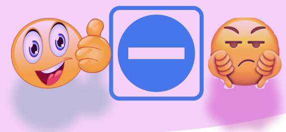
antonym — good; bad = -0.5; not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

What we found

Basic 

Hard 

RI+partial:
Range = .557-.638

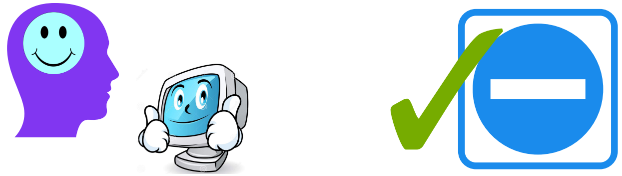
inverting

So let's turn to the hard cases, where doing nothing guarantees you the wrong sentiment.

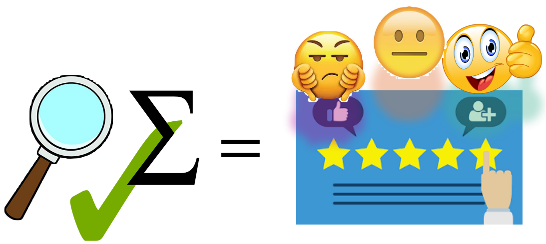
~~nothing~~

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"

Hii, Yuen, & Pearl in prep.



Negation resolution



What we found

Basic

Hard

RI+partial:
Range = .557-.638

inverting

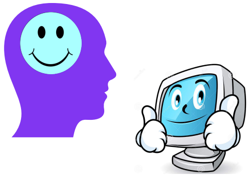
So let's turn to the hard cases, where doing nothing guarantees you the wrong sentiment.

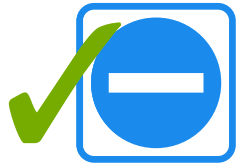
"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"

invert	not good = -0.66
asym shift	not good vs. not terrible
antonym	good bad = -0.5 not good = -0.5
meaning specificity	not good ≈ Freq, InvDisp, Freq*InvDisp
nothing	not good = 0.66

~~nothing~~

Hii, Yuen, & Pearl in prep.





Negation resolution

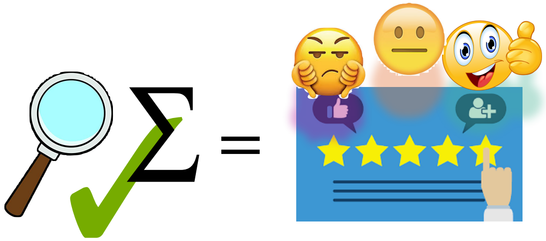
invert: not good = -0.66

asym shift: not good vs. not terrible

antonym: good
bad = -0.5
not good = -0.5

meaning specificity: not good ≈ Freq, InvDisp, Freq*InvDisp

nothing: not good = 0.66



What we found

Basic

Hard

RI+partial:
Range = .557-.638

inverting

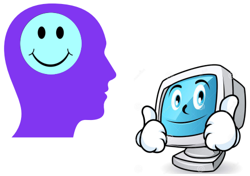
So let's turn to the hard cases, where doing nothing guarantees you the wrong sentiment.

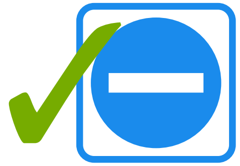
~~nothing~~

"This product truly did **not** live up to the expectations; or advertised results! Will **not** repurchase. Do **not** recommend it"

$\sum =$

Hii, Yuen, & Pearl in prep.





