S1 Appendix: Computational methods

All of the code to perform these tests is available and document on GitHub. The repository can be found here: https://github.com/andyreagan/sentiment-analysis-comparison.

Stem matching

Of the dictionaries tested, both LIWC and MPQA use "word stems". Here we quickly note some of the technical difficulties with using word stems, and how we processed them, for future research to build upon and improve.

An example is `abandon*`, which is intended to the match words of the standard RE form `abandon[a-z]*`. A naive approach is to check each word against the regular expression, but this is prohibitively slow. We store each of the dictionaries in a "trie" data structure with a record. We use the easily available "marisa-trie" Python library, which wraps the C++ counterpart. The speed of these libraries made the comparison possible over large corpora, in particular for the dictionaries with stemmed words, where the `prefix` search is necessary. Specifically, the "trie" structure is 70 times faster than a regular expression based search for stem words. In particular, we construct two tries for each dictionary: a fixed and stemmed trie. We first attempt to match words against the fixed list, and then turn to the prefix match on the stemmed list.

Regular expression parsing

The first step in processing the text of each corpora is extracting the words from the raw text. Here we rely on a regular expression search, after first removing some punctuation. We choose to include a set of all characters that are found within the words in each of the six dictionaries tested in detail, such that it respects the parse used to create these dictionaries by retaining such characters. This takes the following form in Python, for `raw_text` as a string (note, `pdflatex` renders correctly locally, but arXiv seems to explode the link match group):

```
punctuation_to_replace = ["---","--","’’"]
for punctuation in punctuation_to_replace:
    raw_text = raw_text.replace(punctuation," ")
words = [x.lower() for x in re.findall(r"(?:[0-9][0-9,\.]*[0-9])|
                                        (?:http://[\w\./\-\?\&\#]+)|
                                        (?:[\w\@\#\’\&\]\[]+)|
                                        (?:[b}/3D;p)|’\-@x#^_0\\P(o:O{X$[=<>\]*B]+)",
                                      raw_text,flags=re.UNICODE)]
```

S2 Appendix: Continued individual comparisons

Picking up right where we left off in Section , we next compare ANEW with the other dictionaries. The ANEW-WK comparison in Panel I of Fig. 1 contains all 1030 words of ANEW, with a fit of $h_{\mathrm{ANEW}}(w) = 1.07{*}h_{\mathrm{WK}}(w) - 0.30$, making ANEW more positive and with increasing positivity for more positive words. The 20 most different scores are (ANEW,WK): fame (7.93,5.45), god (8.15,5.90), aggressive (5.10,3.08), casino (6.81,4.68), rancid (4.34,2.38), bees (3.20,5.14), teacher (5.68,7.37), priest (6.42,4.50), aroused (7.97,5.95), skijump (7.06,5.11), noisy (5.02,3.21), heroin (4.36,2.74), insolent (4.35,2.74), rain (5.08,6.58), patient (5.29,6.71), pancakes (6.08,7.43), hospital (5.04,3.52), valentine (8.11,6.40), and book (5.72,7.05). We again see some of the same words from the LabMT comparisons with these dictionaries, and again can attribute some differences to small sample sizes and differing demographics.

For the ANEW-MPQA comparison in Panel J of Fig. 1 we show the same matched word lists as before. The happiest 10 words in ANEW matched by MPQA are: clouds (6.18), bar (6.42), mind (6.68), game (6.98), sapphire (7.00), silly (7.41), flirt (7.52), rollercoaster (8.02), comedy (8.37), laughter (8.45). The least happy 5 neutral words and happiest 5 neutral words in MPQA, matched with MPQA, are: pressure (3.38), needle (3.82), quiet (5.58), key (5.68), alert (6.20), surprised (7.47), memories (7.48), knowledge (7.58), nature (7.65), engaged (8.00), baby (8.22). The least happy words in ANEW with score +1 in MPQA that are matched by MPQA are: terrified (1.72), meek (3.87), plain (4.39), obey (4.52), contents (4.89), patient (5.29), reverent (5.35), basket (5.45), repentant (5.53), trumpet (5.75). Again we see some very questionable matches by the MPQA dictionary, with broad stems capturing words with both positive and negative scores.

For the ANEW-LIWC comparison in Panel K of Fig. 1 we show the same matched word lists as before. The happiest 10 words in ANEW matched by LIWC are: lazy (4.38), neurotic (4.45), startled (4.50), obsession (4.52), skeptical (4.52), shy (4.64), anxious (4.81), tease (4.84), serious (5.08), aggressive (5.10). There are only 5 words in ANEW that are matched by LIWC with LIWC score of 0: part (5.11), item (5.26), quick (6.64), couple (7.41), millionaire (8.03). The least happy words in ANEW with score +1 in LIWC that are matched by LIWC are: heroin (4.36), virtue (6.22), save (6.45), favor (6.46), innocent (6.51), nice (6.55), trust (6.68), radiant (6.73), glamour (6.76), charm (6.77).

For the ANEW-Liu comparison in Panel L of Fig. 1 we show the same matched word lists as before, except the neutral word list because Liu has no explicit neutral words. The happiest 10 words in ANEW matched by Liu are: pig (5.07), aggressive (5.10), tank (5.16), busybody (5.17), hard (5.22), mischief (5.57), silly (7.41), flirt (7.52), rollercoaster (8.02), joke (8.10). The least happy words in ANEW with score +1 in Liu that are matched by Liu are: defeated (2.34), obsession (4.52), patient (5.29), reverent (5.35), quiet (5.58), trumpet (5.75), modest (5.76), humble (5.86), salute (5.92), idol (6.12).

For the WK-MPQA comparison in Panel P of Fig. 1 we show the same matched word lists as before. The happiest 10 words in WK matched by MPQA are: cutie (7.43), pancakes (7.43), panda (7.55), laugh (7.56), marriage (7.56), lullaby (7.57), fudge (7.62), pancake (7.71), comedy (8.05), laughter (8.05). The least happy 5

neutral words and happiest 5 neutral words in MPQA, matched with MPQA, are: sociopath (2.44), infectious (2.63), sob (2.65), soulless (2.71), infertility (3.00), thinker (7.26), knowledge (7.28), legacy (7.38), surprise (7.44), song (7.59). The least happy words in WK with score +1 in MPQA that are matched by MPQA are: kidnapper (1.77), kidnapping (2.05), kidnap (2.19), discriminating (2.33), terrified (2.51), terrifying (2.63), terrify (2.84), courtroom (2.84), backfire (3.00), indebted (3.21).

For the WK-LIWC comparison in Panel Q of Fig. 1 we show the same matched word lists as before. The happiest 10 words in WK matched by LIWC are: geek (5.56), number (5.59), fiery (5.70), trivia (5.70), screwdriver (5.76), foolproof (5.82), serious (5.88), yearn (5.95), dumpling (6.48), weeping willow (6.53). The least happy 5 neutral words and happiest 5 neutral words in LIWC, matched with LIWC, are: negative (2.52), negativity (2.74), quicksand (3.62), lack (3.68), wont (4.09), unique (7.32), millionaire (7.32), first (7.33), million (7.55), rest (7.86). The least happy words in WK with score +1 in LIWC that are matched by LIWC are: heroin (2.74), friendless (3.15), promiscuous (3.32), supremacy (3.48), faithless (3.57), laughingstock (3.77), promiscuity (3.95), tenderfoot (4.26), succession (4.52), dynamite (4.79).

For the WK-Liu comparison in Panel R of Fig. 1 we show the same matched word lists as before, except the neutral word list because Liu has no explicit neutral words. The happiest 10 words in WK matched by Liu are: goofy (6.71), silly (6.72), flirt (6.73), rollercoaster (6.75), tenderness (6.89), shimmer (6.95), comical (6.95), fanciful (7.05), funny (7.59), fudge (7.62), joke (7.88). The least happy words in WK with score +1 in Liu that are matched by Liu are: defeated (2.59), envy (3.05), indebted (3.21), supremacy (3.48), defeat (3.74), overtake (3.95), trump (4.18), obsession (4.38), dominate (4.40), tough (4.45).

Now we'll focus our attention on the MPQA row, and first we see comparisons against the three full range dictionaries. For the first match against LabMT in Panel D of Fig. 1, the MPQA match catches 431 words with MPQA score 0, while LabMT (without stems) matches 268 words in MPQA in Panel S (1039/809 and 886/766 for the positive and negative words of MPQA). Since we've already highlighted most of these words, we move on and focus our attention on comparing the ± 1 dictionaries.

In Panels V–X, BB–DD, and HH–JJ of Fig. 1 there are a total of 6 bins off of the diagonal, and we focus out attention on the bins that represent words that have opposite scores in each of the dictionaries. For example, consider the matches made my MPQA in Panel BB: the words in the top left corner and bottom right corner with are scored in a opposite manner in LIWC, and are of particular concern. Looking at the words from Panel W with a +1 in MPQA and a -1 in LIWC (matched by LIWC) we see: stunned, fiery, terrified, terrifying, yearn, defense, doubtless, foolproof, risk-free, exhaustively, exhaustive, blameless, low-risk, low-cost, lower-priced, guiltless, vulnerable, yearningly, and yearning. The words with a -1 in MPQA that are +1 in LIWC (matched by LIWC) are: silly, madly, flirt, laugh, keen, superiority, supremacy, sillily, dearth, comedy, challenge, challenging, cheerless, faithless, laughable, laughably, laughingstock, laughter, laugh, grating, opportunistic, joker, challenge, flirty.

In Panel W of 1, the words with a +1 in MPQA and a -1 in Liu (matched by Liu) are: solicitude, flair, funny, resurgent, untouched, tenderness, giddy, vulnerable, and

joke. The words with a -1 in MPQA that are +1 in Liu, matched by Liu, are: superiority, supremacy, sharp, defeat, dumbfounded, affectation, charisma, formidable, envy, empathy, trivially, obsessions, and obsession.

In Panel BB of 1, the words with a +1 in LIWC and a -1 in MQPA (matched by MPQA) are: silly, madly, flirt, laugh, keen, determined, determina, funn, fearless, painl, cute, cutie, and gratef. The words with a -1 in LIWC and a +1 in MQPA, that are matched by MPQA, are: stunned, terrified, terrifying, fiery, yearn, terrify, aversi, pressur, careless, helpless, and hopeless.

In Panel DD of 1, the words with a -1 in LIWC and a +1 in Liu, that are matched by Liu, are: silly, and madly. The words with a +1 in LIWC and a -1 in Liu, that are matched by Liu, are: stunned, and fiery.

In Panel HH of 1, the words with a -1 in Liu and a +1 in MPQA, that are matched by MPQA, are: superiority, supremacy, sharp, defeat, dumbfounded, charisma, affectation, formidable, envy, empathy, trivially, obsessions, obsession, stabilize, defeated, defeating, defeats, dominated, dominates, dominate, dumbfounding, cajole, cuteness, faultless, flashy, fine-looking, finer, finest, panoramic, pain-free, retractable, believeable, blockbuster, empathize, err-free, mind-blowing, marvelled, marveled, trouble-free, thumb-up, thumbs-up, long-lasting, and viewable. The words with a +1 in Liu and a -1 in MPQA, that are matched by MPQA, are: solicitude, flair, funny, resurgent, untouched, tenderness, giddy, vulnerable, joke, shimmer, spurn, craven, aweful, backwoods, backwood, back-woods, back-wood, back-logged, backaches, backache, backaching, backbite, tingled, glower, and gainsay.

In Panel II of 1, the words with a +1 in Liu and a -1 in LIWC, that are matched by LIWC, are: stunned, fiery, defeated, defeating, defeats, defeat, doubtless, dominated, dominates, dominate, dumbfounded, dumbfounding, faultless, foolproof, problem-free, problem-solver, risk-free, blameless, envy, trivially, trouble-free, tougher, toughest, tough, low-priced, low-price, low-risk, low-cost, lower-priced, geekier, geeky, guiltless, obsessions, and obsession. The words with a -1 in Liu and a +1 in LIWC, that are matched by LIWC, are: silly, madly, sillily, dearth, challenging, cheerless, faithless, flirty, flirt, funnily, funny, tenderness, laughable, laughably, laughingstock, grating, opportunistic, joker, and joke.

In the off-diagonal bins for all of the ±1 dictionaries, we see many of the same words. Again MPQA stem matches are disparagingly broad. We also find matches by LIWC that are concerning, and should in all likelihood be removed from the dictionary.

S3 Appendix: Coverage for all corpuses

We provide coverage plots for the other three corpuses.

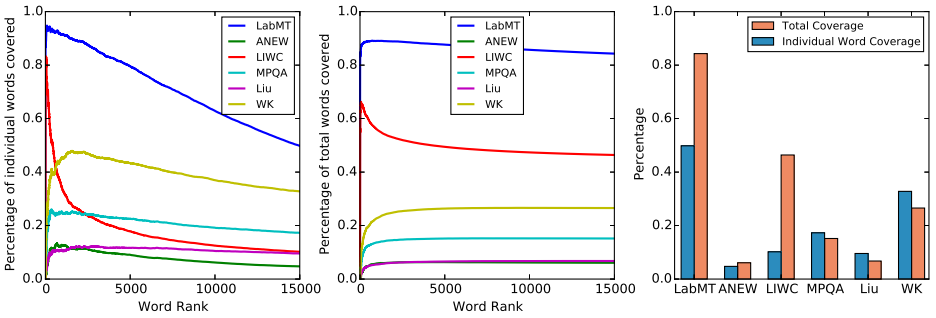

Figure S1 Coverage of the words on twitter by each of the dictionaries.

