CHAPTER

# 20 Lexicons for Sentiment and Affect Extraction

*"[W]e write, not with the fingers, but with the whole person. The nerve which controls the pen winds itself about every fibre of our being, threads the heart, pierces the liver."*

Virginia Woolf, Orlando

*"She runs the gamut of emotions from A to B."*

Dorothy Parker, reviewing Hepburn's performance in *Little Women*

**affective**

In this chapter we turn to tools for interpreting **affective** meaning, extending our study of sentiment analysis in Chapter 4. We use the word 'affective', following the tradition in *affective computing* (Picard, 1995) to mean emotion, sentiment, personality, mood, and attitudes. Affective meaning is closely related to **subjectivity**, the study of a speaker or writer's evaluations, opinions, emotions, and speculations (Wiebe et al., 1999).

**subjectivity**

How should affective meaning be defined? One influential typology of affective states comes from Scherer (2000), who defines each class of affective states by factors like its cognition realization and time course:

**Emotion:** Relatively brief episode of response to the evaluation of an external or internal event as being of major significance.
(*angry, sad, joyful, fearful, ashamed, proud, elated, desperate*)

**Mood:** Diffuse affect state, most pronounced as change in subjective feeling, of low intensity but relatively long duration, often without apparent cause.
(*cheerful, gloomy, irritable, listless, depressed, buoyant*)

**Interpersonal stance:** Affective stance taken toward another person in a specific interaction, colouring the interpersonal exchange in that situation.
(*distant, cold, warm, supportive, contemptuous, friendly*)

**Attitude:** Relatively enduring, affectively colored beliefs, preferences, and predispositions towards objects or persons.
(*liking, loving, hating, valuing, desiring*)

**Personality traits:** Emotionally laden, stable personality dispositions and behavior tendencies, typical for a person.
(*nervous, anxious, reckless, morose, hostile, jealous*)

**Figure 20.1** The Scherer typology of affective states, with descriptions from Scherer (2000).

We can design extractors for each of these kinds of affective states. Chapter 4 already introduced *sentiment analysis*, the task of extracting the positive or negative orientation that a writer expresses toward some object. This corresponds in Scherer's typology to the extraction of **attitudes**: figuring out what people like or dislike, whether from consumer reviews of books or movies, newspaper editorials, or public sentiment from blogs or tweets.

Detecting **emotion** and **moods** is useful for detecting whether a student is confused, engaged, or certain when interacting with a tutorial system, whether a caller to a help line is frustrated, whether someone's blog posts or tweets indicated depression. Detecting emotions like fear in novels, for example, could help us trace what groups or situations are feared and how that changes over time.

Detecting different **interpersonal stances** can be useful when extracting information from human-human conversations. The goal here is to detect stances like friendliness or awkwardness in interviews or friendly conversations, or even to detect flirtation in dating. For the task of automatically summarizing meetings, we'd like to be able to automatically understand the social relations between people, who is friendly or antagonistic to whom. A related task is finding parts of a conversation where people are especially excited or engaged, conversational **hot spots** that can help a summarizer focus on the correct region.

Detecting the **personality** of a user—such as whether the user is an **extrovert** or the extent to which they are **open to experience**— can help improve conversational agents, which seem to work better if they match users' personality expectations (Mairesse and Walker, 2008).

Affect is important for generation as well as recognition; synthesizing affect is important for conversational agents in various domains, including literacy tutors such as children's storybooks, or computer games.

In Chapter 4 we introduced the use of Naive Bayes classification to classify a document's sentiment, an approach that has been successfully applied to many of these tasks. In that approach, all the words in the training set are used as features for classifying sentiment.

In this chapter we focus on an alternative model, in which instead of using every word as a feature, we focus only on certain words, ones that carry particularly strong cues to sentiment or affect. We call these lists of words **sentiment or affective lexicons**. In the next sections we introduce lexicons for sentiment, semi-supervised algorithms for inducing them, and simple algorithms for using lexicons to perform sentiment analysis.

We then turn to the extraction of other kinds of affective meaning, beginning with emotion, and the use of online tools for crowdsourcing emotion lexicons, and then proceeding to other kinds of affective meaning like interpersonal stance and personality.

## 20.1 Available Sentiment Lexicons

The most basic lexicons label words along one dimension of semantic variability, called "sentiment", "valence", or "semantic orientation".

