Sentiment lexicons

Chris Potts

Linguist 287 / CS 424P: Extracting Social Meaning and Sentiment, Fall 2010

Sep 21

1 Overview

Goals

- Gain new insights into how sentiment is expressed lexically.
- Begin developing resources that are useful for higher-level classification tasks (phrases, sentences, documents, websites, ...).
- Explore different philosophies for how to build large sentiment lexicons:
  - what to classify (strings (with parts-of-speech) (in context) ...)?
  - which sentiment categories?
  - comparatively small hand-built resources or comparatively large wild ones (or both)?

Plan

§3 Hand annotated/compiled lexicons: Harvard Inquirer, WordNet(s), Micro-WNOp.

§4 A few notes on assessment: accuracy, precision, recall, and friends.

§5 WordNet-based approaches, with and without scores.

§6 Distributional approaches using less-structured corpora.

§7 Summary assessment of theories reviewed.

Code: `http://code.google.com/p/linguist287/source/browse/#svn/trunk/lexicons`

2 Linguistic and psycholinguistic intuitions

What are we classifying (organizing, characterizing)?

(1) gross

(2) (gross, Adj)

(3) (gross, Noun)

(4) (gross, Verb)

(5) gross out

(6) GROSS!!!

(7) gross {soup/amount}

(8) the soup was gross — 1 star!

(9) the slasher movie was gross — 5 stars!

(10) What a {pleasure/disappointment/day}!

(11) {This/That} Kissinger is really something!

(12) Susie's work is good.
(But not great, superb?)

(13) Damn! (impressed by a friend's juggling)

(14) Damn! (you lost at ping-pong)

(15) Damn! (as said by Tony Soprano)

(16) Damn! (as said by Obama)

3 Human annotated/compiled lexicons

3.1 Harvard Inquirer (Stone et al. 1966)

Download: `http://www.wjh.harvard.edu/~inquirer/spreadsheet_guide.htm`
Documentation: `http://www.wjh.harvard.edu/~inquirer/homecat.htm`

	Entry	Positiv	Negativ	Hostile	... (184 classes)	Othtags	Defined
1	A					DET ART	article: Indefinite singular article–some or any one
2	ABANDON		Negativ			SUPV	
3	ABANDONMENT		Negativ			Noun	
4	ABATE		Negativ			SUPV	
5	ABATEMENT					Noun	
⋮							
35	ABSENT#1		Negativ			Modif	
36	ABSENT#2					SUPV	
⋮							
11788	ZONE					Noun	

Table 1: Harvard Inquirer fragment. '*#n*' differentiates senses. 5,394 words have definitions. Binary category values: 'Yes' = category name; 'No' = blank.

Category	Words
Positiv	1,915
Negativ	2,291
Hostile	833
Strong	1,902
Weak	755
Active	2,045
Passive	911

(a) A few prominent sentiment categories

Positiv	Negativ
Pstv	Ngtv
Strong	Weak
Active	Passive
Pleasur	Pain
Ovrst	Undrst
Yes	No

(b) Some sentiment oppositions. Pstv and Ngtv are old versions of Positiv and Negativ. Ovrst and Undrst are over- and under-stated.

Categories	Correlation
{RspOth, RspTot}	0.85
{WltOth, WltTot}	0.84
{NUMB, CARD}	0.84
⋮	
{Negativ, Hostile}	0.48
⋮	
{Strong, Power}	0.34
⋮	
{Active, Passive}	-0.13
{Positiv, Ngtv}	-0.15
{Negativ, Pstv}	-0.15
{Positiv, Negativ}	-0.22

(c) Some correlations between categories: highest and lowest with selections in between.

Table 2: A high-level look at some of the Inquirer categories.

3.2 WordNet (Miller 1995; Fellbaum 1998)

Download/documentation: `http://wordnet.princeton.edu/`
Web interface: `http://wordnetweb.princeton.edu/perl/webwn`
APIs: `http://wordnet.princeton.edu/wordnet/related-projects/`

- *idle*, a
  - *Synset*: idle·a·01
    Definition: not in action or at work
    Examples : {an idle laborer, idle drifters, the idle rich, an idle mind}
    Also sees: {ineffective·a·01, unemployed·a·01}
    Similar tos: {unengaged·s·01, . . . , bone-idle·s·01}
    . . .
    Lemma: idle·a·01·idle
    · *Antonyms*: {busy·a·01·busy}
    · *Derivationally related forms*: {faineance·n·01·idleness}
    · *Pertainyms*: { }
    · . . .
  - *Synset*: baseless·s·01
    Definition: without a basis in reason or fact
    Examples: {baseless gossip, unfounded suspicions, . . . , unwarranted jealousy}
    Also sees: { }
    Similar tos: {unsupported·a·01}
    . . .
    Lemma: baseless·s·01·baseless
    · *Antonyms*: { }
    · *Derivationally related forms*: { }
    · *Pertainyms*: { }
    · . . .
    Lemma: baseless·s·01·groundless
    · . . .
  - . . .

Table 3: Example — from strings to lexical structure. Fields marked with the empty set happen to be empty for this example but can be filled

In WordNet 3.0 (Miller 2009), the string *idle* with category a has 7 synsets. Collectively, those synsets have 18 lemmas. Each synset and each lemma can reach other synsets (lemmas) via relations like those exemplified above . . . (see sec. 5.1).

OpenThesaurus (German): `http://www.openthesaurus.de/`
More globally: `http://www.globalwordnet.org/`
(with free downloads for at least Arabic, Danish, French, Hindi, Russian, Tamil)

3.3 Micro-WNOp (Cerini et al. 2007)

Documentation and download: `http://www.unipv.it/wnop`

	Pos	Neg	Synset
1	1	0	true·a·2 real·a·4
2	1	0	illustrious·a·1 famous·a·1 ...
3	0.5	0	real·a·6 tangible·a·2
4	0.25	0	existent·a·2 real·a·1
5	0.125	0.125	real·a·2
⋮			
110	0	0	demand·v·6

(a) 'Common': Five evaluators working together, 110 synsets.

	Evaluator 1		Evaluator 2		Evaluator 3		
	Pos1	Neg1	Pos2	Neg2	Pos3	Neg3	Synset
1	1	0	1	0	1	0	good·a·15 well·a·2
2	1	0	1	0	0.75	0	sweet-smelling·a·1 perfumed·a·2 ...
3	1	0	1	0	1	0	good·a·23 unspoilt·a·1 unspoiled·a·1
4	0.5	0	0.25	0	0.25	0	hot·a·16
⋮							
496	0.5	0	1	0	0.5	0	heal·v·3 bring_around·v·2 cure·v·1

(b) 'Group 1': Three evaluators working separately, 496 synsets. Complete agreement on 197 (40%). Polarity agreement (sign(Pos1−Neg1) = sign(Pos2−Neg2) = sign(Pos3−Neg3)) on 387 (78%).

	Evaluator 1		Evaluator 2		
	Pos1	Neg1	Pos2	Neg2	Synset
1	0	1	0	1	forlorn·a·1 godforsaken·a·2 lorn·a·1 desolate·a·2
2	0	1	0	1	rotten·a·2
3	1	0	1	0	intimate·a·2 cozy·a·2 informal·a·4
4	0	0	0	0	federal·a·1
⋮					
499	0	0	0	0	term·v·1

(c) 'Group 2': Two evaluators working separately, 499 synsets. Complete agreement on 395 (79%). Polarity agreement (sign(Pos1−Neg1) = sign(Pos2−Neg2)) on 471 (90.4%).

Table 4: The three groups of the Micro-WNOp. The strings listed under Synset identify lemmas.

	Pos	Neg	Obj	Total
a (adj)	62	45	50	157
n (noun)	30	58	273	361
r (adv)	4	0	24	28
v (verb)	27	36	93	156
Total	330	139	440	702

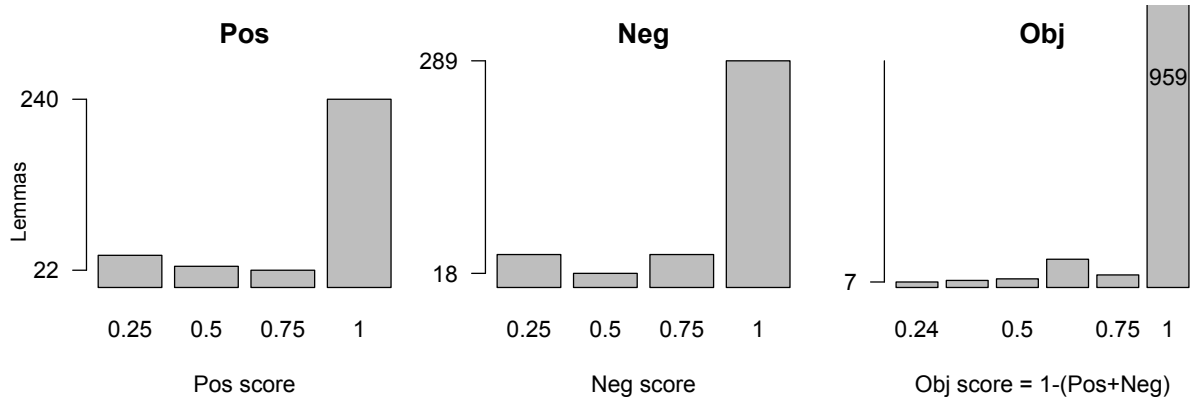
(a) By synset.