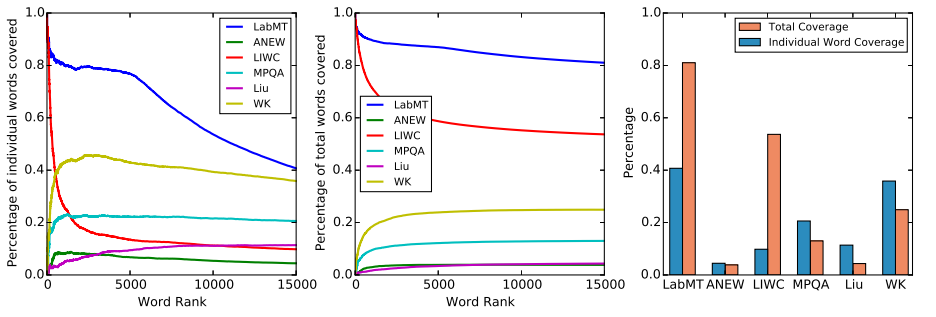

Figure S2 Coverage of the words in Google books by each of the dictionaries.

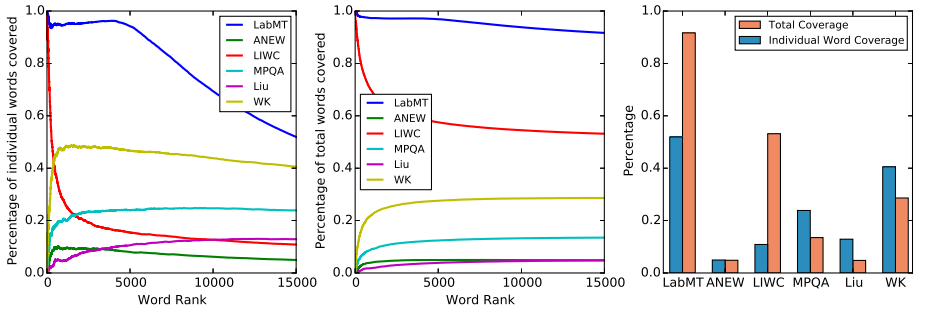

Figure S3 Coverage of the words in the New York Times by each of the dictionaries.

S4 Appendix: Sorted New York Times rankings

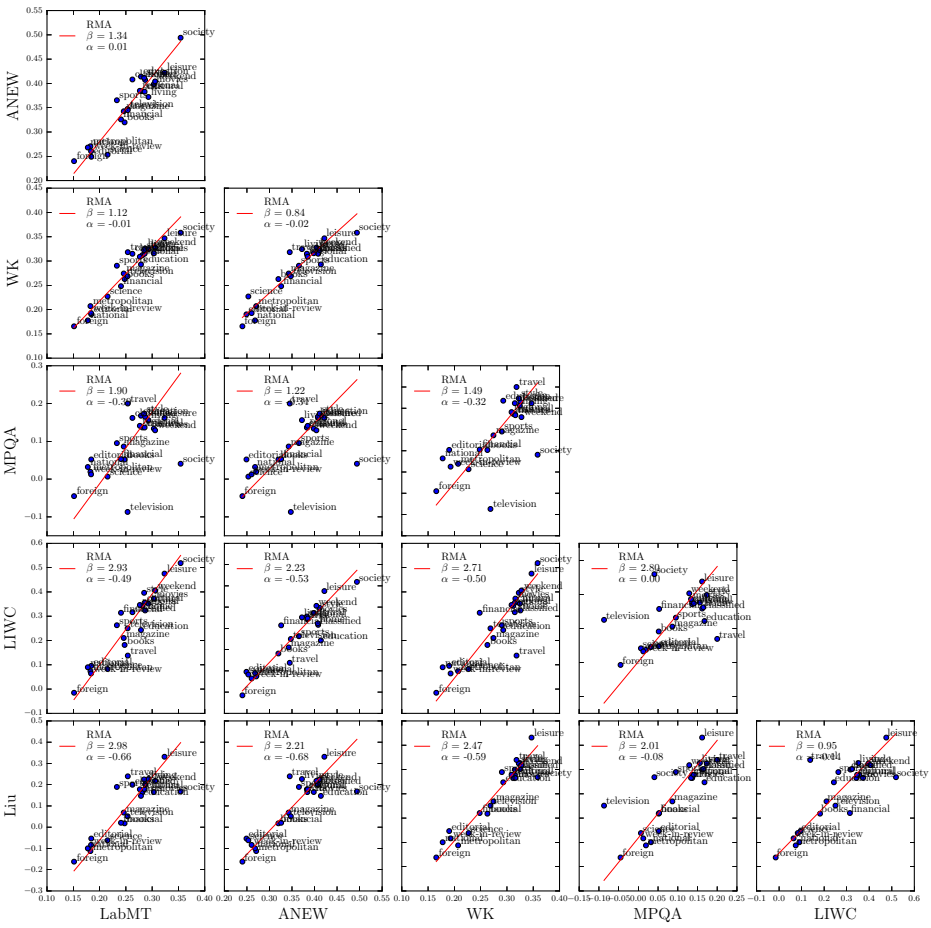

Figure S4 NYT Sections scatterplot. The RMA fit α and β for the formula $y = \alpha + \beta x$. For the sake of comparison, we normalized each dictionary to the range [-.5,.5] by subtracting the mean score (5 or 0) and dividing by the range (8 or 2).

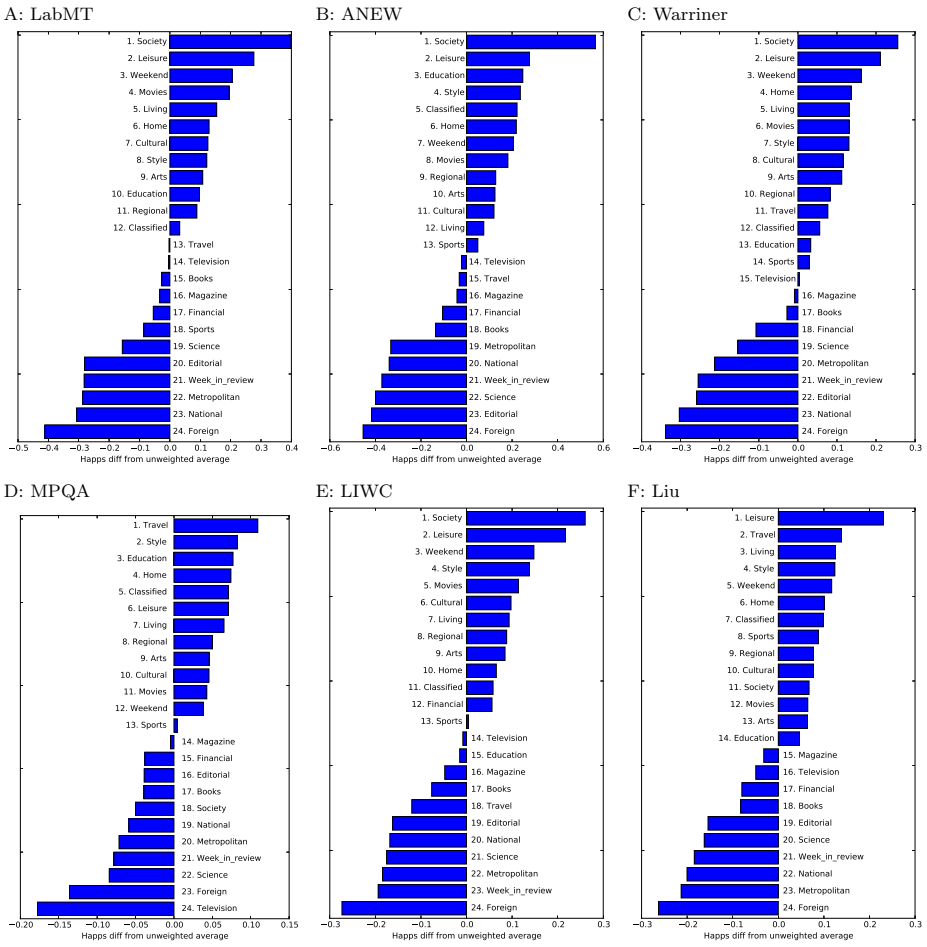

Figure S5 Sorted bar charts ranking each of the 24 New York Times Sections for each dictionary tested.

S5 Appendix: Movie Review Distributions

Here we examine the distributions of movie review scores. These distributions are each summarized by their mean and standard deviation in panels of Figure 2 for each dictionary. For example, the left most error bar of each panel in Figure 2 shows the standard deviation around the mean for the distribution of individual review scores (Figure S6).

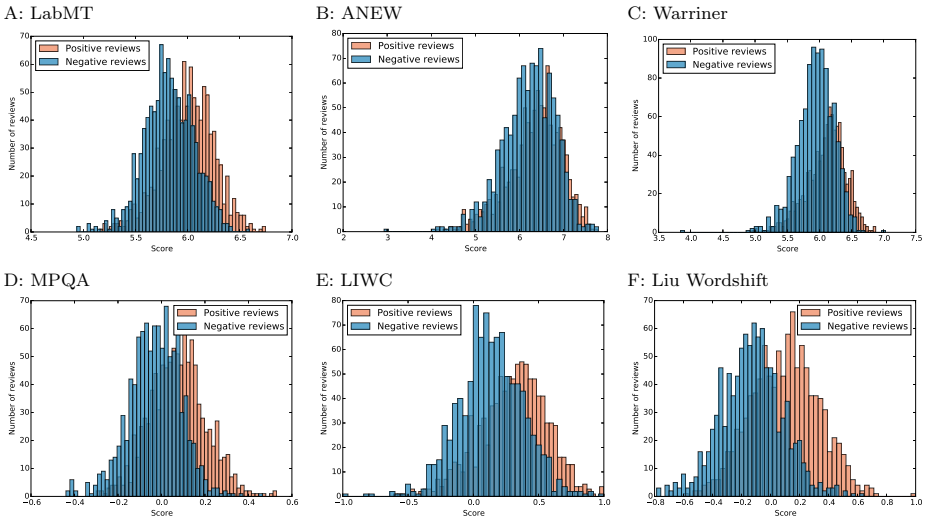

Figure S6 Binned scores for each review by each corpus with a stop value of $\Delta_h = 1.0$.

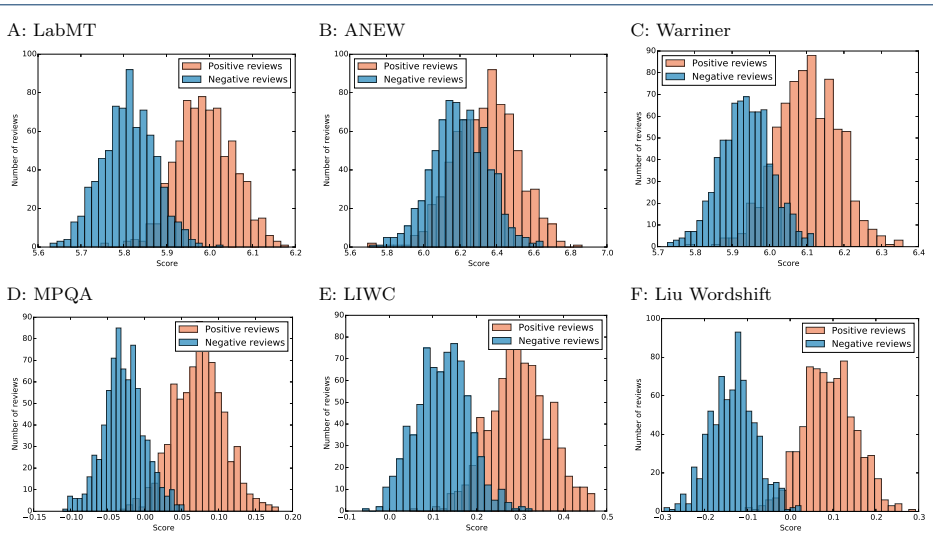

Figure S7 Binned scores for samples of 15 concatenated random reviews. Each dictionary uses stop value of $\Delta_h = 1.0$.

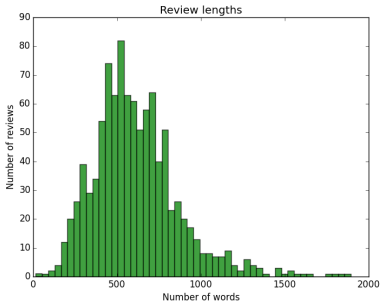

Figure S8 Binned length of positive reviews, in words.

S6 Appendix: Google Books correlations and word shifts

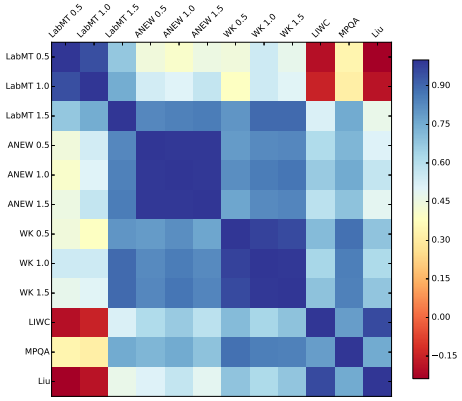

Figure S9 Google Books correlations. Here we include correlations for the google books time series, and word shifts for selected decades (1920's,1940's,1990's,2000's).