Negation resolution

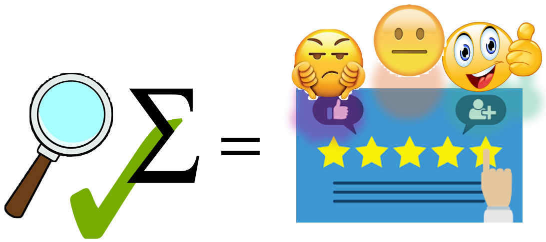
invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good / bad = -0.5 / not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

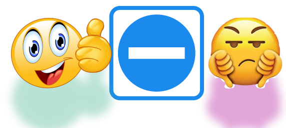


What we found

Basic 

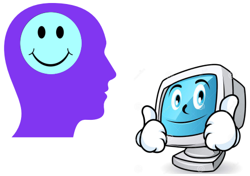
RI+partial:
Range = .557-.638

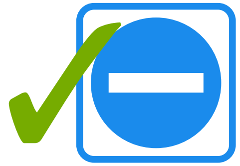
inverting

Hard 

RI+partial:
Range = .272-.658

Hii, Yuen, & Pearl in prep.





Negation resolution

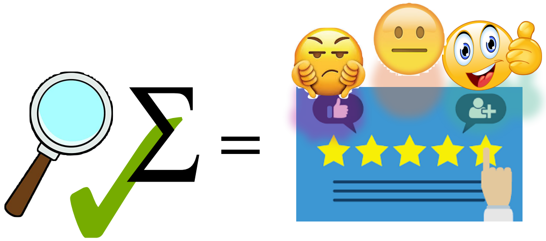
invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

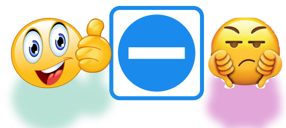


What we found

Basic 

RI+partial:
Range = .557-.638

inverting

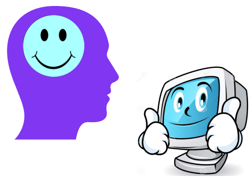
Hard 

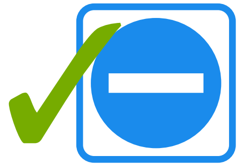
RI+partial:
Range = .272-.658

meaning specificity

Here, it's meaning specificity that gets you the best performance — and a much higher boost over the lower-performing approaches.

Hii, Yuen, & Pearl in prep.





Negation resolution

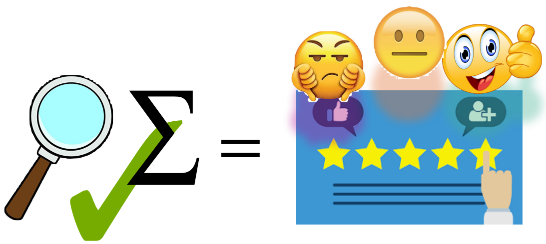
invert | not good = -0.66

asym shift | not good vs. not terrible

antonym | good
bad = -0.5
not good = -0.5

meaning specificity | not good ≈
Freq, InvDisp,
Freq*InvDisp

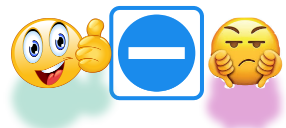
nothing | not good = 0.66



What we found

Basic 

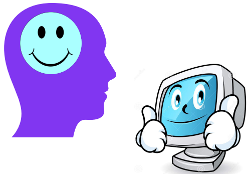
RI+partial:
Range = .557-.638
inverting

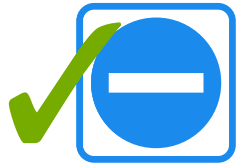
Hard 

RI+partial:
Range = .272-.658
meaning specificity

(And importantly, doing nothing gets you a 0.0.)

Hii, Yuen, & Pearl in prep.