	Pos	Neg	Obj	Total
a	151	135	89	375
n	83	144	663	890
r	9	0	53	62
v	87	112	233	432
Total	330	391	1,038	1,759

(b) By lemma.

	Pos	Neg	Obj	Total
a	105	113	62	280
n	66	107	483	656
r	7	0	37	44
v	68	90	178	336
Total	246	310	760	1,316

(c) By ⟨string, tag⟩ pair.



(d) The distribution of scores, by lemma. Objectivity values are $1-(\text{Pos}+\text{Neg})$, so a score of 1 there means that both Positive and Negative were 0. Relatively few of the synsets received non-0/1 ratings.

Table 5: Micro-WNOp limited to the 702 synsets on which all annotators agreed exactly on all values (tab. 5(a)). Pos means that the positive score was higher, Neg that the negative score was higher, and Obj that the two scores were the same (even if they were greater than 0).

3.4 POS tags: relating Harvard Inquirer and WordNet/Micro-WNOp

matches 'noun' (case-insensitive)	↦	n
matches 'verb' or 'supv' (case-insensitive)	↦	v
matches 'modif' (case-insensitive)	↦	a
is 'LY' or matches ' LY' (case-sensitive)	↦	r
matches none of the above		no change

Table 6: Partial mapping from Inquirer categories to WordNet categories. The mapping is partial for two reasons. First, Inquirer includes determiners, prepositions, and other parts of speech not found in WordNet (and hence not represented at all in Micro-WNOp). Second, even setting those classes aside, Inquirer parts-of-speech are highly subcategorized and don't necessarily fall into the four-way WordNet typology.

3.5 Inquirer$_{WN}$, the WordNet subset of Inquirer

	Positive	Negative	Objective	Total
a (adj)	29	11	101	141
n (noun)	638	727	2,906	4,271
r (adv)	52	31	95	178
v (verb)	366	621	1,167	2,154
Total	1,085	1,390	4,269	6,744

Table 7: Inquirer$_{WN}$, the subset of Inquirer that is also in WordNet, at the level of ⟨string, tag⟩ pairs. Inquirer makes sense distinctions, but I am unsure how to align them WordNet lemmas. I am not sure why these numbers are larger than those of Rao and Ravichandran (2009:tab. 1).

3.6 Inquirer and Micro-WNOp

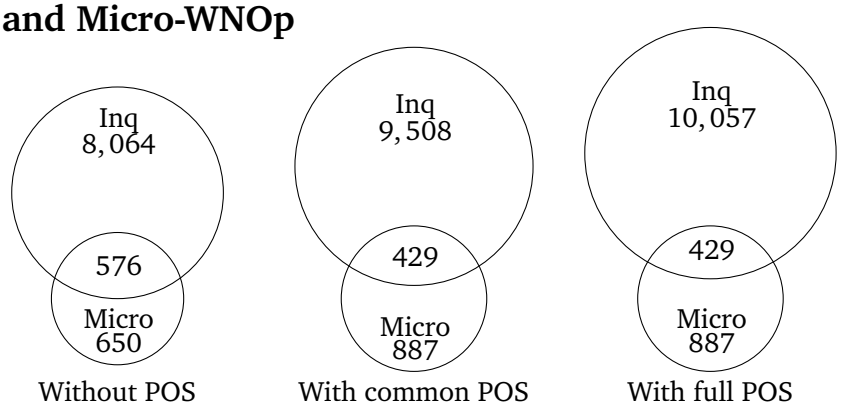


Table 8: ⟨string, tag⟩ pairs in the gold-standard corpora. The differences from left to middle are due to the different ways in which tags disambiguate strings in the two corpora. Inquirer includes many word classes that are not in WordNet, hence not in Micro-WNOp. Some of the non-overlap on both sides is due to formatting differences that could be corrected.

3.7 Thinking ahead to assessment

- Not just strings: words with part-of-speech categories and definitions.
- Multiple *n*-valued sentiment categories, for different values of *n* (mostly 2 or 3).
- Continuous values for constructing scales.

Before looking at specific proposals: how might we use these resources to evaluate proposals?

Reminder: There are 8,640 strings (sense disambiguations removed) in the Inquirer, but there are 13,588,391 unigrams in the Google N-grams vocabulary (Brants and Franz 2006),[1] so we need robust methods for projecting beyond these gold-standard resources.

[1] `http://googleresearch.blogspot.com/2006/08/all-our-n-gram-are-belong-to-you.html`

4 A few notes on assessment

		Predicted Pos	Neg	Obj
	Pos	15	10	100
Observed	Neg	10	15	10
	Obj	10	100	1,000

$$\frac{\text{sum\_diagonal}}{\text{sum\_all\_cells}}$$

(a) Accuracy. Typically appropriate only if the categories are equal in size.

		Predicted Pos	Neg	Obj
	Pos	15	10	100
Observed	Neg	10	15	10
	Obj	10	100	1,000

$$\frac{\text{true\_positive}}{\text{true\_positive} + \text{false\_positive}}$$

(b) Precision: the correct guesses penalized by the number of incorrect guesses. You can often get high precision for a category C by rarely guessing C, but this will ruin your recall.

		Predicted Pos	Neg	Obj
	Pos	15	10	100
Observed	Neg	10	15	10
	Obj	10	100	1,000

$$\frac{\text{true\_positive}}{\text{true\_positive} + \text{false\_negative}}$$

(c) Recall: the correct guesses penalized by the number of missed items. You can often get high recall for a category C by always guessing C, but this will ruin your precision.

Table 9: Confusion matrices. These values should all be interpreted as bounded by $[0, 1]$. (If you are tempted to divide by 0, return 0.)

Definition 1 (F1)**.** $$2\left(\frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}\right)$$

Note: F1 gives equal weight to precision and recall. However, depending on what you're studying, you might value one more than the other.

Definition 2 (Macroaverage)**.** Average precision/recall/F1 over all the classes of interest.

Definition 3 (Microaverage)**.** Sum corresponding cells to create a 2×2 confusion matrix, and calculate precision = recall = F1 in terms of that new matrix.

	Pos	Neg	Obj
Pos	15	10	100
Neg	10	15	10
Obj	10	100	1,000

⇒

	Pos	Other
Pos	15	110
Other	20	1,125

	Neg	Other
Neg	15	20
Other	110	1,125

	Obj	Other
Obj	1,000	110
Other	110	50

⇒

	Yes	No
Yes	1,030	240
No	240	2,300

Note: Before using these summary measures, look at the distribution of your by-category numbers to make sure that you're not losing important structure.

For more: Manning and Schütze 1999:§8.1; Manning et al. 2009:§8.3, 13.6.

5 WordNet-based approaches

5.1 Simple sense/sentiment propagation

Hypothesis: Sentiment is constant throughout regions of lexically related items. Thus, sentiment properties of hand-built seed-sets will be preserved as we follow WordNet relations out from them.

5.1.1 Algorithm (Valitutti et al. 2004; Esuli and Sebastiani 2006; sec. 5.3 for precedents)

WordNetSensePropagate(S, *iter*)

▷ Input
▷ S: a list of synsets. For example: $\langle$ {brilliant·s·01, n·win·01}, {sadly·r·01, gross·a·01} $\rangle$
▷ *iter*: the number of iterations
▷ Output
▷ T: Length(S) × 1+*iter* synset matrix: $\begin{pmatrix} S[1] & \cdots & \textit{iter}\text{-th propagation of } S[1] \\ \vdots & & \vdots \\ S[\text{Length}(S)] & \cdots & \textit{iter}\text{-th propagation of } S[n] \end{pmatrix}$

initialize T: a Length(S) × 1+*iter* matrix such that $T[i][1] = S[i]$ for $1 \leqslant i \leqslant 1+iter$
for $i \leftarrow 1$ **to** *iter*
    for $j \leftarrow 1$ **to** Length(S)
        $newSame \leftarrow$ SamePolarity($T[j][i]$)
        $others \leftarrow \bigcup_{k=1}^{\text{Length}(S)} T[k][i]$ for $k \neq j$ ▷ The other seed-sets in this column.
        $newDiff \leftarrow$ OtherPolarity($others$)
        ▷ For the experiments, I first calculate all the propagation sets and then eliminate their
        ▷ pairwise intersection from each, to ensure no overlap.
        $T[j][i+1] \leftarrow (newSame \cup newDiff)$
return T

SamePolarity(*synsets*)

newsynsets ← { }
for $s \in synsets$ ▷ Synset-level relations.
    newsynsets ← *newsynsets* ∪ {s} ∪ **AlsoSees**(s) ∪ **SimilarTos**(s)
    for *lemma* ∈ **Lemmas**(s) ▷ Lemma-level relations.
        for *altLemma* ∈ (**DerivationallyRelatedForms**(*lemma*) ∪ **Pertainyms**(*lemma*))
            newsynsets ← *newsynsets* ∪ {**Synset**(*altLemma*)}
return *newsynsets*

OtherPolarity(*synsets*)

newsynsets ← { }
for $s \in synsets$
for *lemma* ∈ **Lemmas**(s) ▷ Lemma-level relations.
    for *altLemma* ∈ **Antonyms**(*lemma*)
        newsynsets ← *newsynsets* ∪ {**Synset**(*altLemma*)}
return *newsynsets*

Free variables The seed-sets, the WordNet relations called in SAMEPOLARITY and OTHERPOLARITY, the number of iterations, the decision to remove overlap.