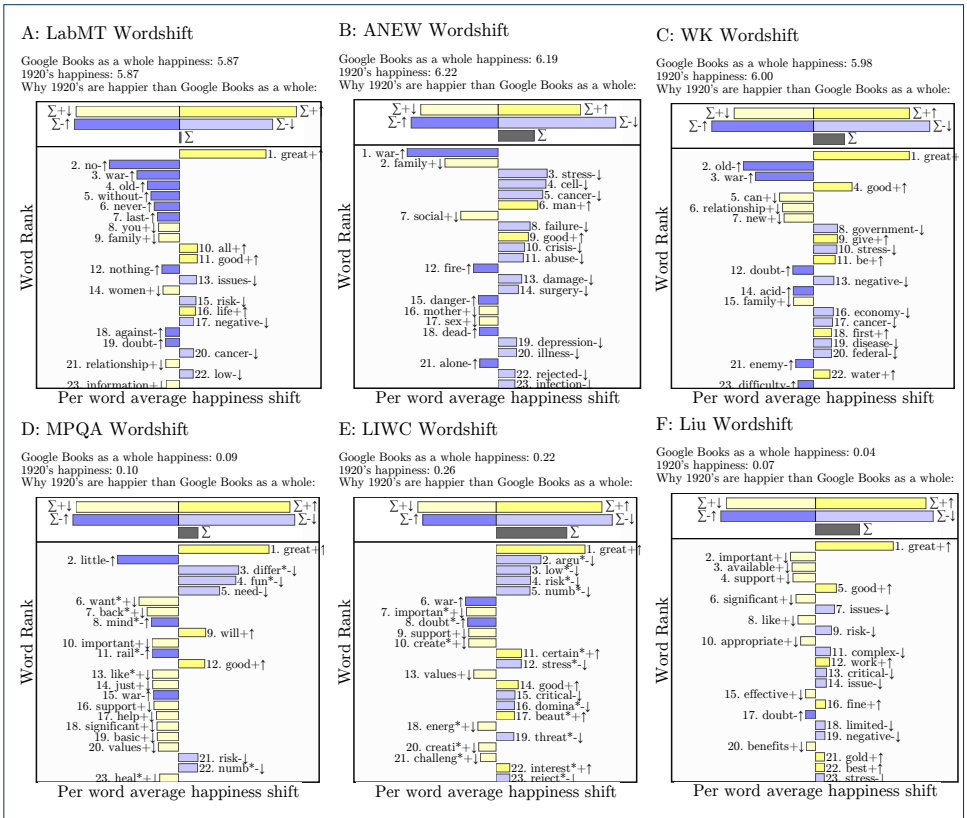

Figure S10 Google Books shifts in the 1920's against the baseline of Google Books.

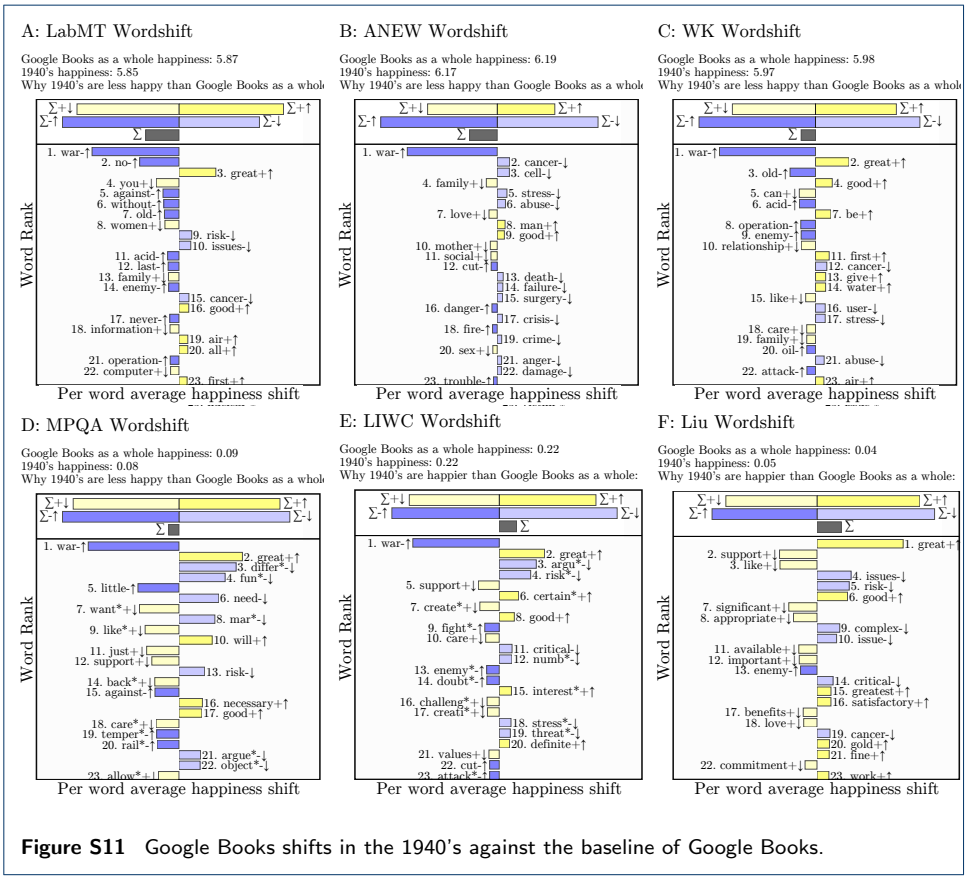

Figure S11 Google Books shifts in the 1940's against the baseline of Google Books.

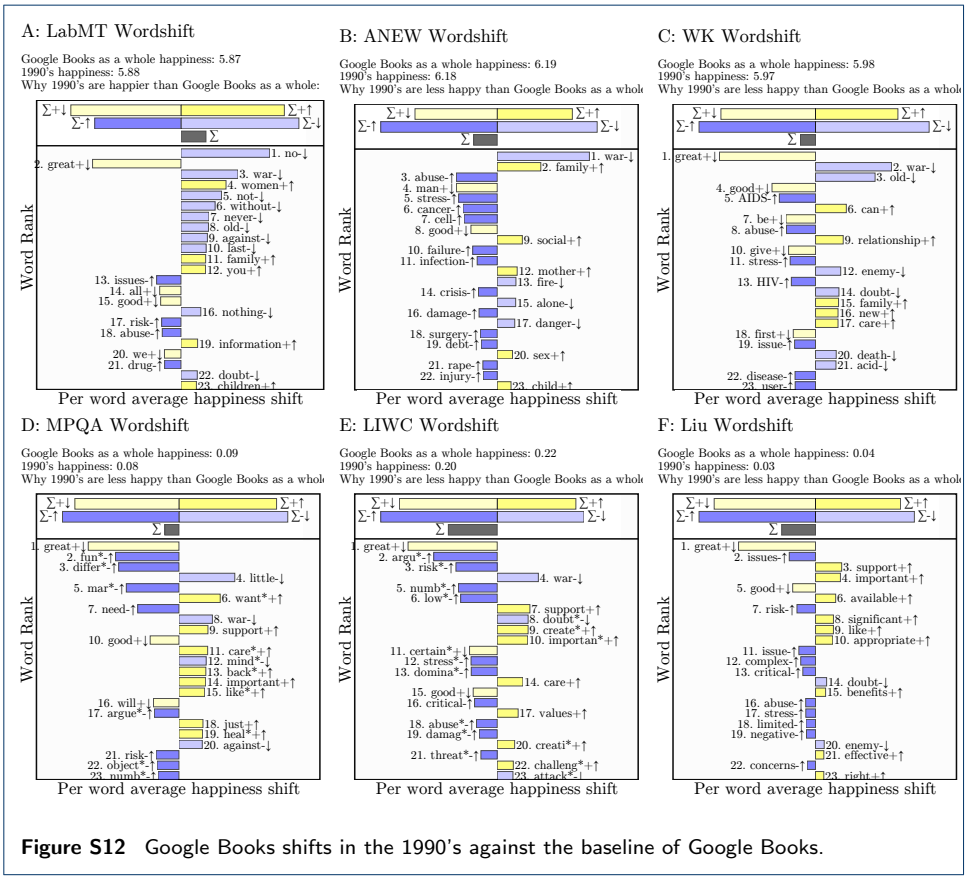

Figure S12 Google Books shifts in the 1990's against the baseline of Google Books.

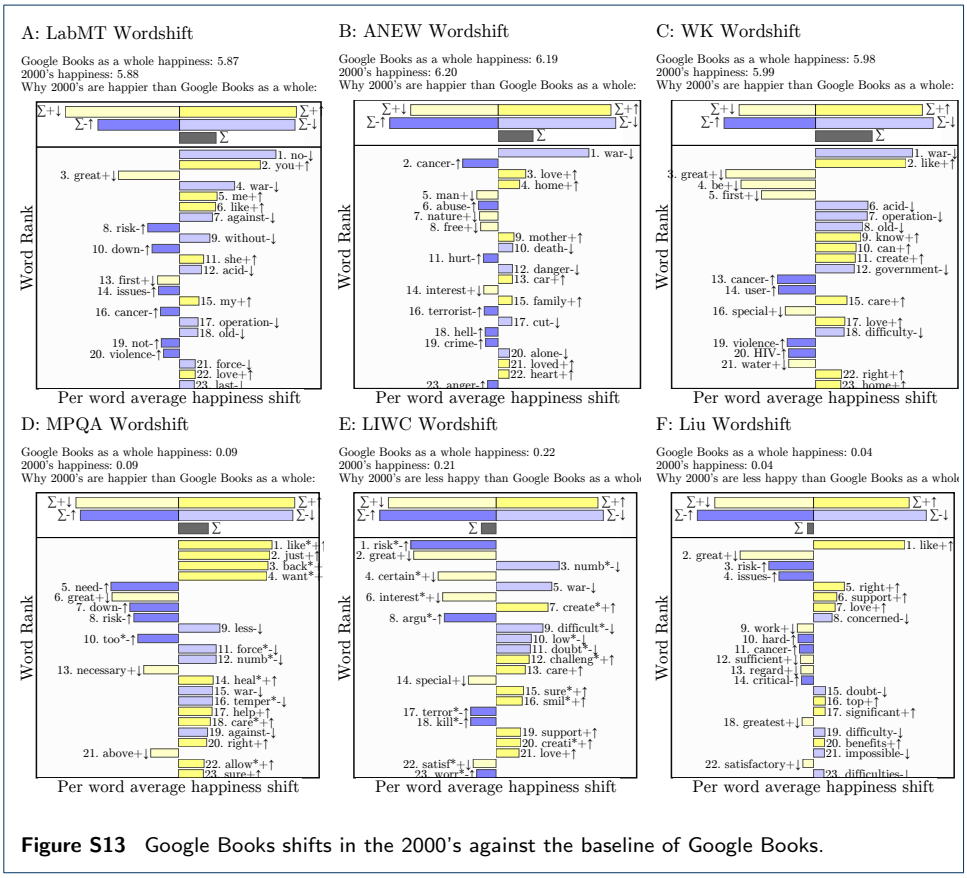

Figure S13 Google Books shifts in the 2000's against the baseline of Google Books.

S7 Appendix: Additional Twitter time series, correlations, and shifts

First, we present additional Twitter time series:

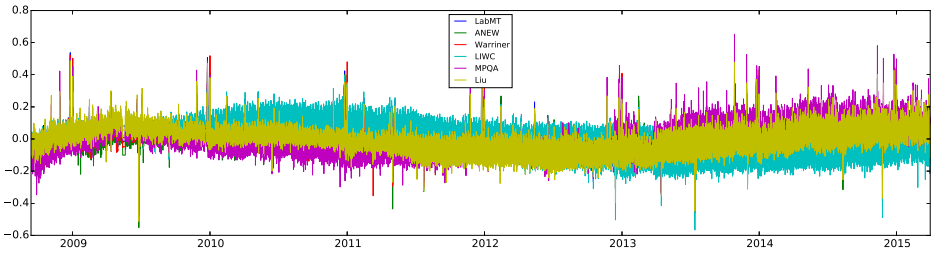

Figure S14 Normalized time series on Twitter using Δ_h of 1.0 for all. For resolution of 3 hours. We do not include any of the time series with resolution below 3 hours here because there are too many data points to see.

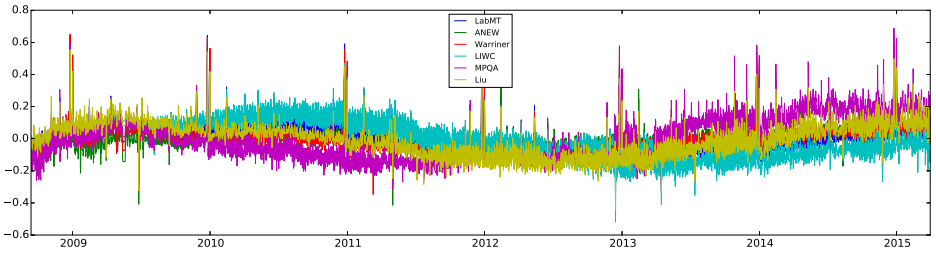

Figure S15 Normalized time series on Twitter using Δ_h of 1.0 for all. For resolution of 12 hours.

Next, we take a look at more correlations:

Now we include word shift graphs that are absent from the manuscript itself.

Finally, we include the results of each dictionary applied to a set of annotated Twitter data. We apply sentiment dictionaries to rate individual Tweets and classify a Tweet as positive (negative) if the Tweet rating is greater (less) than the average of all scores in dictionary.