**General Inquirer**

In the simplest lexicons this dimension is represented in a binary fashion, with a wordlist for positive words and a wordlist for negative words. The oldest is the **General Inquirer** (Stone et al., 1966), which drew on early work in the cognition psychology of word meaning (Osgood et al., 1957) and on work in content analysis.

The General Inquirer is a freely available web resource with lexicons of 1915 positive words and 2291 negative words (and also includes other lexicons we'll discuss in the next section).

The MPQA Subjectivity lexicon (Wilson et al., 2005) has 2718 positive and 4912 negative words drawn from a combination of sources, including the General Inquirer lists, the output of the Hatzivassiloglou and McKeown (1997) system described below, and a bootstrapped list of subjective words and phrases (Riloff and Wiebe, 2003) that was then hand-labeled for sentiment. Each phrase in the lexicon is also labeled for reliability (strongly subjective or weakly subjective). The polarity lexicon of (Hu and Liu, 2004) gives 2006 positive and 4783 negative words, drawn from product reviews, labeled using a bootstrapping method from WordNet described in the next section.

| | |
|---|---|
| **Positive** | admire, amazing, assure, celebration, charm, eager, enthusiastic, excellent, fancy, fantastic, frolic, graceful, happy, joy, luck, majesty, mercy, nice, patience, perfect, proud, rejoice, relief, respect, satisfactorily, sensational, super, terrific, thank, vivid, wise, wonderful, zest |
| **Negative** | abominable, anger, anxious, bad, catastrophe, cheap, complaint, condescending, deceit, defective, disappointment, embarrass, fake, fear, filthy, fool, guilt, hate, idiot, inflict, lazy, miserable, mourn, nervous, objection, pest, plot, reject, scream, silly, terrible, unfriendly, vile, wicked |

**Figure 20.2** Some samples of words with consistent sentiment across three sentiment lexicons: the General Inquirer (Stone et al., 1966), the MPQA Subjectivity lexicon (Wilson et al., 2005), and the polarity lexicon of Hu and Liu (2004).

## 20.2 Semi-supervised induction of sentiment lexicons

Some affective lexicons are built by having humans assign ratings to words; this was the technique for building the General Inquirer starting in the 1960s (Stone et al., 1966), and for modern lexicons based on crowd-sourcing to be described in Section 20.5.1. But one of the most powerful ways to learn lexicons is to use semi-supervised learning.

In this section we introduce three methods for semi-supervised learning that are important in sentiment lexicon extraction. The three methods all share the same intuitive algorithm which is sketched in Fig. 20.3.

**function** BUILDSENTIMENTLEXICON(*posseeds*,*negseeds*) **returns** *poslex*,*neglex*

*poslex* ← *posseeds*
*neglex* ← *negseeds*
**Until done**
  *poslex* ← *poslex* + FINDSIMILARWORDS(*poslex*)
  *neglex* ← *neglex* + FINDSIMILARWORDS(*neglex*)
*poslex*,*neglex* ← POSTPROCESS(*poslex*,*neglex*)

**Figure 20.3** Schematic for semi-supervised sentiment lexicon induction. Different algorithms differ in the how words of similar polarity are found, in the stopping criterion, and in the post-processing.

As we will see, the methods differ in the intuitions they use for finding words with similar polarity, and in steps they take to use machine learning to improve the quality of the lexicons.

### 20.2.1 Using seed words and adjective coordination

The Hatzivassiloglou and McKeown (1997) algorithm for labeling the polarity of adjectives is the same semi-supervised architecture described above. Their algorithm has four steps.

**Step 1: Create seed lexicon:** Hand-label a seed set of 1336 adjectives (all words that occurred more than 20 times in the 21 million word WSJ corpus). They labeled 657 positive adjectives (e.g., *adequate, central, clever, famous, intelligent, remarkable, reputed, sensitive, slender, thriving*) and 679 negative adjectives (e.g., *contagious, drunken, ignorant, lanky, listless, primitive, strident, troublesome, unresolved, unsuspecting*).

**Step 2: Find cues to candidate similar words:** Choose words that are similar or different to the seed words, using the intuition that adjectives conjoined by the words *and* tend to have the same polarity. Thus we might expect to see instances of positive adjectives coordinated with positive, or negative with negative:

fair and legitimate, corrupt and brutal

but less likely to see positive adjectives coordinated with negative:

*fair and brutal, *corrupt and legitimate

By contrast, adjectives conjoined by *but* are likely to be of opposite polarity:

fair but brutal

The idea that simple patterns like coordination via *and* and *but* are good tools for finding lexical relations like same-polarity and opposite-polarity is an application of the pattern-based approach to relation extraction described in Chapter 17.