5.1.2 Experiment

Formulate seed-sets by hand, using intuitions. Use the algorithm to derive positive, negative, and objective sets of ⟨string, pos⟩ pairs.

positive	excellent, good, nice, positive, fortunate, correct, superior
negative	nasty, bad, poor, negative, unfortunate, wrong, inferior
objective	administrative, financial, geographic, constitute, analogy, ponder, material, public, department, measurement, visual

Table 10: Seed-sets.

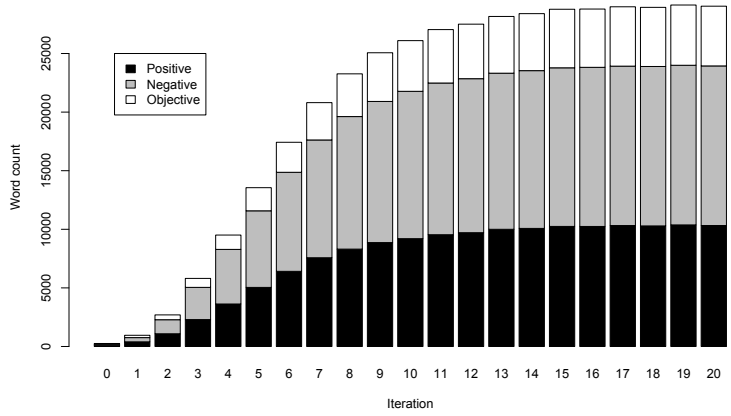


Figure 1: Growth in the size of the sets as the number of iterations grows.

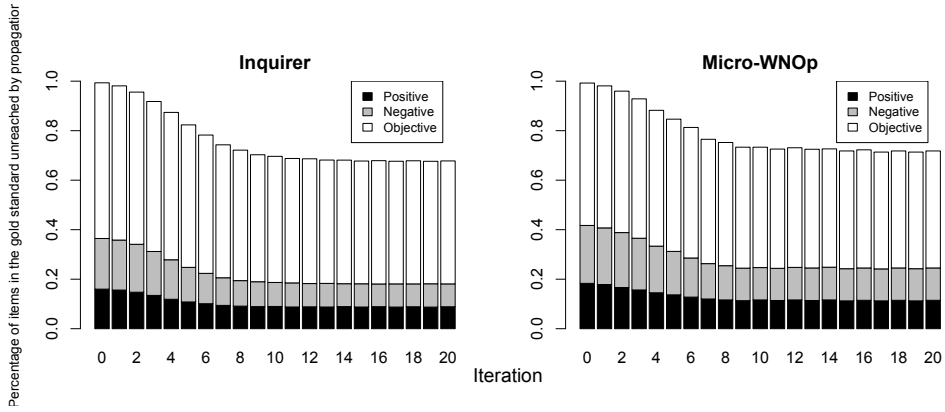


Figure 2: Percentage of items from the gold standard that are unreached by propagation.

5.1.3 Assessment

Definition 4 (Scoring the propagation algorithm)**.**

Derived by propagation	Matching Inquirer category	Micro-WNOp
positive	Positiv	Positive > Negative
negative	Negativ	Positive < Negative
objective	$\text{Inquirer}_{\text{WN}} - (\text{Positiv} \cup \text{Negativ})$	Positive = Negative

Options:

i. Score only ⟨string, pos⟩ pairs that are in the intersection of $\text{Inquirer}_{\text{WN}}$ (Micro-WNOp) with the union of the three derived sets.

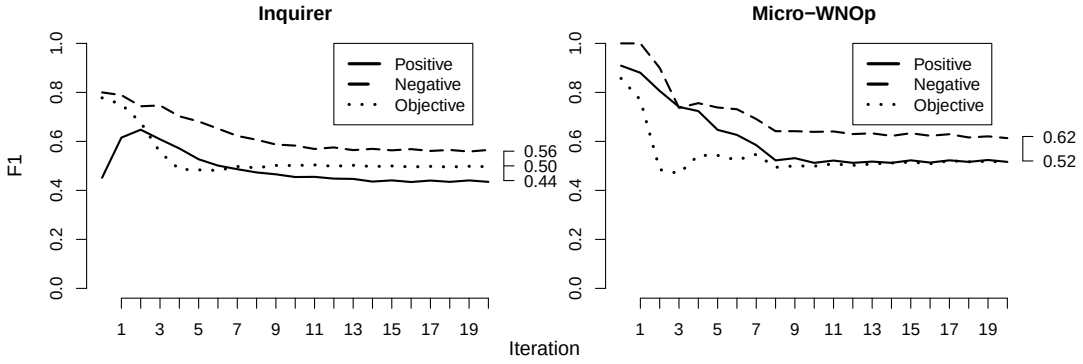
   Drawbacks:

ii. Treat all items not reached by the propagation algorithm as members of objective.

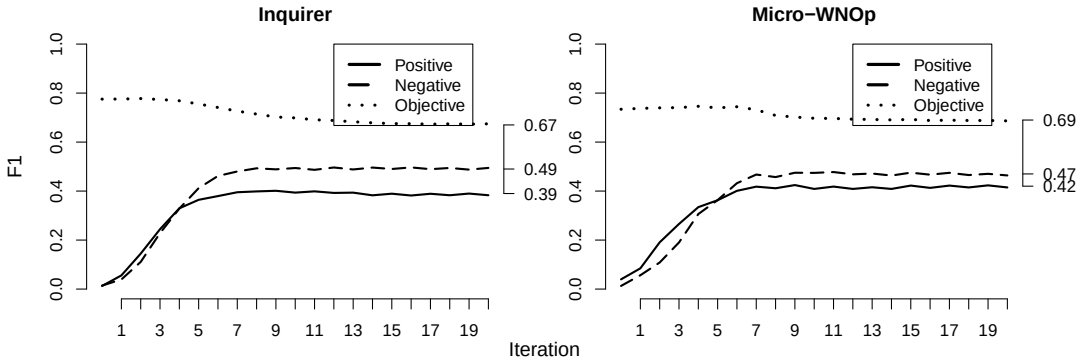
   Drawbacks:

iii. Create a category 'missing' to penalize effectiveness measures for the other categories.

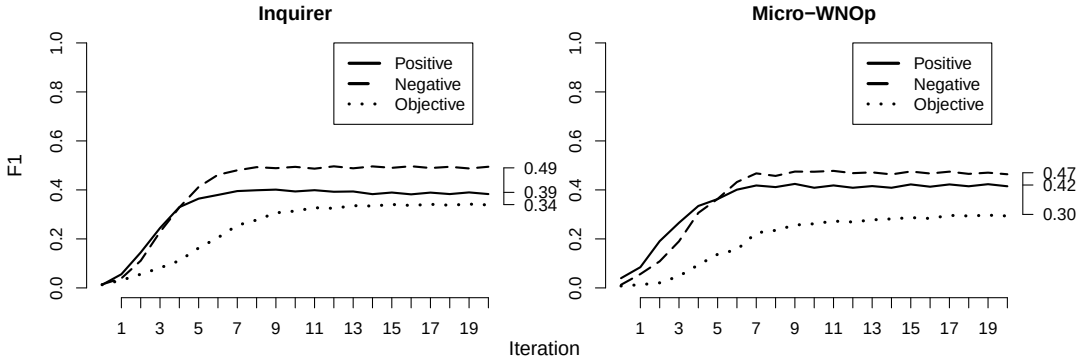
   Drawbacks:



(a) F1 using def. 4, option (i): "Score only ⟨string, pos⟩ pairs that are in the intersection of $\text{Inquirer}_{\text{WN}}$ (Micro-WNOp) with the union of the three derived sets."



(b) F1 using def. 4, option (ii): "Treat all items not reached by the propagation algorithm as members of objective."



(c) F1 using def. 4, option (iii): "Create a category 'missing' to penalize effectiveness measures for the other categories."