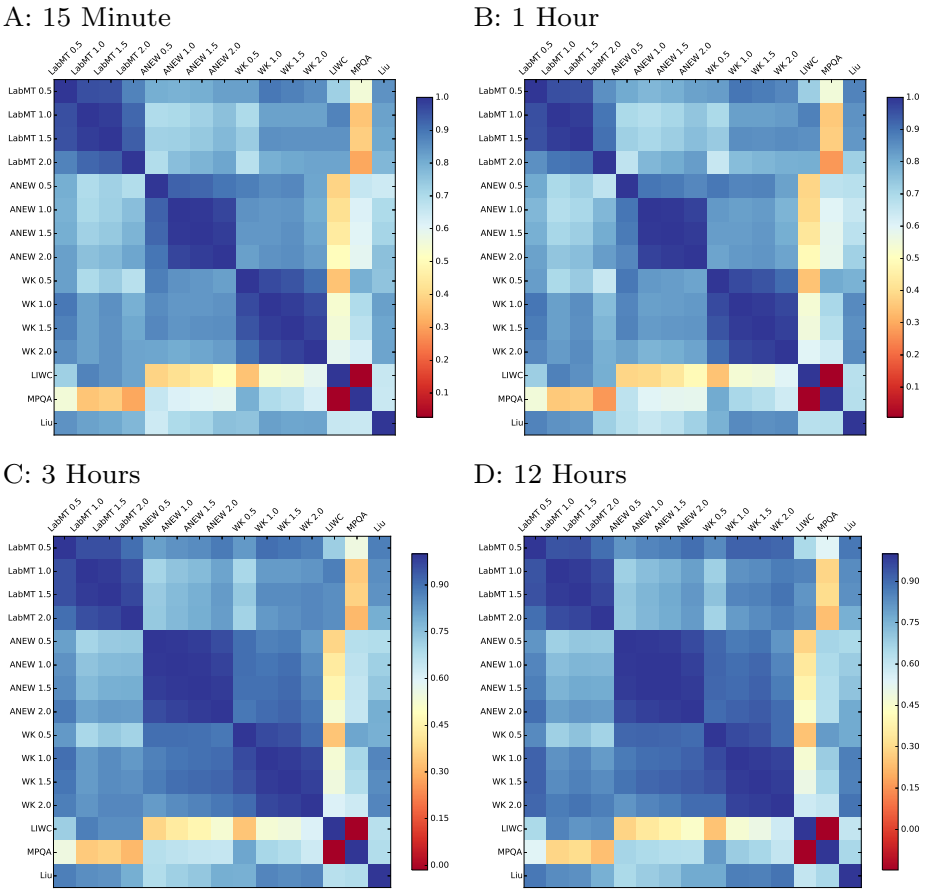

Figure S16 Pearson's r correlation between Twitter time series for all resolutions below 1 day.

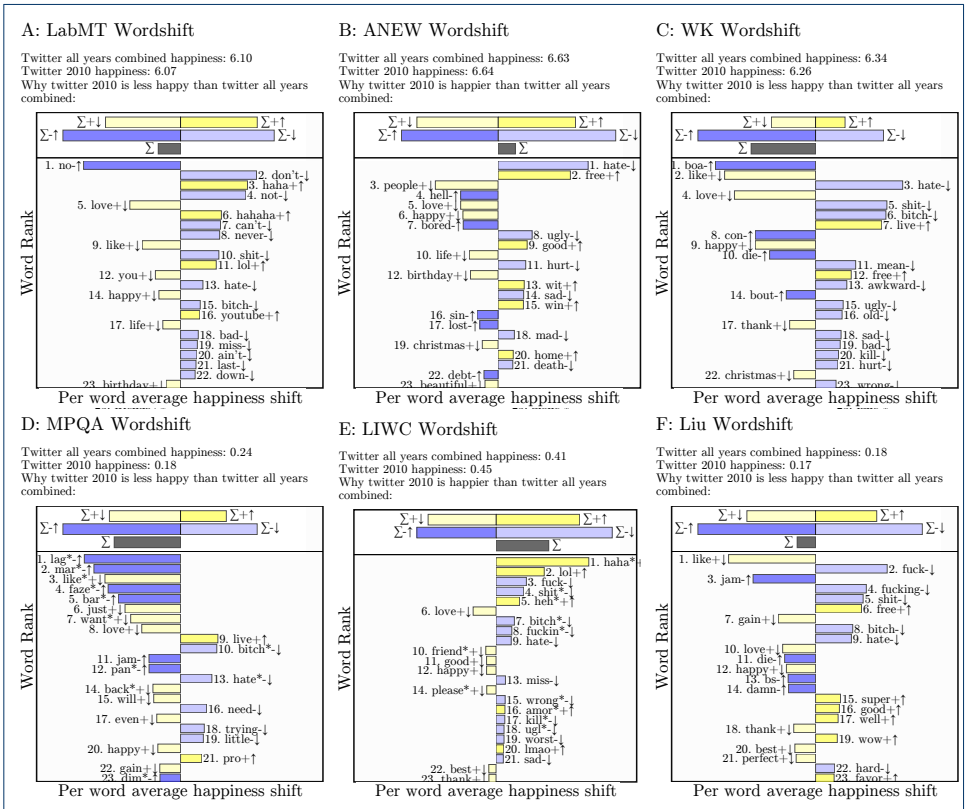

Figure S17 Word Shifts for Twitter in 2010. The reference word usage is all of Twitter (the 10% Gardenhose feed) from September 2008 through April 2015, with the word usage normalized by year.

Rank	Dictionary	% Tweets scored	F1 of Tweets scored	Calibrated F1	Overall F1
1.	Sent140Lex	100.0	0.89	0.88	0.89
2.	labMT	100.0	0.69	0.78	0.69
3.	HashtagSent	100.0	0.67	0.64	0.67
4.	SentiWordNet	98.6	0.67	0.68	0.67
5.	VADER	81.3	0.75	0.81	0.61
6.	SentiStrength	73.9	0.83	0.81	0.61
7.	SenticNet	97.3	0.61	0.64	0.59
8.	Umigon	67.1	0.87	0.85	0.58
9.	SOCAL	82.2	0.71	0.75	0.58
10.	WDAL	99.9	0.58	0.64	0.58
11.	AFINN	73.6	0.78	0.80	0.57
12.	OL	66.7	0.83	0.82	0.55
13.	MaxDiff	94.1	0.58	0.70	0.54
14.	EmoSenticNet	96.0	0.56	0.59	0.54
15.	MPQA	73.2	0.73	0.72	0.53
16.	WK	96.5	0.53	0.72	0.51
17.	LIWC15	61.8	0.81	0.78	0.50
18.	Pattern	69.0	0.71	0.75	0.49
19.	GI	67.6	0.72	0.70	0.49
20.	LIWC07	60.3	0.80	0.75	0.48
21.	LIWC01	54.3	0.83	0.75	0.45
22.	EmoLex	59.4	0.73	0.69	0.43
23.	ANEW	64.1	0.65	0.68	0.42
24.	USent	4.5	0.74	0.73	0.03
25.	PANAS-X	1.7	0.88	–	0.01
26.	Emoticons	1.4	0.72	0.77	0.01

Table S1 Ranked results of sentiment dictionary performance on individual Tweets from STS-Gold dataset (Saif, 2013). We report the percentage of Tweets for which each dictionary contains at least 1 entry, the F1 score on those Tweets, and the overall classification F1 score. The calibrated F1 score tunes the decision threshold between positive and negative Tweets with a random 10% training sample.

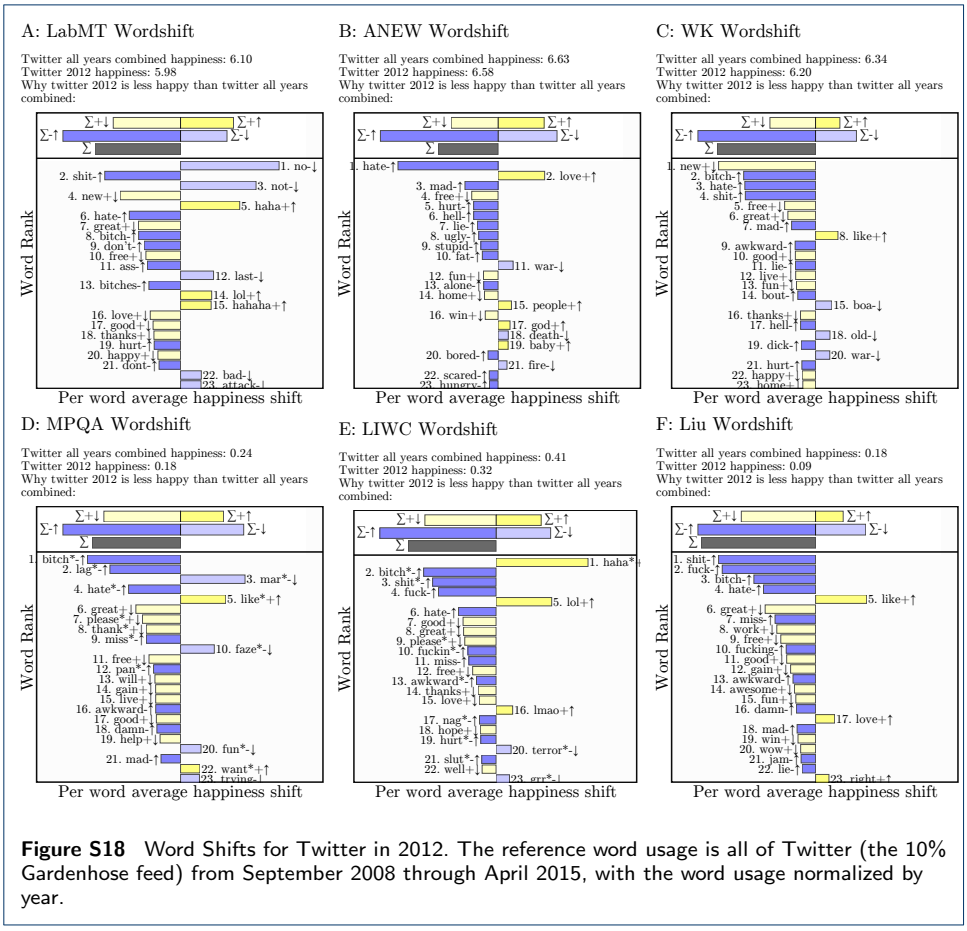

Figure S18 Word Shifts for Twitter in 2012. The reference word usage is all of Twitter (the 10% Gardenhose feed) from September 2008 through April 2015, with the word usage normalized by year.

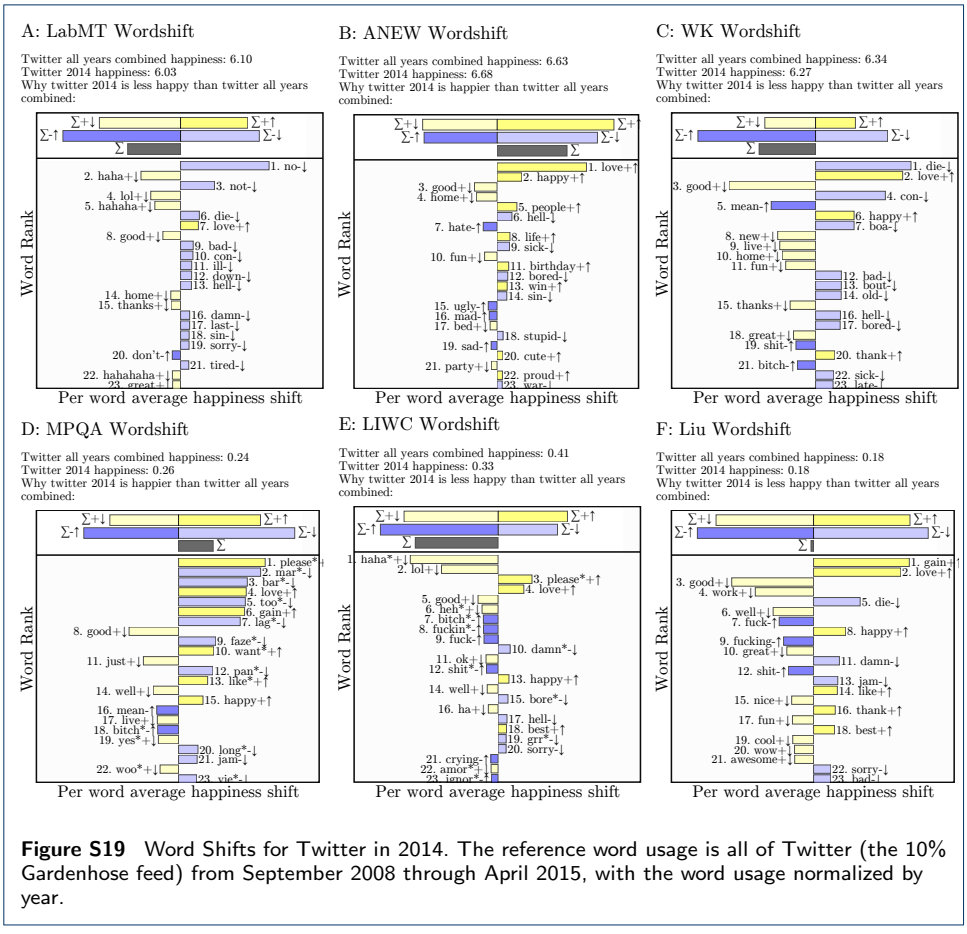

Figure S19 Word Shifts for Twitter in 2014. The reference word usage is all of Twitter (the 10% Gardenhose feed) from September 2008 through April 2015, with the word usage normalized by year.

S8 Appendix: Naive Bayes results and derivation

We now provide more details on the implementation of Naive Bayes, a derivation of the linearity structure, and more results from the classification of Movie Reviews.