Another cue to opposite polarity comes from morphological negation (*un-*, *im-*, *-less*). Adjectives with the same root but differing in a morphological negative (*adequate/inadequate*, *thoughtful/thoughtless*) tend to be of opposite polarity.

**Step 3: Build a polarity graph**

These cues are integrated by building a graph with nodes for words and links representing how likely the two words are to have the same polarity, as shown in Fig. 20.4.

A simple way to build a graph would predict an opposite-polarity link if the two adjectives are connected by at least one *but*, and a same-polarity link otherwise (for any two adjectives connected by at least one conjunction). The more sophisticated method used by Hatzivassiloglou and McKeown (1997) is to build a supervised classifier that predicts whether two words are of the same or different polarity, by using these 3 features (occurrence with *and*, occurrence with *but*, and morphological negations).

The classifier is trained on a subset of the hand-labeled seed words, and returns a probability that each pair of words is of the same or opposite polarity. This 'polarity similarity' of each word pair can be viewed as the strength of the positive or negative links between them in a graph.

**Step 4: Clustering the graph** Finally, any of various graph clustering algorithms can be used to divide the graph into two subsets with the same polarity; a graphical intuition is shown in Fig. 20.5.

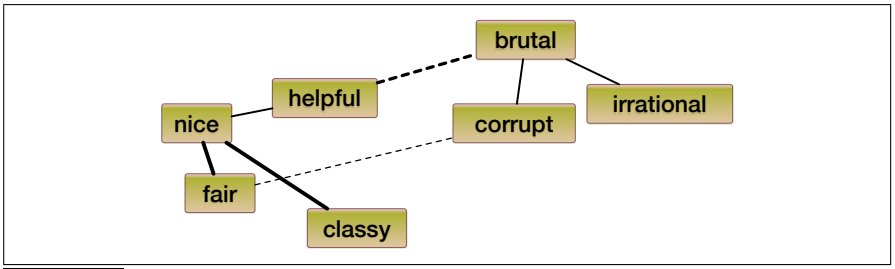


**Figure 20.4** A graph of polarity similarity between all pairs of words; words are notes and links represent polarity association between words. Continuous lines are same-polarity and dotted lines are opposite-polarity; the width of lines represents the strength of the polarity.

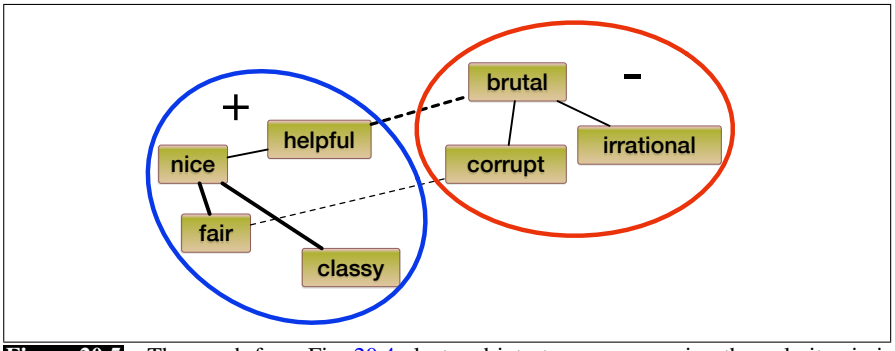


**Figure 20.5** The graph from Fig. 20.4 clustered into two groups, using the polarity similarity between two words (visually represented as the edge line strength and continuity) as a distance metric for clustering.

Some sample output from the Hatzivassiloglou and McKeown (1997) algorithm is shown below, showing system errors in red.

> **Positive:** bold decisive disturbing generous good honest important large mature patient peaceful positive proud sound stimulating straightforward strange talented vigorous witty
>
> **Negative:** ambiguous cautious cynical evasive harmful hypocritical inefficient insecure irrational irresponsible minor outspoken pleasant reckless risky selfish tedious unsupported vulnerable wasteful

### 20.2.2 Pointwise mutual information

Where the first method for finding words with similar polarity relied on patterns of conjunction, we turn now to a second method that uses neighborhood co-occurrence as proxy for polarity similarity. This algorithm assumes that words with similar polarity tend to occur nearby each other, using the pointwise mutual information (PMI) algorithm defined in Chapter 6.

The method of Turney (2002) uses this method to assign polarity to both words and two-word phrases.