Figure 3: Using F1 to assess the propagation algorithm against Inquirer and Micro-WNOp. The three scoring options emphasize different aspects of the algorithm.

5.2 Propagation with continuous values

Hypothesis: As in sec. 5.1, but with the additional claim that a word's strength for property X is connected to (and hence can be estimated from) the number of X-words it is lexically related to.

5.2.1 Algorithm (Blair-Goldensohn et al. 2008, henceforth G08)

Seed-sets	Score vector s_0	Matrix A
P (pos.) N (neg.) M (obj.)	$s_0^i = \begin{cases} +1 \text{ if } w_i \in P \\ -1 \text{ if } w_i \in N \\ 0 \text{ otherwise} \end{cases}$	$a_{i,j} = \begin{cases} 1+\lambda \text{ if } i = j \\ +\lambda \text{ if } w_i \in \text{syn}(w_j) \;\&\; w_i \notin M \\ -\lambda \text{ if } w_i \in \text{ant}(w_j) \;\&\; w_i \notin M \\ 0 \text{ otherwise} \end{cases}$

Repeated $A * s_i$: $A * s_0 = s_1$; $A * s_1 = s_2$; ... sentiment scores from the final vector (and, for each item, change its final sign to its initial sign if the two differ)

Small example

$$\lambda = 0.2$$
$$\text{syn}(\langle \textit{superb}, \text{a}\rangle) = \{\langle \textit{great}, \text{a}\rangle\}$$
$$\text{syn}(\langle \textit{great}, \text{a}\rangle) = \{\langle \textit{superb}, \text{a}\rangle, \langle \textit{good}, \text{a}\rangle\}$$
$$\text{syn}(\langle \textit{good}, \text{a}\rangle) = \{\,\}$$

Seed-sets	Score vector s_0	Matrix A
$P = \{\langle \textit{superb}, \text{a}\rangle\}$ $N = \{\,\}$ $M = \{\,\}$	$\begin{pmatrix} \langle \textit{good}, \text{a}\rangle & 0.0 \\ \langle \textit{great}, \text{a}\rangle & 0.0 \\ \langle \textit{superb}, \text{a}\rangle & 1.0 \end{pmatrix}$	$\begin{pmatrix} & \langle \textit{good}, \text{a}\rangle & \langle \textit{great}, \text{a}\rangle & \langle \textit{superb}, \text{a}\rangle \\ \langle \textit{good}, \text{a}\rangle & 1.2 & 0.2 & 0.0 \\ \langle \textit{great}, \text{a}\rangle & 0.2 & 1.2 & 0.2 \\ \langle \textit{superb}, \text{a}\rangle & 0.0 & 0.0 & 1.2 \end{pmatrix}$

Iteration:

	s_0	s_1	s_2	s_3	s_4	s_5	s_6
$\langle \textit{good}, \text{a}\rangle$	0.00	0.00	0.04	0.14	0.35	0.70	1.28
$\langle \textit{great}, \text{a}\rangle$	0.00	0.20	0.48	0.87	1.42	2.19	3.26
$\langle \textit{superb}, \text{a}\rangle$	1.00	1.20	1.44	1.73	2.07	2.49	2.99

Rescale? G08 propose to rescale scores:

$$s_i \leftarrow \begin{cases} \log(\text{abs}(s_i)) \cdot \text{sign}(s_i) & \text{if } \text{abs}(s_i) > 1 \\ 0 & \text{otherwise} \end{cases}$$

However, in the above example, we would lose the sentiment score for *good* if we stopped before iteration 6. In my experiments, rescaling resulted in dramatically fewer non-0 values.

Free variables The nature of the seed-sets, the number of iterations, the weight λ, whether to rescale, which if any WordNet relations to use beyond syn and ant.

5.2.2 G08 results

In our experiments, the original seed-set contained 20 negative and 47 positive words that were selected by hand to maximize domain coverage, as well as 293 neutral words that largely consist of stop words. [...] Running the algorithm resulted in an expanded sentiment lexicon of 5,705 positive and 6,605 negative words, some of which are shown in Table 2 with their final scores. Adjectives form nearly 90 percent of the induced vocabulary, followed by verbs, nouns and finally adverbs.

Positive	Negative
Good_a (7.73)	Ugly_a (-5.88)
Swell_a (5.55)	Dull_a (-4.98)
Naughty_a (-5.48)	Tasteless_a (-4.38)
Intellectual_a (5.07)	Displace_v (-3.65)
Gorgeous_a (3.52)	Beelzebub_n:(-2.29)
Irreverent_a (3.26)	Bland_a (-1.95)
Angel_n (3.06)	Regrettably_r (-1.63)
Luckily_r (1.68)	Tardily_r (-1.06)

Table 2: Example terms from our induced sentiment lexicon, along with their scores and part-of-speech tags (adjective = a, adverb = r, noun = n, verb = v). The range of scores found by our algorithm is [-7.42,7.73].

G08 do not stop here. They go on to use their lexicon for classification and summarization. We'll return to those portions of the paper in the next two classes.

5.2.3 Our experiment

Micro-WNOp provides the set of seed-sets: we use the entire set of positive, negative, and objective sets as defined in tab. 5(c). The idea is that we will use this algorithm, not to learn new sentiment lexicons, but rather to learn the strength relationships between words of the same polarity.

	Seed-set	After propagation
P	246	1,021
N	310	978
M	760	154,585
Total	1,316	156,584

Table 11: Experimental results. The combined seeds-sets are the entire Micro-WNOp corpus. The total after propagation is the number of lemmas in WordNet 3.0. The gain in the positive and negative categories is not as large as G08 saw. This is almost certainly due to the fact that the Micro-WNOp corpus does not cover as much of the domain as their seed-sets did.

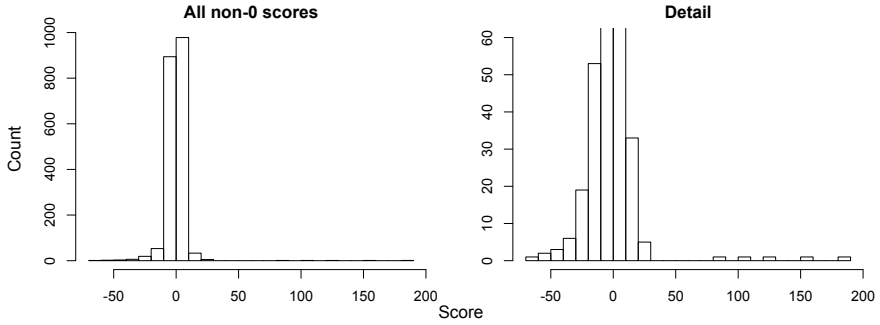


Figure 4: Distribution of non-0 scores in the final lexicon. The vast majority are close to 0. The largest is 185.12 (*valour*) and the smallest is −61.1 (*rophy*, 'street names for flunitrazepan'(!?)).

5.2.4 Assessment

61 examples from Micro-WNOp are missing from the resulting lexicon. This is because Micro-WNOp was created with WordNet 2.0 synsets, but the propagation algorithm was run over WordNet 3.0. For details on relating these for the sake of Micro-WNOp, see Baccianella et al. 2010:§4.

If we study the intersection of Micro-WNOp with the score-propagation results, then the propagation algorithm scores perfectly when it comes to simply classifying items according to their basic polarity (positive, negative, objective). This is as expected; the algorithm is designed to preserve the scores given in s_0.

The more interesting assessment uses the continuous values:

Definition 5 (Scoring the propagation algorithm)**.** We evaluate agreement on the set of all two-membered sets of $\{\langle w_A, x\rangle, \langle w_B, x\rangle\}$, where

- $Micro(\langle w, x\rangle) =$ the larger of the two scores Positive or Negative for w in Micro-WNOp, picking arbitrarily if they are both the same.
- $Goog(\langle w, x\rangle) =$ the score for $\langle w, x\rangle$ from the experiment summarized in sec. 5.2.4.

The gold standard Micro-WNOp and our results agree for all cases on the basic polarity, which means that $Micro(\langle w, x\rangle)$ will never pick the positive score as the higher one when $Goog(\langle w, x\rangle)$ delivers a negative score, nor the reverse. Thus, we can compare absolute values:

The experimental prediction is correct if, for $R \in \{<, >, =\}$

$$\begin{array}{rcll} Micro(\langle w_A, x\rangle) & R & Micro(\langle w_B, x\rangle) & \text{and} \\ \text{abs}(Goog(\langle w_A, x\rangle)) & R & \text{abs}(Goog(\langle w_B, x\rangle)) & \end{array}$$

else the experimental prediction is incorrect.

	stronger	weaker	equal
stronger	7,575	5,532	24,301
weaker	6,345	6,599	22,150
equal	17,516	19,722	215,594

(a) Confusion matrix for the scores experiment.