First, to implement a binary Naive Bayes classifier for a collection of documents, we denote each of the N words in the given document T as w_i, thus the normalized word frequency is $f_i(T) = w_i/N$, and finally we denote the class labels c_1, c_2. The probability of a document T belonging to class c_1 can be written as

$$P(c_1|T) = \frac{P(c_1)P(T|c_1)}{P(T)}.$$

Since we do not know $P(T|c_1)$ explicitly, we make the *naive* assumption that each word appears independently, and thus write

$$P(c_1|T) = \frac{P(c_1) \cdot [P(f_1(T)|c_1) \cdot P(f_2(T)|c_1) \cdots P(f_N(T)|c_1)]}{P(T)}.$$

Since we are only interested in comparing $P(c_1|T)$ and $P(c_2|T)$, we disregard the shared denominator and have

$$P(c_1|T) \propto P(c_1) \cdot [P(f_1(T)|c_1) \cdot P(f_2(T)|c_1) \cdots P(f_N(T)|c_1)]\,.$$

Finally we say that document T belongs to class c_1 if $P(c_1|T) > P(c_2|T)$. Given that the probabilities of individual words are small, to avoid machine truncation error we compute these probabilities in log space, such that the product of individual word likelihoods becomes a sum

$$\log P(c_1|T) \propto \log P(c_1) + \sum_{i=1}^{N} \log P(f_i(T)|c_1).$$

Assigning a classification of class c_1 if $P(c_1|T) > P(c_2|T)$ is the same as saying that the difference between the two is positive, i.e. $P(c_1|T) - P(c_2|T) > 0$ and since the logarithm is monotonic, $\log P(c_1|T) - \log P(c_2|T) > 0$. To examine how individual words contribute to this difference, we can write

$$\begin{aligned}
0 &< \log P(c_1|T) - \log P(c_2|T) \\
&\propto \log P(c_1) + \sum_{i=1}^{N} \log P(f_i(T)|c_1) - \log P(c_2) - \sum_{i=1}^{N} \log P(f_i(T)|c_2) \\
&\propto \log P(c_1) - \log P(c_2) + \sum_{i=1}^{N} [\log P(f_i(T)|c_1) - \log P(f_i(T)|c_2)] \\
&\propto \log \frac{P(c_1)}{P(c_2)} + \sum_{i=1}^{N} \log \frac{P(f_i(T)|c_1)}{P(f_i(T)|c_2)}.
\end{aligned}$$

We can see from the above that the contribution of each word w_i (or more accurately, the likelihood of the frequency in document T being predictive of class c as $P(f_i(T)|c_1)$) is a linear constituent of the classification.

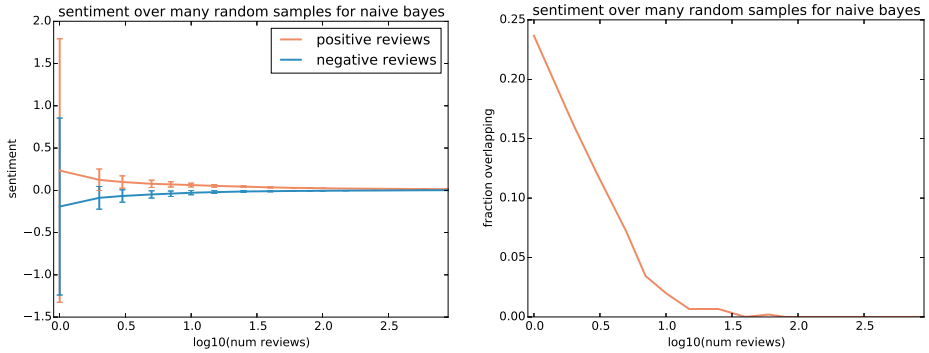

Figure S20 Results of the NB classifier on the Movie Reviews corpus.

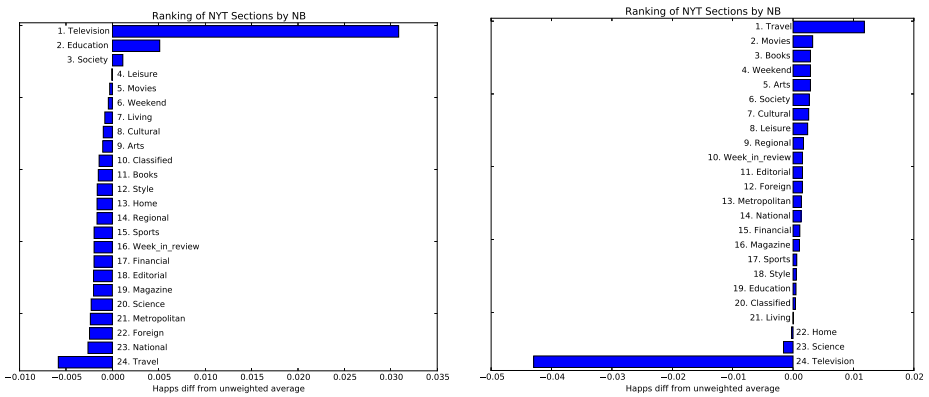

Figure S21 NYT Sections ranked by Naive Bayes in two of the five trials.

Next, we include the detailed results of the Naive Bayes classifier on the Movie Review corpus.

Most informative			
Positive		Negative	
Word	Value	Word	Value
27.27	flynt	20.21	godzilla
26.33	truman	15.95	werewolf
20.68	charles	13.83	gorilla
15.04	event	13.83	spice
14.10	shrek	13.83	memphis
13.16	cusack	13.83	sgt
13.16	bulworth	12.76	jennifer
13.16	robocop	12.76	hill
12.22	jedi	11.70	max
12.22	gangster	11.70	200

NYT Society			
Positive		Negative	
Word	Value	Word	Value
26.08	truman	20.40	godzilla
20.49	charles	12.88	hill
12.11	gangster	12.88	jennifer
10.25	speech	10.73	fatal
9.32	melvin	8.59	freddie
8.85	wars	8.59	=
7.45	agents	8.59	mess
6.52	dance	8.59	gene
6.52	bleak	8.59	apparent
6.52	pitt	7.51	travolta

Table S2 Trial 1 of Naive Bayes trained on a random 10% of the movie review corpus, and applied to the New York Times Society section. We show the words which are used by the trained classifier to classify individual reviews (in corpus), and on the New York Times (out of corpus). In addition, we report a second trial in Table S3, since Naive Bayes is trained on a random subset of data, to show the variation in individual words between trials (while performance is consistent).

Most informative			
Positive		Negative	
Word	Value	Word	Value
18.11	shrek	34.63	west
17.15	poker	24.14	webb
15.25	shark	18.89	jackal
14.29	maggie	17.84	travolta
13.34	guido	17.84	woo
13.34	outstanding	17.84	coach
13.34	political	16.79	awful
13.34	journey	16.79	brenner
13.34	bulworth	15.74	gabriel
12.39	bacon	15.74	general's

NYT Society			
Positive		Negative	
Word	Value	Word	Value
17.79	poker	33.39	west
13.84	journey	17.20	coach
13.84	political	17.20	travolta
8.90	tribe	15.18	gabriel
7.91	tony	12.14	pointless
7.91	price	9.44	stupid
7.91	threat	8.09	screaming
7.12	titanic	7.59	mess
6.92	dicaprio	7.42	boring
6.92	kate	7.08	=

Table S3 Trial 2 of Naive Bayes trained on a random 10% of the movie review corpus, and applied to the New York Times Society section. We show the words which are used by the trained classifier to classify individual reviews (in corpus), and on the New York Times (out of corpus). This second trial is in addition to the first trial in Table S2, since Naive Bayes is trained on a random subset of data, to show the variation in individual words between trials (while performance is consistent).

S9 Appendix: Movie review benchmark of additional dictionaries

Here, we present the accuracy of each dictionary applied to binary classification of Movie Reviews.

Rank	Title	% Scored	F1 Trained	F1 Untrained
1.	OL	100	0.70	0.71
2.	HashtagSent	100	0.67	0.66
3.	MPQA	100	0.67	0.66
4.	SentiWordNet	100	0.65	0.65
5.	labMT	100	0.64	0.63
6.	AFINN	100	0.67	0.63
7.	Umigon	100	0.65	0.62
8.	GI	100	0.65	0.61
9.	SOCAL	100	0.71	0.60
10.	VADER	100	0.67	0.60
11.	WDAL	100	0.60	0.59
12.	SentiStrength	100	0.63	0.58
13.	EmoLex	100	0.65	0.56
14.	LIWC15	100	0.64	0.55
15.	LIWC01	100	0.65	0.54
16.	LIWC07	100	0.64	0.53
17.	Pattern	100	0.73	0.52
18.	PANAS-X	33	0.51	0.51
19.	Sent140Lex	100	0.68	0.47
20.	SenticNet	100	0.62	0.45
21.	ANEW	100	0.57	0.36
22.	MaxDiff	100	0.66	0.36
23.	EmoSenticNet	100	0.58	0.34
24.	WK	100	0.63	0.34
25.	Emoticons	0	–	–
26.	USent	40	–	–

Table S4 Ranked performance of dictionaries on the Movie Review corpus.

Rank	Title	% Scored	F1 Trained of Scored	F1 Untrained of Scored	F1 Untrained, All
1.	HashtagSent	100	0.55	0.55	0.55
2.	LIWC15	99	0.53	0.55	0.55
3.	LIWC07	99	0.53	0.55	0.54
4.	LIWC01	99	0.52	0.55	0.54
5.	labMT	99	0.54	0.54	0.54
6.	Sent140Lex	100	0.55	0.54	0.54
7.	SentiWordNet	99	0.54	0.53	0.53
8.	WDAL	99	0.53	0.53	0.52
9.	EmoLex	95	0.54	0.55	0.52
10.	MPQA	93	0.54	0.55	0.52
11.	SenticNet	97	0.53	0.52	0.50
12.	SOCAL	88	0.56	0.55	0.49
13.	EmoSenticNet	98	0.52	0.46	0.45
14.	Pattern	81	0.55	0.55	0.45
15.	GI	80	0.55	0.55	0.44
16.	WK	97	0.54	0.45	0.44
17.	OL	76	0.56	0.57	0.44
18.	VADER	79	0.56	0.55	0.43
19.	SentiStrength	77	0.54	0.54	0.41
20.	MaxDiff	83	0.54	0.49	0.41
21.	AFINN	70	0.56	0.56	0.39
22.	ANEW	63	0.52	0.48	0.30
23.	Umigon	53	0.56	0.56	0.30
24.	PANAS-X	1	0.53	0.53	0.01
25.	Emoticons	0	–	–	–
26.	USent	2	–	–	–

Table S5 Ranked performance of dictionaries on the Movie Review corpus, broken into sentences.

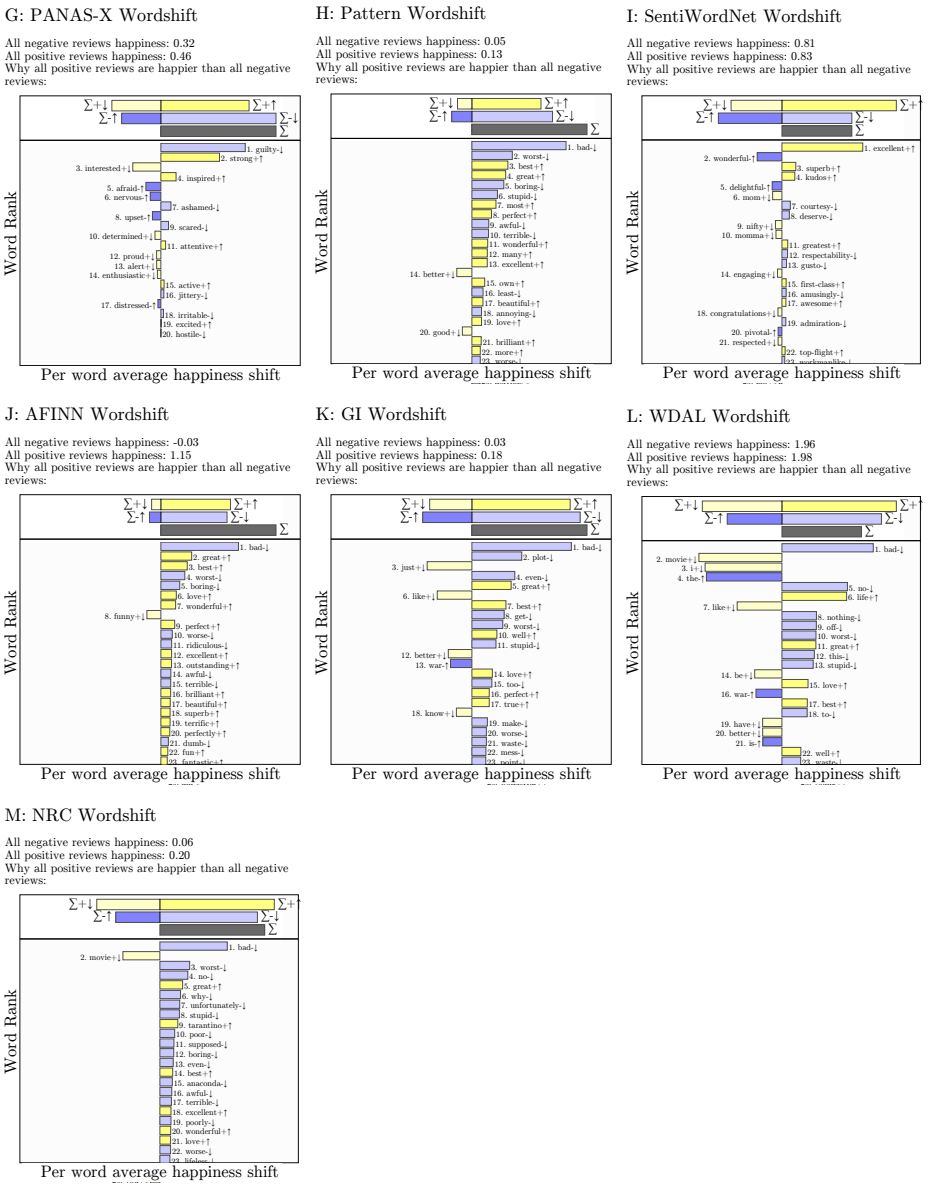

Figure S22 Word shifts for the movie review corpus, with panel letters continuing from Fig. 5. We again see many of the same patterns, and refer the reader to Fig. 5 for a more in depth analysis.