In a prior step, two-word phrases are extracted based on simple part-of-speech regular expressions. The expressions select nouns with preceding adjectives, verbs with preceding adverbs, and adjectival heads (adjectives with no following noun) preceded by adverbs, adjectives or nouns:

| Word 1 POS | Word 2 POS |
|---|---|
| JJ | NN\|NNS |
| RB\|RBR\|RBS | VB\|VBD\|VBN\|VBG |
| RB\|RBR\|RBS\|JJ\|NN\|NNS | JJ (only if following word is not NN\|NNS) |

To measure the polarity of each extracted phrase, we start by choosing positive and negative seed words. For example we might choose a single positive seed word *excellent* and a single negative seed word *poor*. We then make use of the intuition that positive phrases will in general tend to co-occur more with *excellent*. Negative phrases co-occur more with *poor*.

**pointwise mutual information**

The PMI measure can be used to measure this co-occurrence. Recall from Chapter 6 that the **pointwise mutual information** (Fano, 1961) is a measure of how often two events $x$ and $y$ occur, compared with what we would expect if they were independent:

$$PMI(x,y) = \log_2 \frac{P(x,y)}{P(x)P(y)} \tag{20.1}$$

This intuition can be applied to measure the co-occurrence of two words by defining the pointwise mutual information association between a seed word $s$ and another word $w$ as:

$$\text{PMI}(w,s) = \log_2 \frac{P(w,s)}{P(w)P(s)} \tag{20.2}$$

Turney (2002) estimated the probabilities needed by Eq. 20.2 using a search engine with a `NEAR` operator, specifying that a word has to be *near* another word. The probabilities are then estimated as follows:

$$P(w) = \frac{\text{hits}(w)}{N} \tag{20.3}$$

$$P(w1,w2) = \frac{\text{hits}(w1 \text{ NEAR } w2)}{kN} \tag{20.4}$$

That is, we estimate the probability of a word as the count returned from the search engine, normalized by the total number of words in the entire web corpus $N$. (It doesn't matter that we don't know what $N$ is, since it turns out it will cancel out nicely). The bigram probability is the number of bigram hits normalized by $kN$—although there are $N$ unigrams and also approximately $N$ bigrams in a corpus of length $N$, there are $kN$ "NEAR" bigrams in which the two words are separated by a distance of up to $k$.

The PMI between two words $w$ and $s$ is then:

$$\text{PMI}(w,s) = \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR } s)}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(s)} \tag{20.5}$$

The insight of Turney (2002) is then to define the polarity of a word by how much it occurs with the positive seeds and doesn't occur with the negative seeds:

$$\begin{aligned}
\text{Polarity}(w) &= \text{PMI}(w, \text{“excellent”}) - \text{PMI}(w, \text{“poor”}) \\
&= \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR “excellent”})}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(\text{“excellent”})} - \log_2 \frac{\frac{1}{kN}\text{hits}(w \text{ NEAR “poor”})}{\frac{1}{N}\text{hits}(w)\frac{1}{N}\text{hits}(\text{“poor”})} \\
&= \log_2 \left( \frac{\text{hits}(w \text{ NEAR “excellent”})}{\text{hits}(w)\,\text{hits}(\text{“excellent”})} \frac{\text{hits}(w)\,\text{hits}(\text{“poor”})}{\text{hits}(w \text{ NEAR “poor”})} \right) \\
&= \log_2 \left( \frac{\text{hits}(w \text{ NEAR “excellent”})\ \text{hits}(\text{“poor”})}{\text{hits}(\text{“excellent”})\ \text{hits}(w \text{ NEAR “poor”})} \right) \qquad (20.6)
\end{aligned}$$

The table below from Turney (2002) shows sample examples of phrases learned by the PMI method (from reviews of banking services), showing those with both positive and negative polarity:

| Extracted Phrase | Polarity |
|---|---|
| online experience | 2.3 |
| very handy | 1.4 |
| low fees | 0.3 |
| inconveniently located | -1.5 |
| other problems | -2.8 |
| unethical practices | -8.5 |

### 20.2.3 Using WordNet synonyms and antonyms

A third method for finding words that have a similar polarity to seed words is to make use of word synonymy and antonymy. The intuition is that a word's synonyms probably share its polarity while a word's antonyms probably have the opposite polarity.

Since WordNet has these relations, it is often used (Kim and Hovy 2004, Hu and Liu 2004). After a seed lexicon is built, each lexicon is updated as follows, possibly iterated.