	Precision	Recall	F1
stronger	0.24	0.20	0.22
weaker	0.21	0.19	0.20
equal	0.82	0.85	0.84

(b) Effectiveness

just stronger/weaker	54%
stronger/weaker/equal	71%

(c) Accuracy

Table 12: Assessment of scores

5.3 Similar approaches and ideas

Hu and Liu 2004 One of the first to use this kind of propagation for sentiment. Their algorithm is very similar to the one in sec. 5.1.1, though its inputs and outputs are strings rather than synsets and it restricts attention to adjectives (positive and negative).

Kim and Hovy 2006 Sense propagation over WordNet from small seed-sets, dealing with overlap via maximum likelihood, which also delivers sentiment scores/rankings for words (strings).

Godbole et al. 2007 WordNet propagation relative to general weblog topic categories, with scores based on distance from seed-set words, taking into account the dangers of mixed paths.

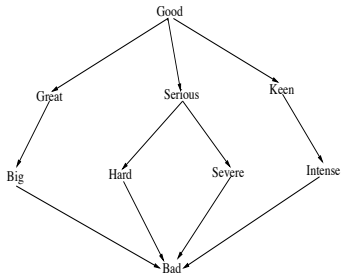


Fig. 1: *Four ways to get from bad to good in three hops*

Andreevskaia and Bergler 2006 Propagation using a limited set of basic WordNet relations, but with creative use of the glosses to further expand the lexicon. The word lists of Hatzivassiloglou and McKeown 1997 (see sec. 6.2) are used to formulate the seed-sets, and the Harvard Inquirer is the gold standard.

	Average run size		Average % correct	
	# of adj	StDev	%	StDev
PASS 1 (WN Relations)	103	29	78.0%	10.5%
PASS 2 (WN Glosses)	630	377	64.5%	10.8%
PASS 3 (POS clean-up)	435	291	71.2%	11.0%

Table 2: Performance statistics on STEP runs.

Rao and Ravichandran 2009 Label propagation using WordNet supplemented with other hand-build resources. Evaluation is partly with Harvard Inquirer. The approach is extended to Hindi and French with strong results.

SentiWordNet (Esuli and Sebastiani 2006; Baccianella et al. 2010) Built by first propagating out from the positive and negative seed-sets in tab. 10 and then building a classifier from the resulting synsets. The features in the classifier are not the synsets themselves, but rather the synsets of their definitions, as provided by the Princeton Annotated Gloss Corpus:[2]

very	good	; of the highest	quality	; “	made	an	excellent	speech	” [. . .]
very·3	good·1	high·3	quality·1		make·2		excellent·3	speech·1	
very·4	good·3	high·4	quality·3		made·3				
	good·4								

`http://sentiwordnet.isti.cnr.it/`

[2] `http://wordnet.princeton.edu/glosstag.shtml`

6 Distributional approaches

6.1 Web-based propagation

Hypothesis: The basic hypothesis of sec. 5.2 is correct, and we can move past the confines of WordNet by relating distributional similarity to sentiment.

6.1.1 Algorithm (Velikovich et al. 2010)

Input: $G = (V, E), w_{ij} \in [0, 1]$,
$P, N, \gamma \in \mathbb{R}, T \in \mathbb{N}$
Output: $\mathbf{pol} \in \mathbb{R}^{|V|}$
Initialize: $\mathbf{pol}_i, \mathbf{pol}_i^+, \mathbf{pol}_i^- = 0$, for all i
$\mathbf{pol}_i^+ = 1.0$ for all $v_i \in P$ and
$\mathbf{pol}_i^- = 1.0$ for all $v_i \in N$

1. set $\alpha_{ii} = 1$, and $\alpha_{ij} = 0$ for all $i \neq j$
2. for $v_i \in P$
3. $F = \{v_i\}$
4. for $t : 1 \ldots T$
5. for $(v_k, v_j) \in E$ such that $v_k \in F$
6. $\alpha_{ij} = \max\{\alpha_{ij},\ \alpha_{ik} \cdot w_{kj}\}$
$F = F \cup \{v_j\}$
7. for $v_j \in V$
8. $\mathbf{pol}_j^+ = \sum_{v_i \in P} \alpha_{ij}$
9. Repeat steps 1-8 using N to compute $\mathbf{pol}^-$
10. $\beta = \sum_i \mathbf{pol}_i^+ / \sum_i \mathbf{pol}_i^-$
11. $\mathbf{pol}_i = \mathbf{pol}_i^+ - \beta \mathbf{pol}_i^-$, for all i
12. if $|\mathbf{pol}_i| < \gamma$ then $\mathbf{pol}_i = 0.0$, for all i

Figure 1: Graph Propagation Algorithm.

- G is a cosine similarity graph
- P and N are seed-sets
- γ is a threshold: sentiment scores below it will be rounded to 0
- T is the number of iterations.

Definition 6 (Cosine similarity)**.** Let A and B be vectors of co-occurrences counts for words w_A and w_B defined over a fixed, ordered vocabulary (rows in tab. 13(b)) of length n. Their cosine similarity:

$$\text{cosim}(w_A, w_B) \stackrel{def}{=} \frac{\sum_{i=1}^{n}(A_i * B_i)}{\sqrt{\sum_{i=1}^{n} A_i^2} * \sqrt{\sum_{i=1}^{n} B_i^2}}$$

6.1.2 Example

superb amazing
superb movie
superb movie
superb movie
superb movie
superb movie
amazing movie
amazing movie
cool superb

(a) Corpus.

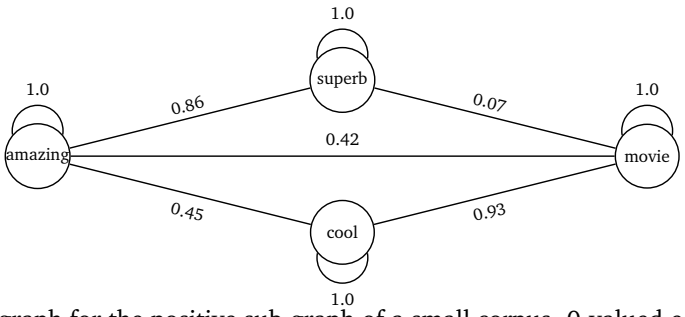
	amazing	cool	movie	superb
amazing	0	0	2	1
cool	0	0	0	1
movie	2	0	0	5
superb	1	1	5	0

(b) Co-occurrence matrix.

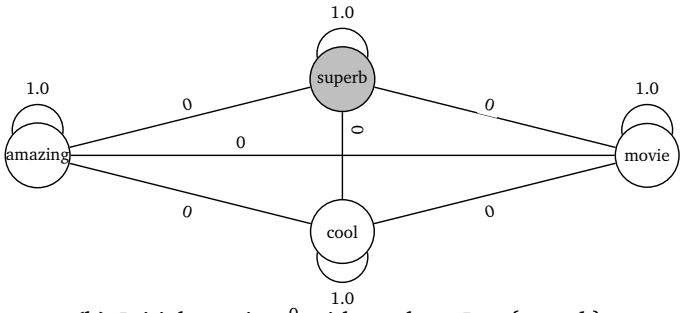
	amazing	cool	movie	superb
amazing	1.0	0.45	0.42	0.86
cool	0.45	1.0	0.93	0.0
movie	0.42	0.93	1.0	0.07
superb	0.86	0.0	0.07	1.0

(c) Cosine similarity matrix.

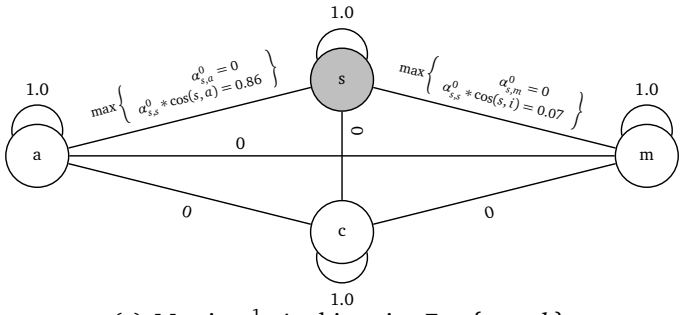
Table 13: Example corpus and similarity measures. Fig. 5 continues the example



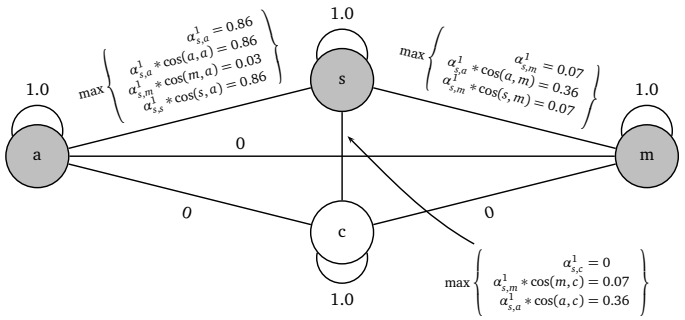
(a) Cosine similarity graph for the positive sub-graph of a small corpus. 0-valued edges not represented.