Negation resolution

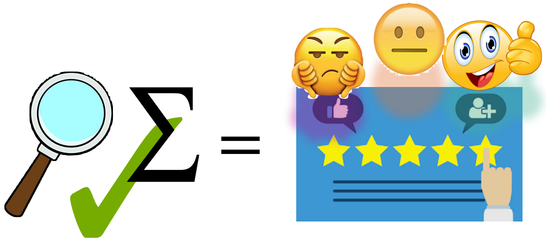
invert — not good = -0.66

asym shift — not good vs. not terrible

antonym — good
bad = -0.5
not good = -0.5

meaning specificity — not good ≈ Freq, InvDisp, Freq*InvDisp

nothing — not good = 0.66

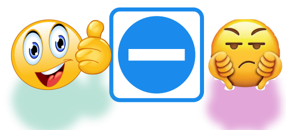


What we found

Basic 

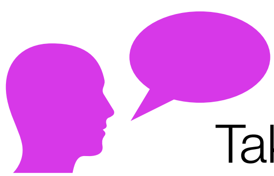
RI+partial:
Range = .557-.638

inverting

Hard 

RI+partial:
Range = .272-.658

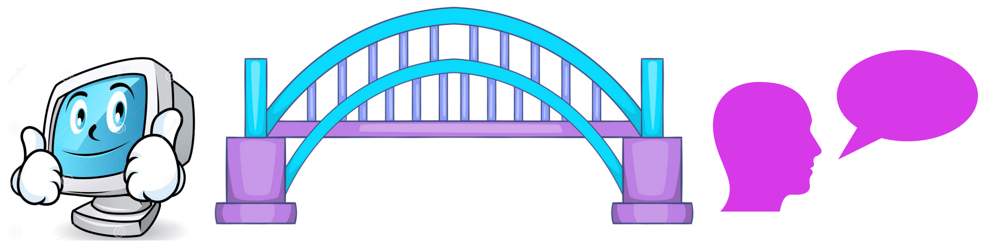
meaning specificity



Takeaway: For hard cases where negation really matters, a meaning specificity approach works the best. But more basic cases can get away with not doing anything particularly clever.

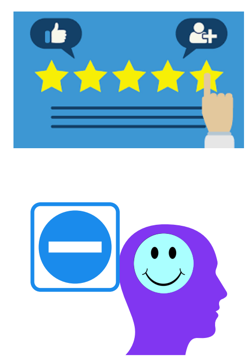
Hii, Yuen, & Pearl in prep.

Big picture: What can be gained by incorporating insights from psychology and linguistics into computational approaches to subtle information extraction

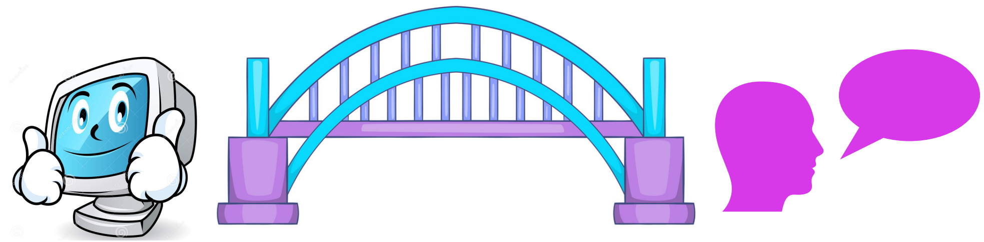


Deception detection

Sentiment analysis



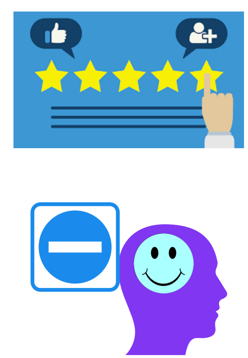
Big picture: What can be gained



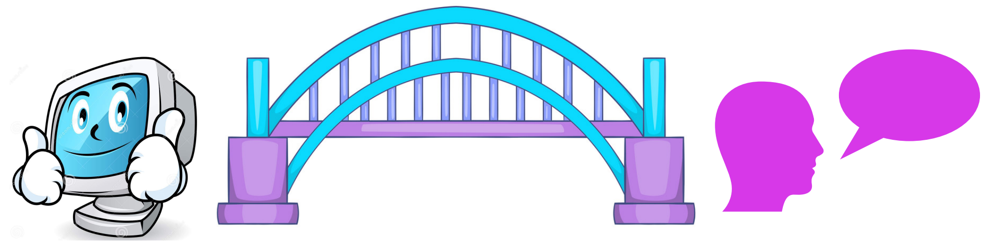
In the hard cases, there can be significant benefit.