$\text{Lex}^+$ : Add synonyms of positive words (*well*) and antonyms (like *fine*) of negative words

$\text{Lex}^-$ : Add synonyms of negative words (*awful*) and antonyms ( like *evil*) of positive words

SentiWordNet

An extension of this algorithm has been applied to assign polarity to WordNet senses, called **SentiWordNet** (Baccianella et al., 2010). Fig. 20.6 shows some examples.

In this algorithm, polarity is assigned to entire synsets rather than words. A positive lexicon is built from all the synsets associated with 7 positive words, and a negative lexicon from synsets associated with 7 negative words. Both are expanded by drawing in synsets related by WordNet relations like antonymy or see-also. A classifier is then trained from this data to take a WordNet gloss and decide if the sense being defined is positive, negative or neutral. A further step (involving a random-walk algorithm) assigns a score to each WordNet synset for its degree of positivity, negativity, and neutrality.

In summary, we've seen three distinct ways to use semisupervised learning to induce a sentiment lexicon. All begin with a seed set of positive and negative words, as small as 2 words (Turney, 2002) or as large as a thousand (Hatzivassiloglou and

| Synset | Pos | Neg | Obj |
|---|---|---|---|
| good#6 'agreeable or pleasing' | 1 | 0 | 0 |
| respectable#2 honorable#4 good#4 estimable#2 'deserving of esteem' | 0.75 | 0 | 0.25 |
| estimable#3 computable#1 'may be computed or estimated' | 0 | 0 | 1 |
| sting#1 burn#4 bite#2 'cause a sharp or stinging pain' | 0 | 0.875 | .125 |
| acute#6 'of critical importance and consequence' | 0.625 | 0.125 | .250 |
| acute#4 'of an angle; less than 90 degrees' | 0 | 0 | 1 |
| acute#1 'having or experiencing a rapid onset and short but severe course' | 0 | 0.5 | 0.5 |

**Figure 20.6** Examples from SentiWordNet 3.0 (Baccianella et al., 2010). Note the differences between senses of homonymous words: *estimable#3* is purely objective, while *estimable#2* is positive; *acute* can be positive (*acute#6*), negative (*acute#1*), or neutral (*acute #4*)

McKeown, 1997). More words of similar polarity are then added, using pattern-based methods, PMI-weighted document co-occurrence, or WordNet synonyms and antonyms. Classifiers can also be used to combine various cues to the polarity of new words, by training on the seed training sets, or early iterations.

## 20.3 Supervised learning of word sentiment

The previous section showed semi-supervised ways to learn sentiment when there is no supervision signal, by expanding a hand-built seed set using cues to polarity similarity. An alternative to semi-supervision is to do supervised learning, making direct use of a powerful source of supervision for word sentiment: *online reviews*.

The web contains an enormous number of online reviews for restaurants, movies, books, or other products, each of which have the text of the review along with an associated review score: a value that may range from 1 star to 5 stars, or scoring 1 to 10. Fig. 20.7 shows samples extracted from restaurant, book, and movie reviews.

We can use this review score as supervision: positive words are more likely to appear in 5-star reviews; negative words in 1-star reviews. And instead of just a binary polarity, this kind of supervision allows us to assign a word a more complex representation of its polarity: its distribution over stars (or other scores).

Thus in a ten-star system we could represent the sentiment of each word as a 10-tuple, each number a score representing the word's association with that polarity level. This association can be a raw count, or a likelihood $P(w|c)$, or some other function of the count, for each class $c$ from 1 to 10.

For example, we could compute the IMDB likelihood of a word like *disappoint(ed/ing)* occuring in a 1 star review by dividing the number of times *disappoint(ed/ing)* occurs in 1-star reviews in the IMDB dataset (8,557) by the total number of words occurring in 1-star reviews (25,395,214), so the IMDB estimate of $P(disappointing|1)$ is .0003.