(b) Initial matrix α^0 with seed-set $P = \{superb\}$.



(c) Matrix α^1. At this point $F = \{superb\}$.



(d) Matrix α^2. F has expanded to all the nodes in fig. 5(a) reachable from *superb* in one step. *cool* and *movie* benefit from their closeness to *amazing*, which is close to the seed-word. Your values might be different if you modify α in place as in line 6 of the algorithm, which involves some α^n values in α^n calculations.

Figure 5: The inner loop (lines 3-6) of the web propagation algorithm with $P\{superb\}$.

6.1.3 Velikovich et al.'s (2010) lexicon

"For this study, we used an English graph where the node set V was based on all n-grams up to length 10 extracted from 4 billion web pages. This list was filtered to 20 million candidate phrases using a number of heuristics including frequency and mutual information of word boundaries. A context vector for each candidate phrase was then constructed based on a window of size six aggregated over all mentions of the phrase in the 4 billion documents. The edge set E was constructed by first, for each potential edge (v_i, v_j), computing the cosine similarity value between context vectors. All edges (v_i, v_j) were then discarded if they were not one of the 25 highest weighted edges adjacent to either node v_i or v_j."

Phrase length	1	2	3	4	5	6	7	8	9
# of phrases	37,449	108,631	27,822	3,489	598	71	29	4	1

Table 14: Sizes for the lexicons that Velikovich et al. (2010) obtained from their Web graph. This method can assign scores to quasi-phrase-level things. The gains diminish sharply after 4. If you want to try your own experiments you might use the 4- or 5-gram data from the Google N-grams corpus (Brants and Franz 2006) to build your graph.

				Recall wrt other lexicons		
	All Phrases	Pos. Phrases	Neg. Phrases	Wilson et al.	WordNet LP	Web GP
Wilson et al.	7,628	2,718	4,910	100%	37%	2%
WordNet LP	12,310	5,705	6,605	21%	100%	3%
Web GP	178,104	90,337	87,767	70%	48%	100%

Table 1: Lexicon statistics. Wilson et al. is the lexicon used in Wilson et al. (2005), WordNet LP is the lexicon constructed by Blair-Goldensohn et al. (2008) that uses label propagation algorithms over a graph constructed through WordNet, and Web GP is the web-derived lexicon from this study.

POSITIVE PHRASES			**NEGATIVE PHRASES**		
Typical	**Multiword expressions**	**Spelling variations**	**Typical**	**Multiword expressions**	**Vulgarity**
cute	once in a life time	loveable	dirty	run of the mill	fucking stupid
fabulous	state - of - the - art	nicee	repulsive	out of touch	fucked up
cuddly	fail - safe operation	niice	crappy	over the hill	complete bullshit
plucky	just what the doctor ordered	cooool	sucky	flash in the pan	shitty
ravishing	out of this world	coooool	subpar	bumps in the road	half assed
spunky	top of the line	koool	horrendous	foaming at the mouth	jackass
enchanting	melt in your mouth	kewl	miserable	dime a dozen	piece of shit
precious	snug as a bug	cozy	lousy	pie - in - the - sky	son of a bitch
charming	out of the box	cosy	abysmal	sick to my stomach	sonofabitch
stupendous	more good than bad	sikk	wretched	pain in my ass	sonuvabitch

Table 3: Example positive and negative phrases from web lexicon.

Table 15: Additional information about the Velikovich et al. (2010) lexicon. 'Wilson et al.' refers to Wilson et al. (2005), which uses a largely hand-annotated corpus that goes beyond unigrams. WordNet LP is the approach of sec. 5.2. The recall numbers provide information about the extent to which the lexicons overlap.

6.2 Coordination and related features

Hypothesis: The morphosyntactic properties of coordination provide reliable information about adjectival oppositions and lexical polarities (Hatzivassiloglou and McKeown 1997, henceforth H&M97).

6.2.1 Data

H&M97 "For our experiments, we use the 21 million word 1987 Wall Street Journal corpus, automatically annotated with part-of-speech tags using the PARTS tagger (Church, 1988)."

Here 105,423 coordinated adjectives drawn from user-supplied reviews at IMDB.com and POS-tagged using the Stanford tagger. Homework 1, problem 3, links to the data file (in CSV format) and asks you to explore it yourself looking for patterns.

`http://stanford.edu/class/cs424p/assignments/cs424p-assign01.html`

Thus, I focus on the basic procedure rather than on feature selection. I was too lazy to get at the syntactic features H&M97 use, but my data include new features about the contexts of utterance.

6.2.2 Ground truth

H&M97 The authors classified the adjectives in their corpus appearing 20 or more times, removing those that were problematically ambiguous and checking subsets with other annotators too. The resulting set contained 1,336 adjectives (657 positive, 679 negative).

Here Coordinations in which both members are classified as Positiv or Negativ in the Inquirer. 1,342 adjectives (620 positive, 722 negative).

6.2.3 Baseline

H&M97 "since 77.84% of all links from conjunctions indicate same orientation, we can achieve this level of performance by always guessing that a link is of the same-orientation type."

	Same polarity	Different polarity
Their data	78%	22%
Our data	76% (72,003)	24% (22,784)

Table 16: Baseline accuracy. Always guessing 'same polarity' results in a recall value of 0 for 'different polarity'.

terr.ible and point.less
love.ly and compassionate
warm and gener.ous
silly and hard
cruel but sympathet.ic
trag.ic and somber
point.less and silly
hand.some and enjoy.able
help.less and ignorant
wrong or weird
dumb and dread.ful
not right and wise
excellent and perfect
bizarre but good
criminal and destruct.ive
dead or alive
health.y and robust
significant and controvers.ial
evil and cynic.al
religi.ous and hero.ic
master.ful and meaning.ful
styl.ish and true
rich and wonder.ful
harm.less and witt.y
strict but realist.ic
decent and not bad
gruff and gentle
cred.ible and worth.y
attract.ive and real
moralist.ic or sympathet.ic
cruel and autocratic
sad and un.necessary
ignorant and not will.ing
obscure or pretenti.ous
mature and precise
in.significant and point.less
realist.ic or blood.y
authent.ic and potent
tire.some or lame
creat.ive and dark
real and majest.ic
dumb and scar.ed
clean and crazy
naive and courage.ous
poor and poor
un.attract.ive but gentle
sad and triumphant
meaning.ful and sens.ible
cynic.al and diabolic.al
casual and ridicul.ous
faith.ful but dead
not true and ill
rampant and hide.ous
malignant and venom.ous
humble and real
erroneous and dead.ly
nice and subtle
vain and ridicul.ous
thought.less and pathet.ic
vivid and creat.ive
un.comfort.able and nerv.ous
effect.ive and grotesque
harsh but comic
grace.ful and talent.ed

6.2.4 The *but* rule

H&M97 "conjunctions using *but* exhibit the opposite pattern, usually involving adjectives of different orientations. Thus, a revised but still simple rule predicts a different-orientation link if the two adjectives have been seen in a *but* conjunction, and a same-orientation link otherwise, assuming the two adjectives were seen connected by at least one conjunction."

	Accuracy
Their data	81.81%
Our data	74%

(a) Accuracy. In addition, I picked randomly balanced subgroups with 20,000 coordination tokens in each category. The baseline is then 50%, but the average accuracy of the *but*-rule over 10 runs was 67%.

	Predicted	
	Different	Same
Different	12,201	16,170
Same	10,787	66,265

(b) Confusion matrix.

	Precision	Recall	F1
Different	53%	43%	47%
Same	47%	86%	61%

(c) Effectiveness (our data)

Table 17: Assessing the *but*-rule.

6.2.5 Logistic regression

	Coefficient	Estimate	Standard error
0	Intercept	1.56	0.01
1	Coord=but *the coordinator is but*	−0.85	0.04
2	*but*-rule *as described in sec. 6.2.4*	−1.37	0.02
3	Conj1Negated *the first conjunct has an external negation*	1.04	0.15
4	Conj1Negated *the second conjunct has an external negation*	0.62	0.07
5	StemsMatch *at least one adj is complex and the stems match*	−4.05	0.37
6	Conj1Negated:Conj2Negated *interaction: both conjuncts negated externally*	−3.35	0.29

Table 18: Coefficient estimates with standard errors. All of the features are binary: 1 if the description is true, else 0. A negative coefficent indicates that the feature biases in favor of different orientation for the adjectives, whereas a positive coefficient indicates a bias for same orientation. There are clearly other features and interactions that could be tried — data homework 1, problem 3, asks you to start exploring other options.

$$\Pr(\text{same}) = \text{logit}^{-1}\begin{pmatrix} 1.56 & + \\ -0.85 * 1 & + \\ -1.37 * 1 & + \\ 1.04 * 1 & + \\ 0.62 * 1 & + \\ -4.05 * 0 & + \\ -3.35 * (1 * 1) & + \end{pmatrix} = 0.09$$

(a) *not out.stand.ing but not aw.ful*

$$\Pr(\text{same}) = \text{logit}^{-1}\begin{pmatrix} 1.51 & + \\ -0.85 * 0 & + \\ 1.37 * 0 & + \\ 1.04 * 0 & + \\ 0.62 * 0 & + \\ -4.05 * 0 & + \\ -3.35 * (0 * 0) & + \end{pmatrix} = 0.82$$

(b) *healthy and robust*

Table 19: Examples

Definition 7 (Logistic regression predictions)**.** The prediction is correct if the polarity differs and $\Pr(\text{same}) \leqslant 0.5$ or if the polarity is the same and $\Pr(\text{same}) > 0.5$

		Predicted	
		Different	Same
Empirical	Different	3,082	19,702
	Same	1,627	70,376

(a) Confusion matrix.