S10 Appendix: Coverage removal and binarization tests of labMT dictionary

Here, we perform a detailed analysis of the labMT dictionary to further isolate the effects of dictionary coverage and scoring type. This analysis is motivated by ensuring that the our results are not confounded entirely by the quality of the word scores across dictionaries, such that the effect of coverage and scoring type are isolated. We focus on the Movie Review corpus for this analysis and analyzing the different between positive and negative reviews using word shift graphs. While our attention is focused on a qualitative understanding of the differences in these two sets of documents, we also report the accuracy of the labMT dictionary with the aforementioned modifications using the F1 score.

Binarization

First, we gradually reduce the range of scores in the labMT dictionary from a centered -4 $\rightarrow$ 4, down to just the integer scores -1 and 1. This process is accomplished by first using a $\Delta_h = 1.00$, leaving words with scores from 1–4 and 6–9, and then applying a linear transformation to these sets of words. We subtract the center value of 5.0 from the words, leaving words with ranges from -4– -1 and 1–4, and then linearly map these sets to scores with a reduced range. For a binarization of 25%, we map -4– -1 to -3.25 – -1 and 1–4 to 1–3.25, reducing the range in direction from 3 to 2.25 (a 25% reduction). For a binarization of 50%, this becomes a map of -4– -1 to -2.5 – -1 and 1–4 to 1–2.5, leaving only half of the original range of values. Finally, a binarization of 100% sets the score for all words -4– -1 to -1, and words 1–4 to 1.

In Figs. S23–S26 we observe that the binarization of the labMT dictionary results in observably different word shift graphs by changing which words contribute to the sentiment differences as well as reducing the difference in sentiment scores between the two corpora. Looking specifically at Fig. S26, the top 5 words in the control word shift graph are bad, no, movie, worst, and war. In the binarized version, the top 5 are bad, no, movie, nothing, and worst. The top 5 from the continuous dictionary move into places 1, 2, 3, 5, and 10. Examining only the positive words that increased in frequency (not all shown in the Figure), we have "3. movie (3)", "11. like (24)", "32. funny (102)", "33. better (46)", and "43. jokes (133)" in the control version, with these words' positions in the binarized version in parenthesis. In the binarized version, these top words are "3. movie (3)", "24. like (11)", "30. you (84)", "36. up (126)", "37. all (98)", where the first number is the place in the overall list for the given labMT score list, with the place for that word in the control word shift graph in parenthesis.

In Figure S27, the F1 score is show across this gradual, linear change to a binary dictionary. We observe that the full binarization of the labMT dictionary results in a degradation of performance, although the differences are not statistically significant.

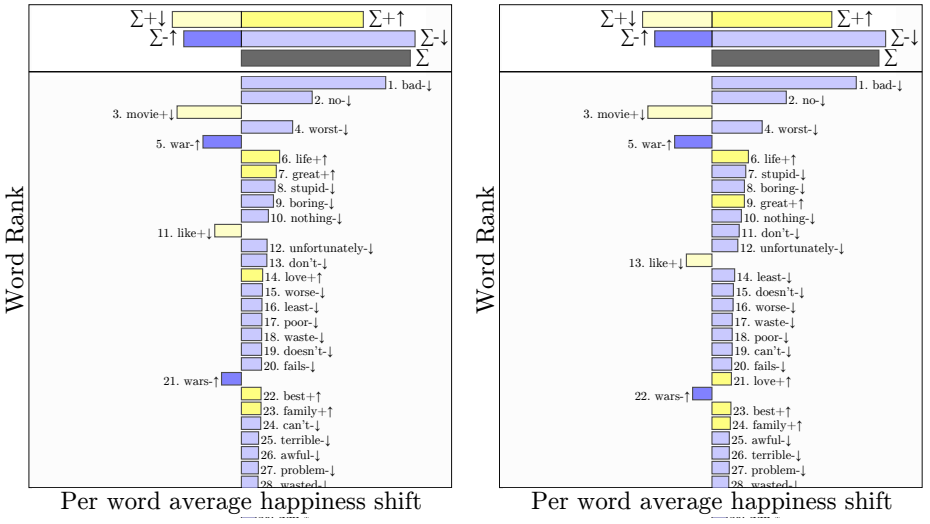

Figure S23 Word shift graph resulting from the 25% binarization of the labMT dictionary.

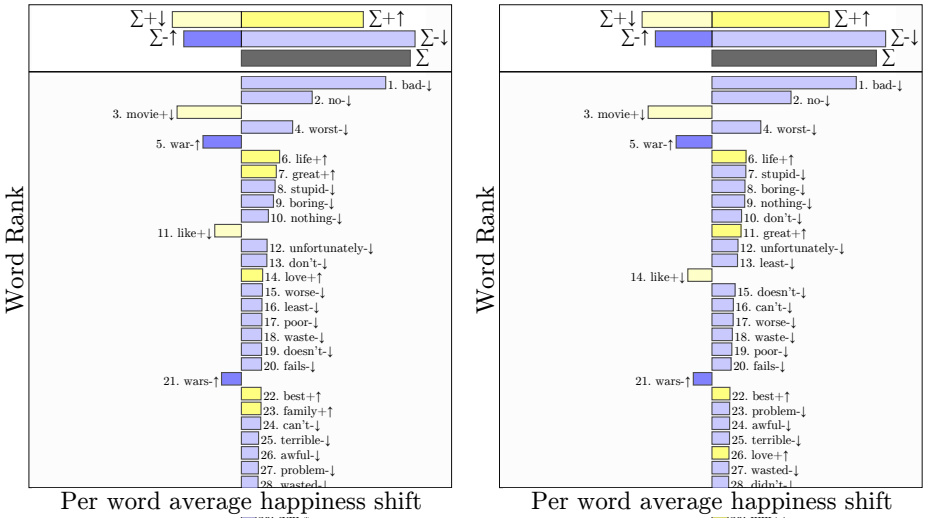

Figure S24 Word shift graph resulting from the 50% binarization of the labMT dictionary.

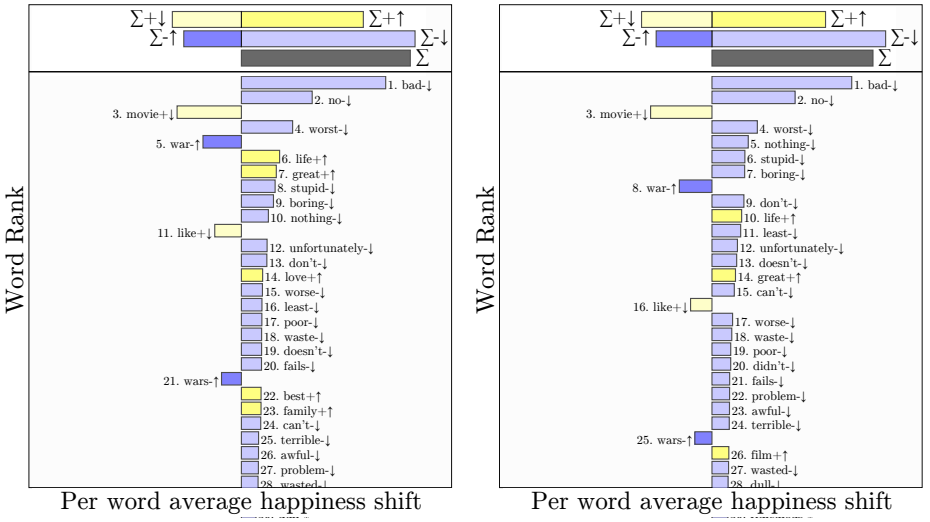

Figure S25 Word shift graph resulting from the 75% binarization of the labMT dictionary.

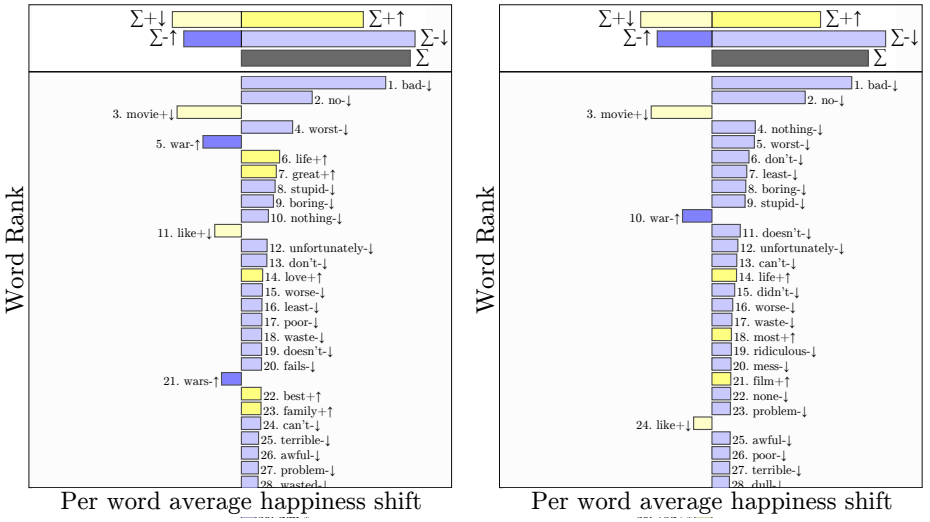

Figure S26 Word shift graph resulting from the full binarization of the labMT dictionary.

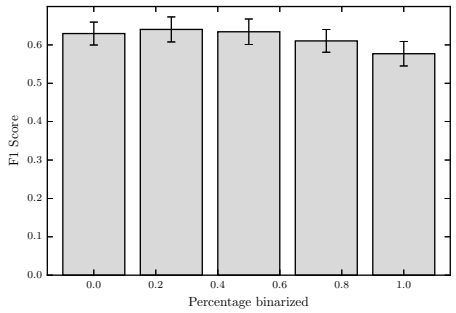

Figure S27 The direct binarization of the labMT dictionary results in a degradation of performance. The binarization is accomplished by linearly reducing the range of scores in the labMT dictionary from a centered -4 $\rightarrow$ 4 to the integer scores -1 and 1.

Reduced coverage

Second, to test the effect of coverage alone, we systematically reduce the coverage of the labMT dictionary and again attempt the binary classification task of identifying Movie Review polarity. Three possible strategies to reduce the coverage are (1) removing the most frequent words, (2) removing the least frequent words, and (3) removing words randomly (irrespective of their frequency of usage).

In Figs. S28–S46, we show the resulting word shift graphs with the control (all words included) alongside word shift graphs using the labMT dictionary with the least frequent (LF) and most frequent (MF) words removed. Each word shift graph with reduced coverage shows the number of words removed in parenthesis in the title, e.g., in Fig. S28 we see the titles "LF Reduced coverage (511)" and "MF Reduced coverage (511)" which indicate that 511 words were removed in the indicated fashion. We first observe that the difference in sentiment scores between the positive and negative movie reviews is decreased from 0.17 to 0.02–0.05 and 0.09–0.15 for the LF and MF strategies, respectively, while noting that these differences do not result in predictive accuracy (i.e., classification accuracy is not statistically significant worsened). Examining the words in Fig. S28 more closely, where only 5% of the words have been removed, we already observe departures in individual word contributions. Of the top 5 words in the control graph ("bad", "no", "movie", "worst", and "war"), we see only 3 of these in the top 5 for LF (all in the top 8) and only 1 in the top for MF (with 2 of the 5 showing on the graph at all). In the LF graph we lose words like "don't", "least", "doesn't", "terrible", "awful", "problem", and instead see the words "the", "of", "i", "is", "have" contribute more strongly. In the MF graph we lose common words like "best", "family", "love", "life", "like" and instead see the less common words "excellent", "perfect", "funny", "wonderful", "kill", "jokes", "beautiful", "dull", "performance", "annoying", and "lame". As one might expect, these trends of common/uncommon words varying across the word shifts graphs continue for increasingly reduced coverage.

With approximately half of the words from the labMT dictionary removed, in Fig. S37 we observe high overlap between the words in the control and LF, and only a single word in common between the control and MF word shift graphs. In addition to this, the sentiment score difference between the positive and negative reviews is 0.17 for the control, 0.04 for LF, and 0.14 for MF. In Fig. , only 1,024 (of 10,222) words remain in the LF and MF reduced coverage dictionaries, and again we see similar trends. Higher overlap exists between the LF and control, with only two words ("don't", "can't") in common between MF and control. While coverage remains above 50% for the LF strategy, the word shift graph shows more words that are weighting the classification incorrectly: "the", "i", "war", "like", etc. The MF word shift graph shows interesting words but also has many words that detracting from the classification: "i'm", "spice", "they're", "drunken", etc. We can conclude again, with these observations, that sentiment classification and sentiment understanding using word shifts graphs relies on broad coverage of the words used in the text being analyzed.