A slight modification of this weighting, the normalized likelihood, can be used as an illuminating visualization (Potts, 2011)[1]:

[1] Potts shows that the normalized likelihood is an estimate of the posterior $P(c|w)$ if we make the incorrect but simplifying assumption that all categories $c$ have equal probability.

| | |
|---|---|
| | **Movie review excerpts (IMDB)** |
| **10** | A great movie. This film is just a wonderful experience. It's surreal, zany, witty and slapstick all at the same time. And terrific performances too. |
| **1** | This was probably the worst movie I have ever seen. The story went nowhere even though they could have done some interesting stuff with it. |
| | **Restaurant review excerpts (Yelp)** |
| **5** | The service was impeccable. The food was cooked and seasoned perfectly... The watermelon was perfectly square ... The grilled octopus was ... mouthwatering... |
| **2** | ...it took a while to get our waters, we got our entree before our starter, and we never received silverware or napkins until we requested them... |
| | **Book review excerpts (GoodReads)** |
| **1** | I am going to try and stop being deceived by eye-catching titles. I so wanted to like this book and was so disappointed by it. |
| **5** | This book is hilarious. I would recommend it to anyone looking for a satirical read with a romantic twist and a narrator that keeps butting in |
| | **Product review excerpts (Amazon)** |
| **5** | The lid on this blender though is probably what I like the best about it... enables you to pour into something without even taking the lid off! ... the perfect pitcher! ... works fantastic. |
| **1** | I hate this blender... It is nearly impossible to get frozen fruit and ice to turn into a smoothie... You have to add a TON of liquid. I also wish it had a spout ... |

**Figure 20.7** Excerpts from some reviews from various review websites, all on a scale of 1 to 5 stars except IMDB, which is on a scale of 1 to 10 stars.

$$P(w|c) = \frac{count(w,c)}{\sum_{w \in C} count(w,c)}$$
$$PottsScore(w) = \frac{P(w|c)}{\sum_c P(w|c)} \tag{20.7}$$

Dividing the IMDB estimate $P(disappointing|1)$ of .0003 by the sum of the likelihood $P(w|c)$ over all categories gives a Potts score of 0.10. The word *disappointing* thus is associated with the vector [.10, .12, .14, .14, .13, .11, .08, .06, .06, .05]. The

Potts diagram

**Potts diagram** (Potts, 2011) is a visualization of these word scores, representing the prior sentiment of a word as a distribution over the rating categories.

Fig. 20.8 shows the Potts diagrams for 3 positive and 3 negative scalar adjectives. Note that the curve for strongly positive scalars have the shape of the letter J, while strongly negative scalars look like a reverse J. By contrast, weakly positive and negative scalars have a hump-shape, with the maximum either below the mean (weakly negative words like *disappointing*) or above the mean (weakly positive words like *good*). These shapes offer an illuminating typology of affective word meaning.

Fig. 20.9 shows the Potts diagrams for emphasizing and attenuating adverbs. Again we see generalizations in the characteristic curves associated with words of particular meanings. Note that emphatics tend to have a J-shape (most likely to occur in the most positive reviews) or a U-shape (most likely to occur in the strongly positive and negative). Attenuators all have the hump-shape, emphasizing the middle of the scale and downplaying both extremes.

The diagrams can be used both as a typology of lexical sentiment, and also play a role in modeling sentiment compositionality.

In addition to functions like posterior $P(c|w)$, likelihood $P(w|c)$, or normalized

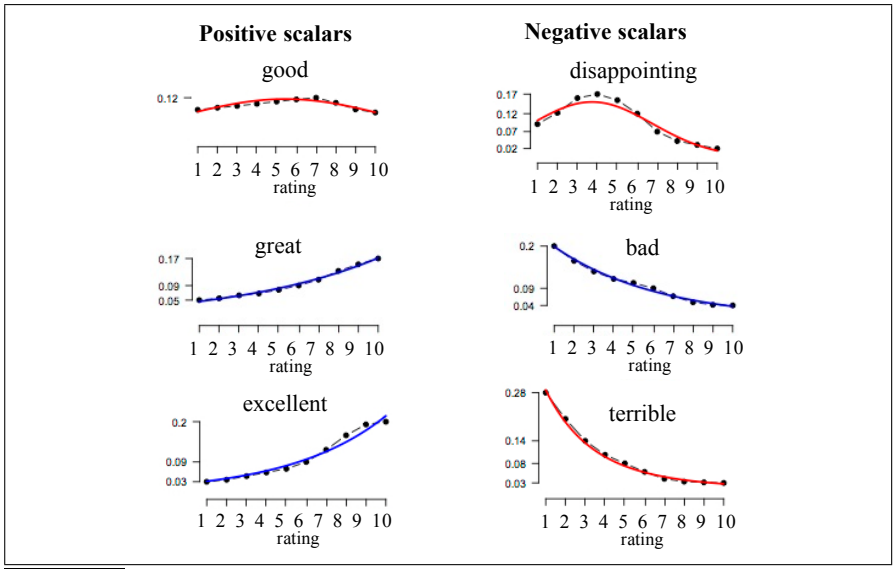


**Figure 20.8** Potts diagrams (Potts, 2011) for positive and negative scalar adjectives, showing the J-shape and reverse J-shape for strongly positive and negative adjectives, and the hump-shape for more weakly polarized adjectives.