Deception detection
across domains

Sentiment analysis
when negation is present

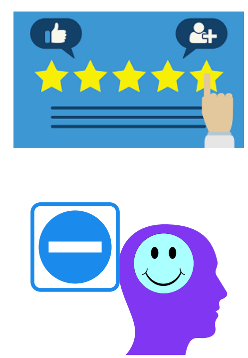


Big picture: What can be gained



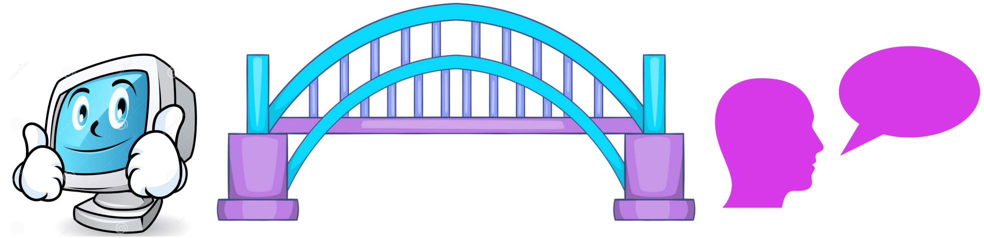
Deception detection
across domains

Sentiment analysis
when negation is present



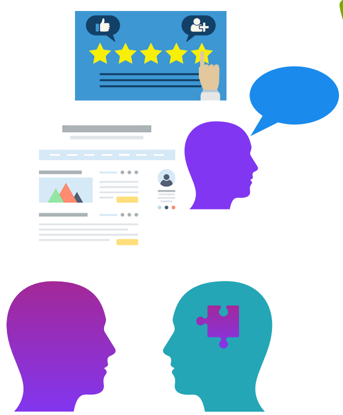
Notably, these areas are ones where
trained or untrained humans can perform well.

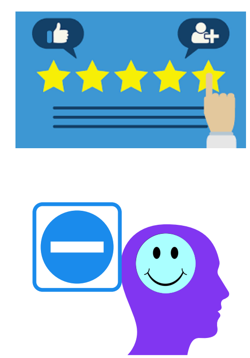
Big picture: What can be gained



Deception detection across domains ✓

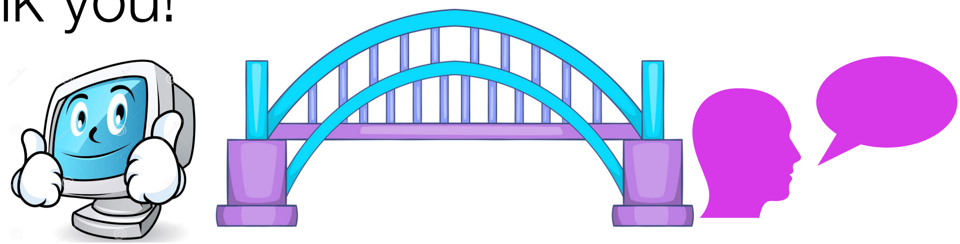
Sentiment analysis when negation is present ✓





Moral of the story: If humans do something well, it may be worthwhile trying to approximate what they're doing when it comes to the features that go into machine learning for handling hard cases.

Thank you!



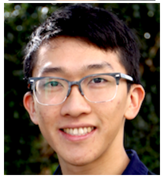
Computation of
Language
Laboratory

UC Irvine

Doreen Hii

Nikolai Vogler

Alan Yuen



Lisa S. Pearl
Professor
Department of Language Science
Department of Cognitive Sciences
SSPB 2219
University of California, Irvine
lpearl@uci.edu