In Figures S47 and S48, we show the resulting F1 score of classification performance for each of these three strategies and the total coverage from each removal strategy. We observe that while certain strategies are more effective at retaining

performance, lower coverage scores are all lower despite substantial variation, and the overall pattern for each strategy is a decrease in performance for decreasing coverage. In both cases these results are consistent with those seen across dictionaries: integer scores and low coverage strongly reduce the performance of the 2-class movie review classification task, as measured by the F1-score. We note that this trend is not statistically significant, as can be observed with the standard deviation error bars.

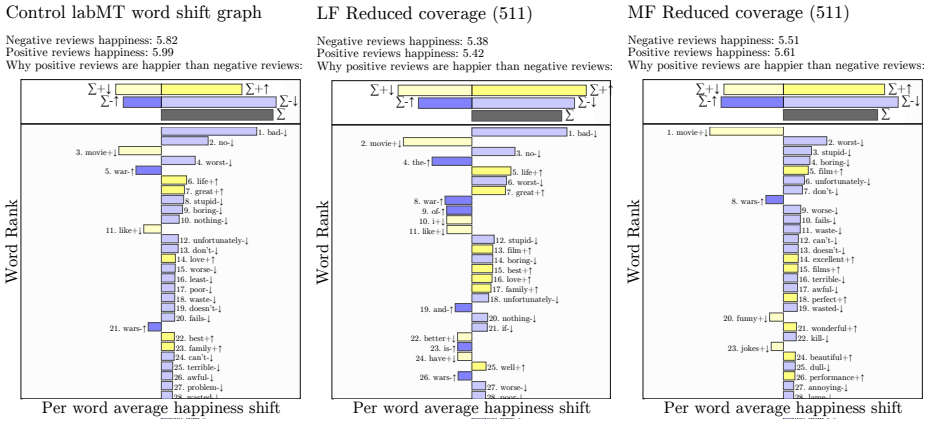

Figure S28 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

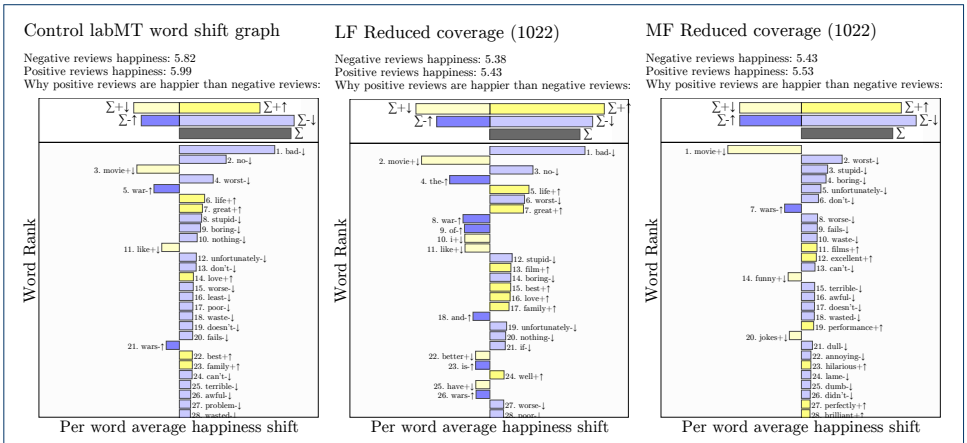

Figure S29 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

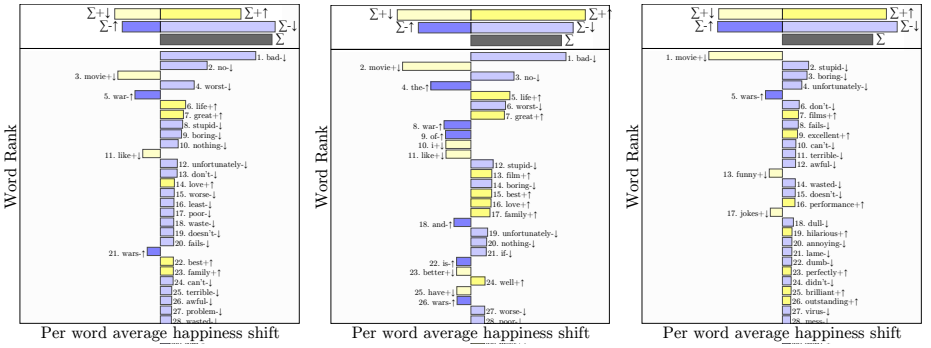

Figure S30 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

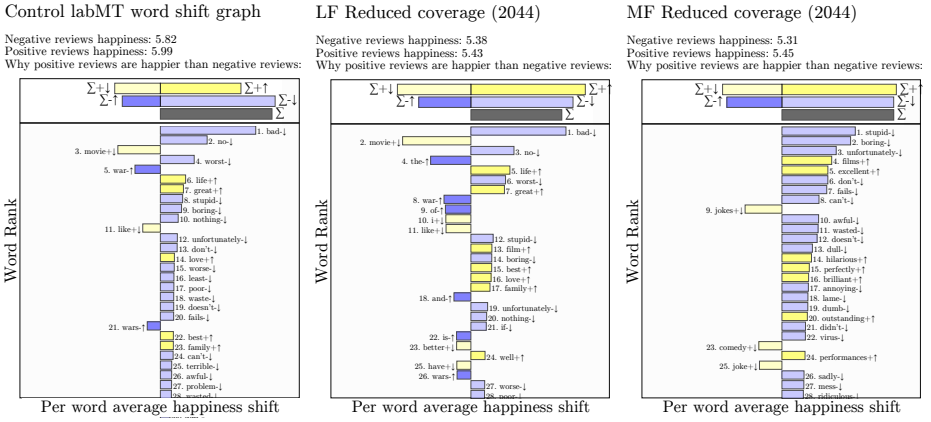

Figure S31 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

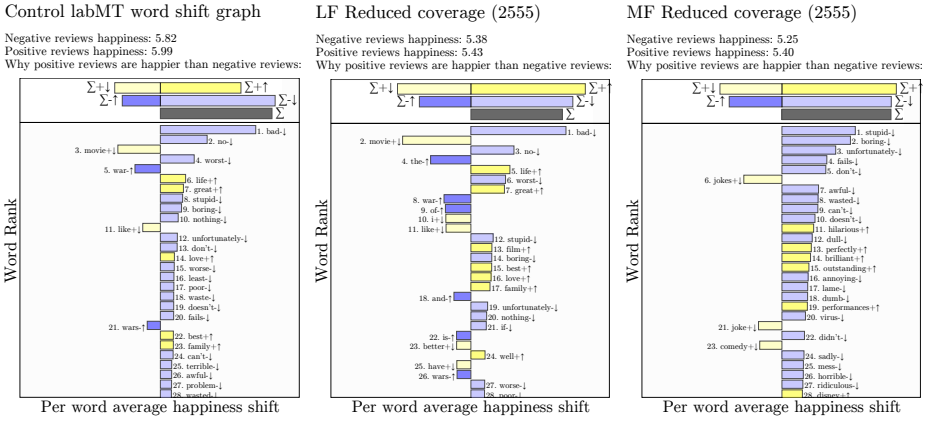

Figure S32 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

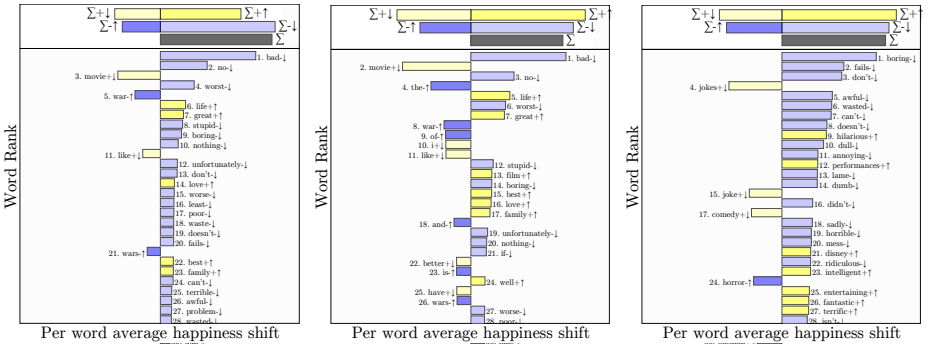

Figure S33 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

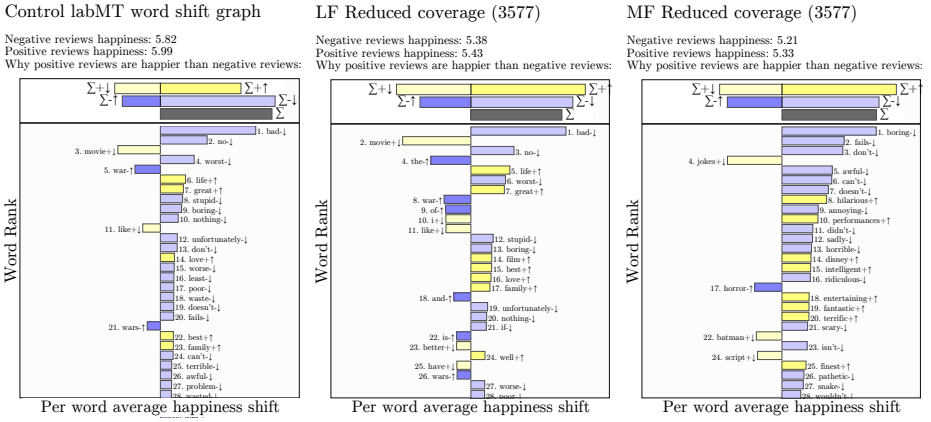

Figure S34 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

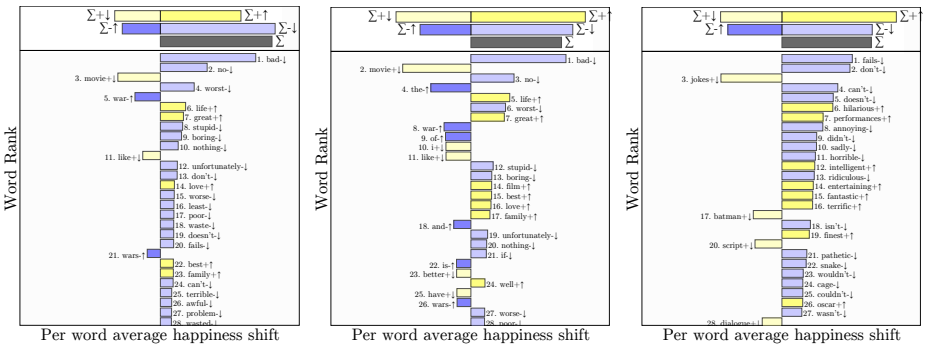

Figure S35 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

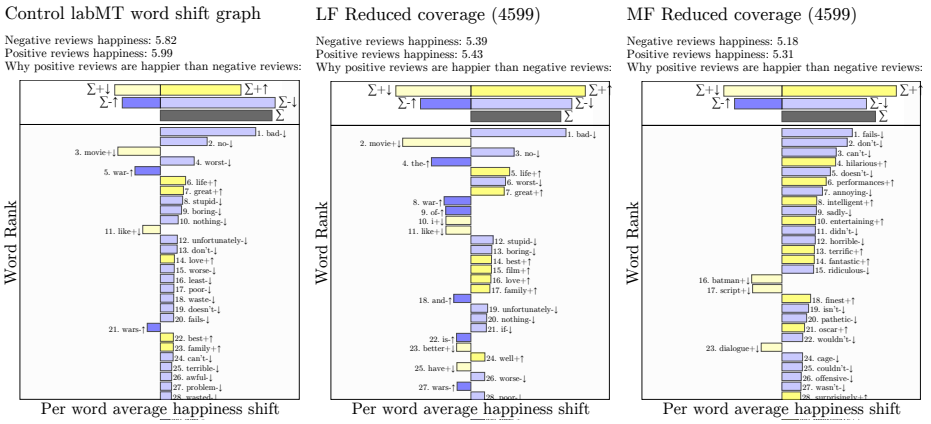

Figure S36 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

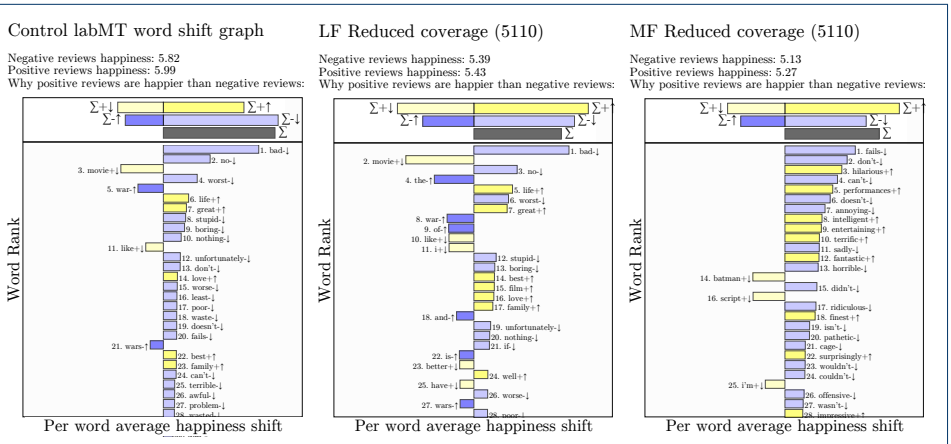

Figure S37 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