	Precision	Recall	F1
Different	65%	14%	22%
Same	78%	98%	87%

(b) Effectiveness.

Table 20: Interim results: The above model correctly classifies 77% of the samples, using def. 7.

6.2.6 Dissimilarity measure

Definition 8 (Dissimilarity)**.** d is a partial function from $(\textit{words} \times \textit{words})$ into $[0, 1]$

$$d(x, y) = \begin{cases} 1.0 - \text{ the median fitted value of } (x, y) \text{ if available, else} \\ 1.0 - \text{ the median fitted value of } (y, x) \text{ if available, else} \\ 0.5 \end{cases}$$

Note: The fitted model will deliver multiple values for a adjective pairs that appear in different environments. As far as I can tell, H&M97 don't specify how to move from these values to a unique one. I chose the median because it seems to have a good chance of being in the right area in cases where there are multiple values, some small and some large.

Note: This isn't a distance measure because it fails to satisfy the triangle inequality (the sum of the lengths of any two sides of a triangle is greater than the length of the third). As a result, it isn't appropriate for use in the objective functions for centroid clustering algorithms like k-means (Hatzivassiloglou and McKeown 1993).

6.2.7 Clustering via the Exchange Method (Späth 1977, 1980)

EXCHANGEMETHOD($words, d, k$)

▷ Input
▷ *words*: an ordered list of strings
▷ d: a partial function from ($words \times words$) into $[0, 1]$, as in def. 8
▷ k: the number of classes
▷ Output
▷ C: a minimal distance k-celled indexed partition of *words*

```
1  C ← RandomPartition(words, k)
2  objVal ← ObjFunc(C, d)
3  while True
4          C, newVal ← Exchange(C, objVal, words, d)
5          if newVal < objVal
6            then objVal ← newVal
                   ▷ Termination: the previous run through words produced no improvements.
7            else return C
```

EXCHANGE($C, objVal, words, d$)

▷ Notation: If $C = \{c_1 \dots c_n\}$, then $C^{w:i \mapsto j} = \{c_1 \dots (c_i - \{w\}) \dots (c_j \cup \{w\}) \dots c_n\}$

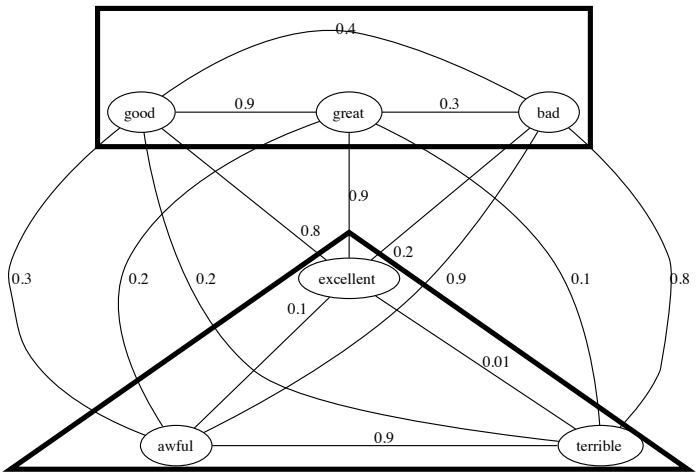
```
1  for w ∈ words
2          c_i ← ClassOf(w, C)
3          initialize candidates    ▷ Dictionary storing viable cell–value pairs.
4          for c_j ∈ (C − c_i):
5                  newVal ← ObjFunc(C^{w:i↦j}, d)
6                  if newVal < objVal
7                    then candidates[c] ← newVal
8          if candidates ≠ ∅        ▷ If any exchanges would be improvements,
9            then c_j, objVal ← ArgminWithMin(candidates)    ▷ then pick the biggest improvement.
10                 C ← C^{w:i↦j}
11 return C, objVal
```

OBJFUNC(C, d)

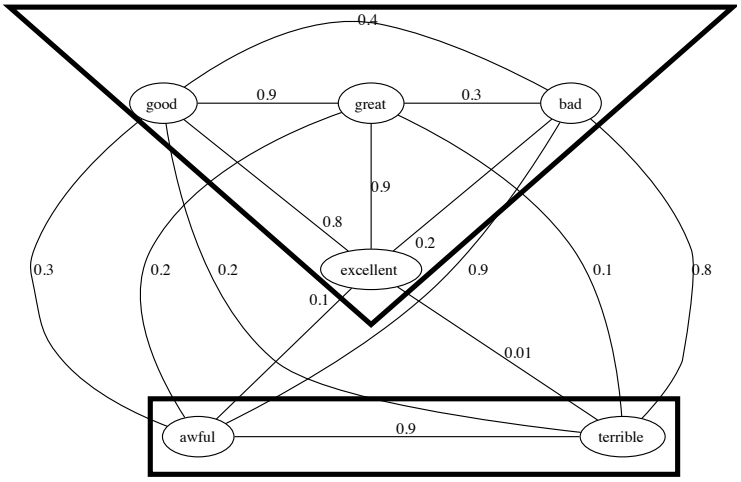
$$\textbf{return} \sum_{c \in C} \left(\frac{1}{|c|} \sum_{x,y \in c, x \neq y} d(x, y) \right)$$

▷ In the inner summation, you needn't iterate over $(c \times c)$, but
▷ rather only over the two-membered subsets of c. In addition,
▷ pairs without values in d can be treated as a group.

Cluster correction? H&M97 take the additional step of 'cluster correction' after the exchange method has converged. In my experiments, this always led to worse results, presumably because it has the potential to raise the objective value. It might be worth exploring a version where such correction is followed by additional iterations of EXCHANGE, since this could pop us out of a local minimum.



$$\frac{(1-0.4)+(1-0.9)+(1-0.3)}{3}+\frac{(1-0.1)+(1-0.9)+(1-0.01)}{3}=1.13$$



$$\frac{(1-0.4)+(1-0.9)+(1-0.3)+(1-0.8)+(1-0.9)+(1-0.2)}{4}+\frac{1-0.9}{2}=0.675$$

Figure 6: Exchange of *excellent*.

6.2.8 Cluster labeling

H&M97 "The clustering algorithm separates each component of the graph into two groups of adjectives, but does not actually label the adjectives as positive or negative. [...] We compute the average frequency of the words in each group, expecting the group with higher average frequency to contain the positive terms."

Here The frequencies for unigrams in the Google N-grams corpus (Brants and Franz 2006).

Free variables The choice of clustering algorithms, the inherent randomness in the initial partition, the order in which the words are gone through (which can lead to different local optima), the nature of the dictionary d, which is determined by the features in the logistic regression, the frequency measure that determines class labels.

6.2.9 Assessment

α	Number of adjectives in test set ($\|A_\alpha\|$)	Number of links in test set ($\|L_\alpha\|$)	Average number of links for each adjective	Accuracy	Ratio of average group frequencies
2	730	2,568	7.04	78.08%	1.8699
3	516	2,159	8.37	82.56%	1.9235
4	369	1,742	9.44	87.26%	1.3486
5	236	1,238	10.49	92.37%	1.4040

Table 3: Evaluation of the adjective classification and labeling methods.

Table 21: H&M97's results.