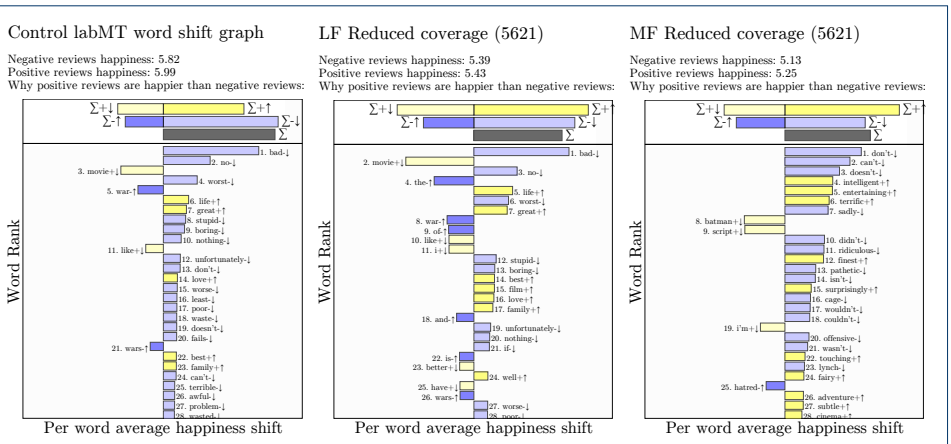

Figure S38 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

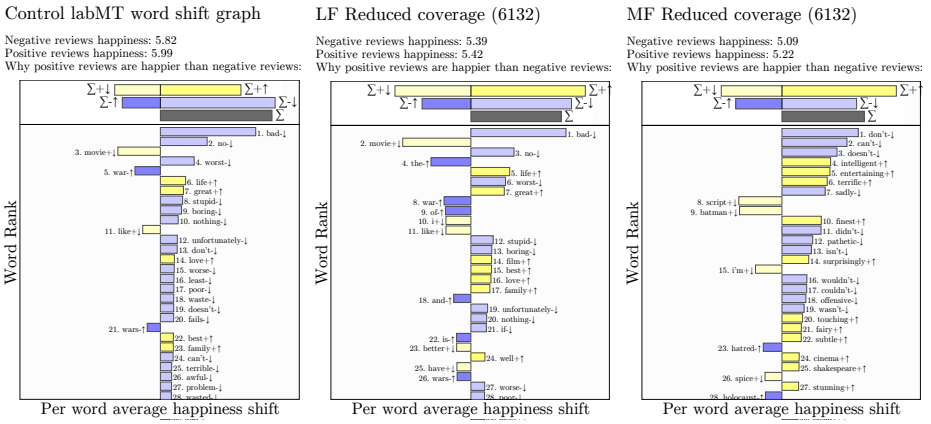

Figure S39 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

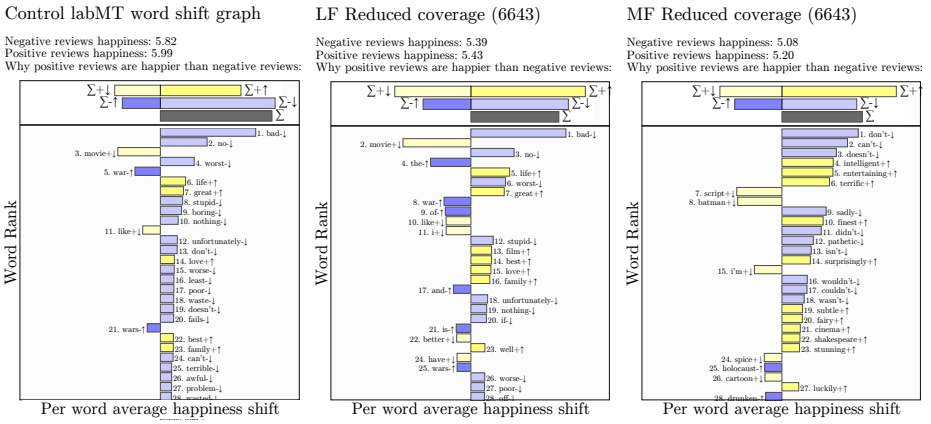

Figure S40 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

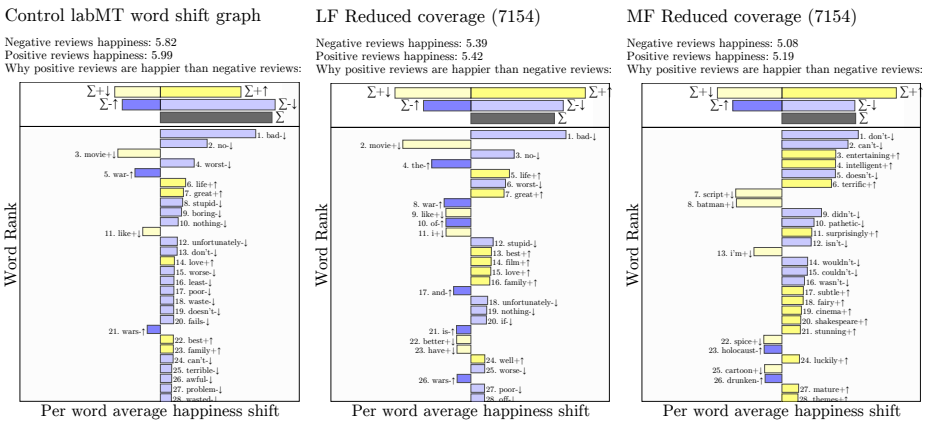

Figure S41 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

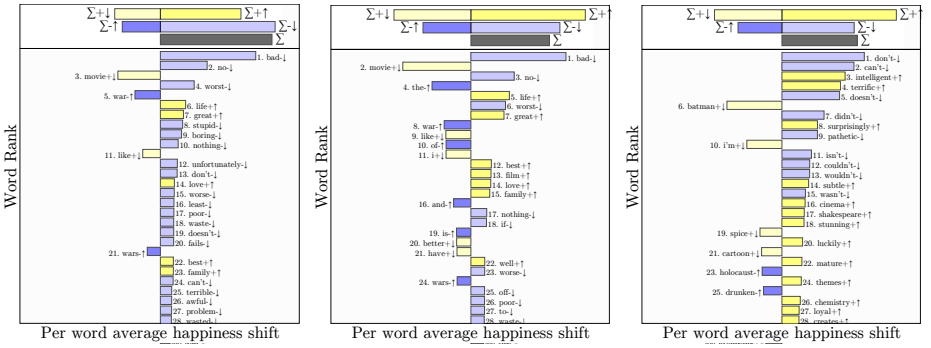

Figure S42 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

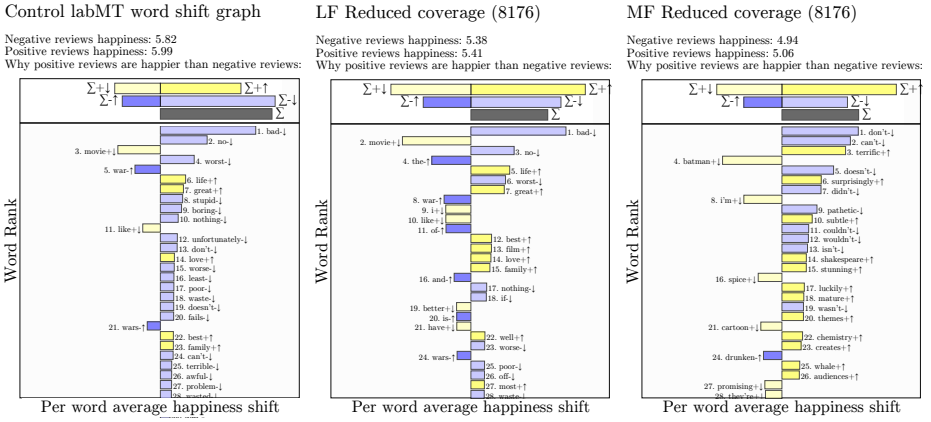

Figure S43 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

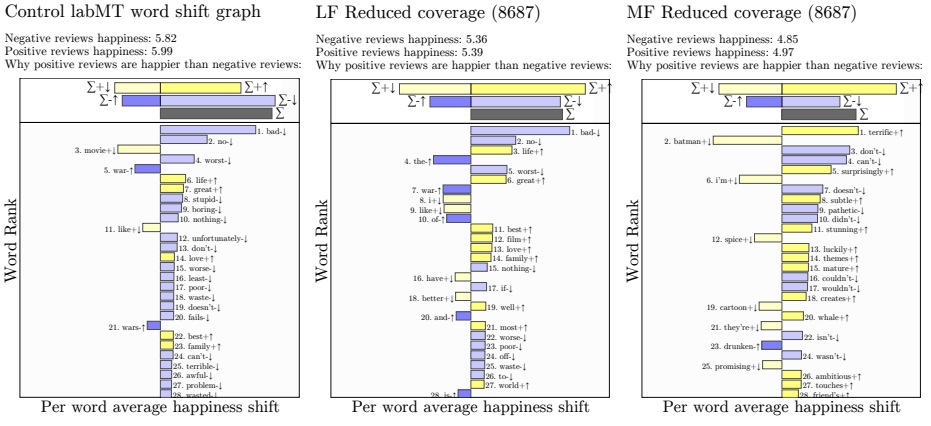

Figure S44 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

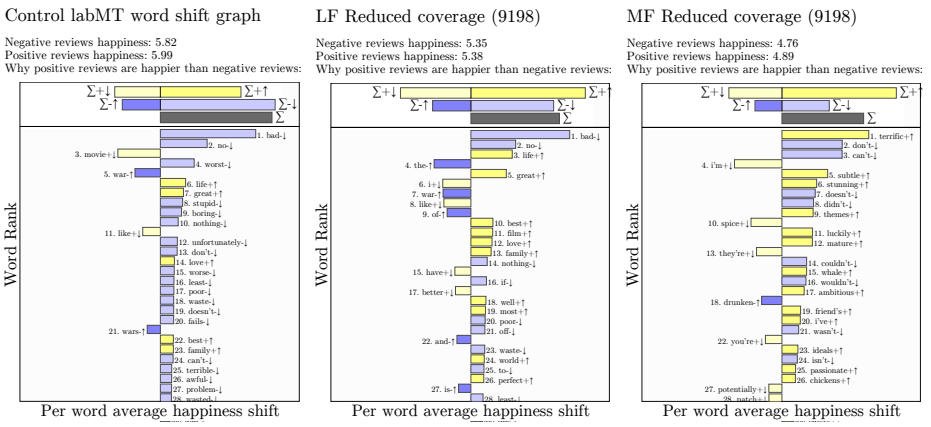

Figure S45 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

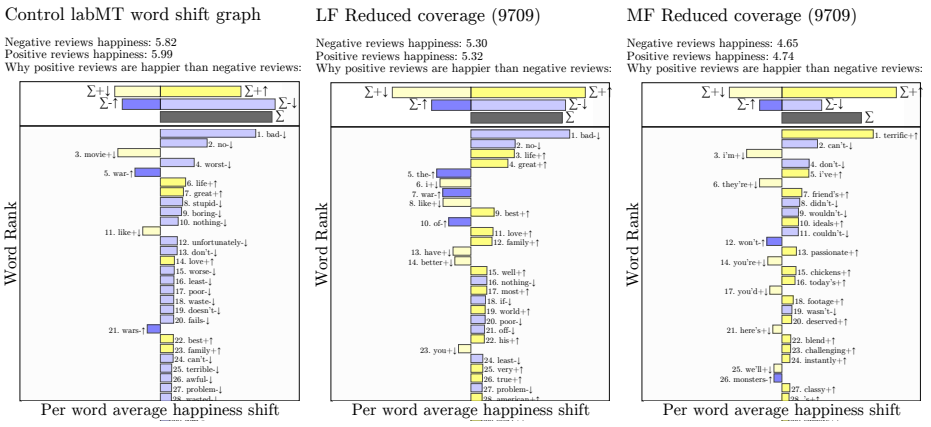

Figure S46 Word shift graphs resulting from the remove of the most frequent (MF) and least frequent (LF) words in the labMT dictionary.

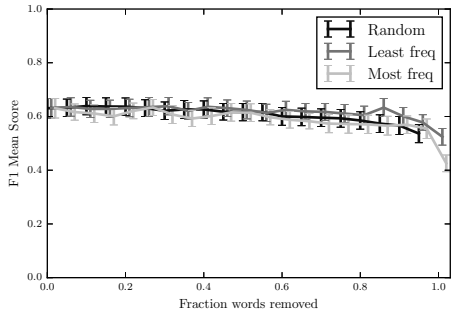

Figure S47 The resulting F1 score of classification performance for each of three coverage removal strategies. These strategies, labeled in the above, are: (1) removing the most frequent words, (2) removing the least frequent words, and (3) removing words randomly (irrespective of their frequency of usage). Error bars shown reflect the standard deviation of the F1 metric over 100 random samples.

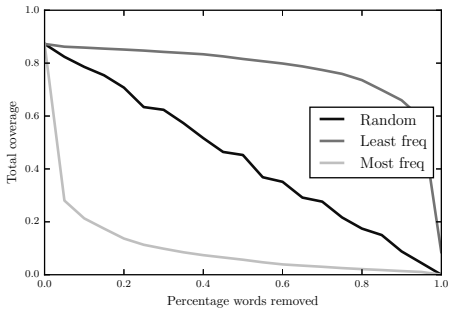

Figure S48 The resulting coverage for each of three coverage removal strategies. Again, these strategies, labeled in the above, are: (1) removing the most frequent words, (2) removing the least frequent words, and (3) removing words randomly (irrespective of their frequency of usage).