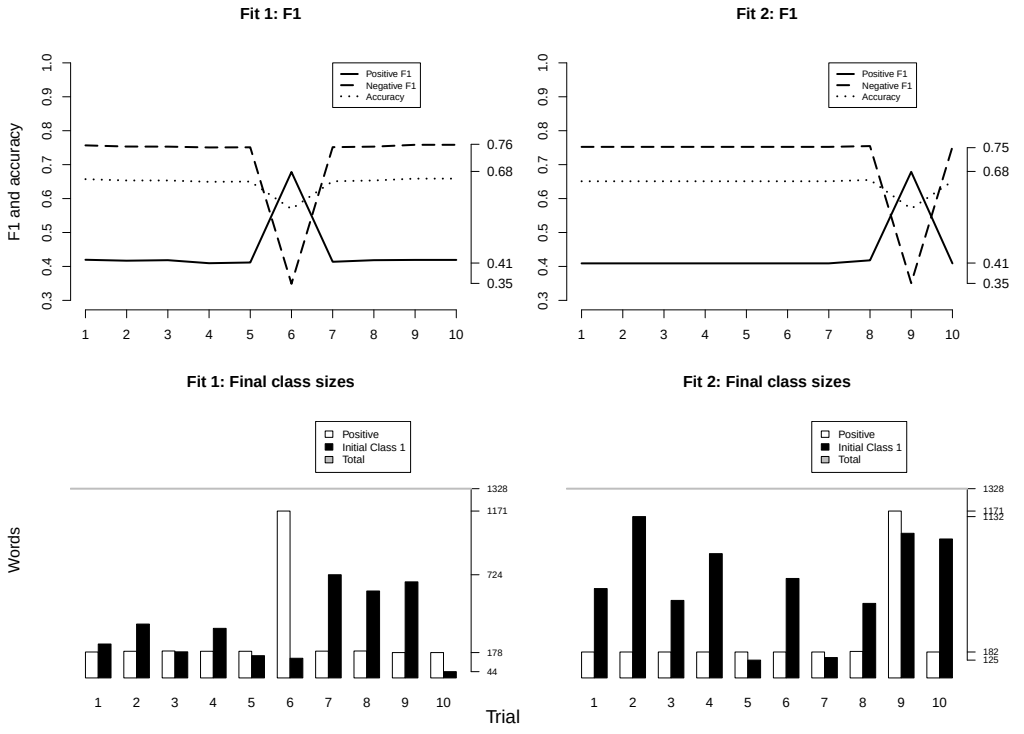


Figure 7: Assessment of the coordination experiment with two slightly different model fits. The cluster labeling procedure was effective, in that no trial had under 50% accuracy, which would have been a sign that we got the labels backwards. Drops in effectiveness seem correlated with imbalances in the initial random split, as we see with trial 6 on the left and trial 9 on the right.

6.3 Similar approaches and ideas

Clustering based on specific properties Hatzivassiloglou and McKeown (1993) apply the method of sec. 6.2 to adjective co-occurrence inside noun phrases.

Polarity by association Turney and Littman (2003) infer polarity for new words based on their distributional similarity with a set of seed-sets of known polarity.

Vector-space semantics The distributional evidence employed by Velikovich et al. (2010) (sec. 6.1) is one specific algorithm falling under the rubric of vector-space semantics, which is reviewed quite thoroughly by Turney and Pantel (2010).

7 Summary assessment

The approaches differ along too many interesting dimensions to permit fair head-to-head comparisons, but here is my overall take on them one-by-one:

Simple WordNet propagation (sec. 5.1) Interesting and informative. It suggests that sentiment and lexical relatedness are linked, which bodes well for approaches that leverage the first to get at the second. The proposals quickly summarized in sec. 5.3 show that we can deal effectively with impure paths and uneven strengths. The major foundational weaknesses: (i) different seed-sets can give very different results, and (ii) the lexicon cannot outgrown WordNet.

WordNet propagation with scores (sec. 5.2) Deepens and expands our understanding of the connection between lexical relatedness and sentiment. Its foundational weaknesses seem to me to be just those of the above.

Web-based propagation Uses the basic insight of the above, but breaks free of WordNet, thereby addressing concern (ii). Moreover, I strongly suspect that it can perform well with fewer than 20 billion words, especially since the 6- 9-grams were not contributing much. (I was getting positive initial results from an 8 million word corpus of short online review summary lines, restricting attention to just unigrams.) Weakness (i) — dependence on seed-sets — remains.

Coordination (and similar morphosyntactic ideas) Helps to highlight, in a quantitative way, the relationships between sentiment and particular words and constructions. Thus, it has substantial linguistic interest. Its major weakness is that it is limited by our cleverness in coming up with useful constructions (an analogue of the seed-set issue for the above) but perhaps Snow et al. (2005) have already shown us how to shake this sort of limitation.

A note on seed-sets Perhaps the dependence on seed-sets is a point of scientific interest — we as researchers use our knowledge of language and the domain to pick worthwhile seed-sets, as a way of extending our limited but high-precision knowledge out to vast quantities of new data. That is, perhaps (i) above is not a weakness.

References

Andreevskaia, Alina and Sabine Bergler. 2006. Mining WordNet for a fuzzy sentiment: Sentiment tag extraction from WordNet glosses. In *Proceedings of the European Chapter of the Association for Computational Linguistics (EACL)*.

Baccianella, Stefano; Andrea Esuli; and Fabrizio Sebastiani. 2010. SentiWordNet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *Proceedings of the Seventh Conference on International Language Resources and Evaluation*, 2200–2204. European Language Resources Association.

Blair-Goldensohn, Sasha; Kerry Hannan; Ryan McDonald; Tyler Neylon; George A. Reis; and Jeff Reynar. 2008. Building a sentiment summarizer for local service reviews. In *WWW Workshop on NLP in the Information Explosion Era (NLPIX)*. Beijing, China.

Brants, Thorsten and Alex Franz. 2006. Web 1T 5-gram version 1. Linguistic Data Consortium, Philadelphia.

Cerini, S.; V. Compagnoni; A. Demontis; M. Formentelli; and G. Gandini. 2007. Micro-WNOp: A gold standard for the evaluation of automatically compiled lexical resources for opinion mining. In Andrea Sansò, ed., *Language Resources and Linguistic Theory: Typology, Second Language Acquisition, English Linguistics*. Milan: Franco Angeli Editore.

Esuli, Andrea and Fabrizio Sebastiani. 2006. SentiWordNet: A publicly available lexical resource for opinion mining. In *Proceedings of the 5th Conference on Language Resources and Evaluation*, 417–422. Genova.

Fellbaum, Christiane, ed. 1998. *WordNet: An Electronic Database*. Cambridge, MA: MIT Press.

Godbole, Namrata; Manjunath Srinivasaiah; and Steven Skiena. 2007. Large-scale sentiment analysis for news and blogs. In *Proceedings of the International Conference on Weblogs and Social Media*.

Hatzivassiloglou, Vasileios and Kathleen R. McKeown. 1993. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, 172–182. ACL.

Hatzivassiloglou, Vasileios and Kathleen R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the European Chapter of the ACL*, 174–181. ACL.

Hu, Minqing and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the 10th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 168–177. ACL.

Kim, Soo-Min and Eduard H Hovy. 2006. Identifying and analyzing judgment opinions. In Robert C. Moore; Jeff A. Bilmes; Jennifer Chu-Carroll; and Mark Sanderson, eds., *Proceedings of the Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, 200–207. ACL.

Manning, Christopher D.; Prabhakar Raghavan; and Hinrich Schütze. 2009. *An Introduction to Information Retrieval*. Cambridge University Press.

Manning, Christopher D. and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. Cambridge, MA: MIT Press.

Miller, George A. 1995. WordNet: A lexical database for English. *Communications of the ACM* 38(11):39–41.

Miller, George A. 2009. WordNet — about us. WordNet. Princeton University.

Rao, Delip and Deepak Ravichandran. 2009. Semi-supervised polarity lexicon induction. In *Proceedings of the 12th Conference of the European Chapter of the ACL*, 675–682. ACL.

Snow, Rion; Daniel Jurafsky; and Andrew Y. Ng. 2005. Learning syntactic patterns for automatic hypernym discovery. In Lawrence K. Saul; Yair Weiss; and Léon Bottou, eds., *Advances in Neural Information Processing Systems 17*, 1297–1304. Cambridge, MA: MIT Press.

Späth, Helmuth. 1977. Computational experiences with the exchange method. *European Journal of Operations Research* 1(1):23–31.

Späth, Helmuth. 1980. *Cluster Analysis Algorithms for Data Reduction and Classification of Objects*. New York: Ellis Horwood.

Stone, Philip J; Dexter C Dunphry; Marshall S Smith; and Daniel M Ogilvie. 1966. *The General Inquirer: A Computer Approach to Content Analysis*. Cambridge, MA: MIT Press.

Turney, Peter D and Michael L Littman. 2003. Measuring praise and criticism: Inference of semantic orientation from association. *ACM Transactions on Information Systems* 21(4):315–346.

Turney, Peter D and Patrick Pantel. 2010. From frequency to meaning: Vector space models of semantics. *Journal of Artificial Intelligence Research* 37:141–188.

Valitutti, Alessandro; Carlo Strapparava; and Oliviero Stock. 2004. Developing affective lexical resources. *PsychNology Journal* 2(1):61–83.

Velikovich, Leonid; Sasha Blair-Goldensohn; Kerry Hannan; and Ryan McDonald. 2010. The viability of web-derived polarity lexicons. In *Proceedings of North American Association for Computational Linguistics 2010*.

Wilson, T; Janyce Wiebe; and Philip Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.