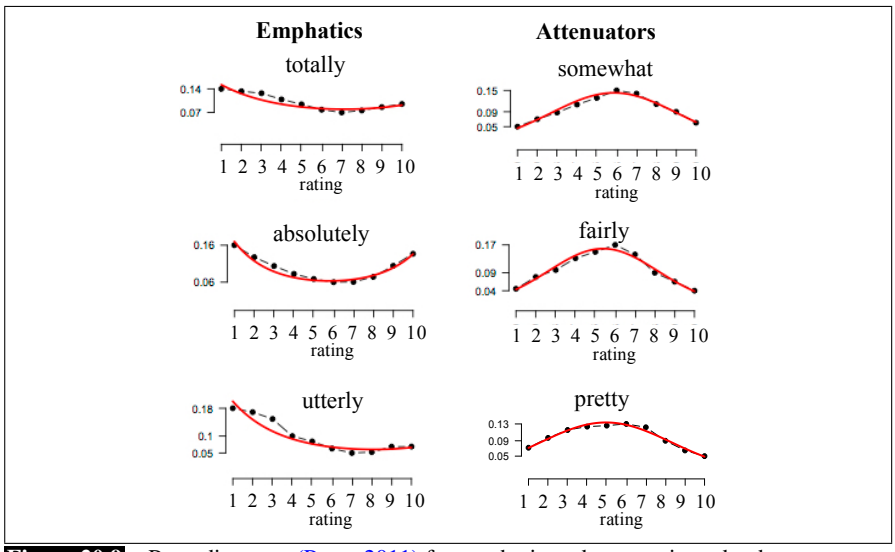


**Figure 20.9** Potts diagrams (Potts, 2011) for emphatic and attenuating adverbs.

likelihood (Eq. 20.7) many other functions of the count of a word occuring with a sentiment label have been used. We'll introduce some of these on page 18, including ideas like normalizing the counts per writer in Eq. 20.15.

### 20.3.1 Log odds ratio informative Dirichlet prior

One thing we often want to do with word polarity is to distinguish between words that are more likely to be used in one category of texts than in another. We may, for example, want to know the words most associated with 1 star reviews versus those associated with 5 star reviews. These differences may not be just related to sentiment. We might want to find words used more often by Democratic than Republican

members of Congress, or words used more often in menus of expensive restaurants than cheap restaurants.

Given two classes of documents, to find words more associated with one category than another, we might choose to just compute the difference in frequencies (is a word $w$ more frequent in class $A$ or class $B$?). Or instead of the difference in frequencies we might want to compute the ratio of frequencies, or the log odds ratio (the log of the ratio between the odds of the two words). Then we can sort words by whichever of these associations with the category we use, (sorting from words overrepresented in category $A$ to words overrepresented in category $B$).

The problem with simple log-likelihood or log odds methods is that they don't work well for very rare words or very frequent words; for words that are very frequent, all differences seem large, and for words that are very rare, no differences seem large.

In this section we walk through the details of one solution ot this problem ¡ the "log odds ratio informative Dirichlet prior" method of Monroe et al. (2008) that is a particularly useful method for finding words that are statistically overrepresented in one particular category of texts compared to another. It's based on the idea of using another large corpus to get a prior estimate of what we expect the frequency of each word to be.

log likelihood ratio

Let's start with the goal: assume we want to know whether the word *horrible* occurs more in corpus $i$ or corpus $j$. We could compute the **log likelihood ratio**, using $f^i(w)$ to mean the frequency of word $w$ in corpus $i$, and $n^i$ to mean the total number of words in corpus $i$:

$$\begin{aligned}\text{llr}(horrible) &= \log \frac{P^i(horrible)}{P^j(horrible)} \\ &= \log P^i(horrible) - \log P^j(horrible) \\ &= \log \frac{\text{f}^i(horrible)}{n^i} - \log \frac{\text{f}^j(horrible)}{n^j} \end{aligned} \tag{20.8}$$

log odds ratio

Instead, let's compute the **log odds ratio**: does *horrible* have higher odds in $i$ or in $j$:

$$\begin{aligned}\text{lor}(horrible) &= \log\left(\frac{P^i(horrible)}{1-P^i(horrible)}\right) - \log\left(\frac{P^j(horrible)}{1-P^j(horrible)}\right) \\ &= \log\left(\frac{\frac{\text{f}^i(horrible)}{n^i}}{1-\frac{\text{f}^i(horrible)}{n^i}}\right) - \log\left(\frac{\frac{\text{f}^j(horrible)}{n^j}}{1-\frac{\text{f}^j(horrible)}{n^j}}\right) \\ &= \log\left(\frac{\text{f}^i(horrible)}{n^i-\text{f}^i(horrible)}\right) - \log\left(\frac{\text{f}^j(horrible)}{n^j-\text{f}^j(horrible)}\right) \end{aligned} \tag{20.9}$$

The Dirichlet intuition is to use a large background corpus to get a prior estimate of what we expect the frequency of each word $w$ to be. We'll do this very simply by adding the counts from that corpus to the numerator and denominator, so that we're essentially shrinking the counts toward that prior. It's like asking how large are the differences between $i$ and $j$ given what we would expect given their frequencies in a well-estimated large background corpus.

The method estimates the difference between the frequency of word $w$ in two corpora $i$ and $j$ via the prior-modified log odds ratio for $w$, $\delta_w^{(i-j)}$, which is estimated

as:

$$\delta_w^{(i-j)} = \log\left(\frac{f_w^i + \alpha_w}{n^i + \alpha_0 - (f_w^i + \alpha_w)}\right) - \log\left(\frac{f_w^j + \alpha_w}{n^j + \alpha_0 - (f_w^j + \alpha_w)}\right) \tag{20.10}$$

(where $n^i$ is the size of corpus $i$, $n^j$ is the size of corpus $j$, $f_w^i$ is the count of word $w$ in corpus $i$, $f_w^j$ is the count of word $w$ in corpus $j$, $\alpha_0$ is the size of the background corpus, and $\alpha_w$ is the count of word $w$ in the background corpus.)

In addition, Monroe et al. (2008) make use of an estimate for the variance of the log–odds–ratio:

$$\sigma^2\left(\hat{\delta}_w^{(i-j)}\right) \approx \frac{1}{f_w^i + \alpha_w} + \frac{1}{f_w^j + \alpha_w} \tag{20.11}$$

The final statistic for a word is then the z–score of its log–odds–ratio:

$$\frac{\hat{\delta}_w^{(i-j)}}{\sqrt{\sigma^2\left(\hat{\delta}_w^{(i-j)}\right)}} \tag{20.12}$$

The Monroe et al. (2008) method thus modifies the commonly used log odds ratio in two ways: it uses the z-scores of the log odds ratio, which controls for the amount of variance in a words frequency, and it uses counts from a background corpus to provide a prior count for words.

Fig. 20.10 shows the method applied to a dataset of restaurant reviews from Yelp, comparing the words used in 1-star reviews to the words used in 5-star reviews (Jurafsky et al., 2014). The largest difference is in obvious sentiment words, with the 1-star reviews using negative sentiment words like *worse, bad, awful* and the 5-star reviews using positive sentiment words like *great, best, amazing*. But there are other illuminating differences. 1-star reviews use logical negation (*no, not*), while 5-star reviews use emphatics and emphasize universality (*very, highly, every, always*). 1-star reviews use first person plurals (*we, us, our*) while 5 star reviews use the second person. 1-star reviews talk about people (*manager, waiter, customer*) while 5-star reviews talk about dessert and properties of expensive restaurants like courses and atmosphere. See Jurafsky et al. (2014) for more details.

## 20.4 Using Lexicons for Sentiment Recognition

In Chapter 4 we introduced the naive Bayes algorithm for sentiment analysis. The lexicons we have focused on throughout the chapter so far can be used in a number of ways to improve sentiment detection.

In the simplest case, lexicons can be used when we don't have sufficient training data to build a supervised sentiment analyzer; it can often be expensive to have a human assign sentiment to each document to train the supervised classifier.

In such situations, lexicons can be used in a simple rule-based algorithm for classification. The simplest version is just to use the ratio of positive to negative words: if a document has more positive than negative words (using the lexicon to decide the polarity of each word in the document), it is classified as positive. Often a threshold $\lambda$ is used, in which a document is classified as positive only if the ratio

| Class | Words in 1-star reviews | Class | Words in 5-star reviews |
|---|---|---|---|
| **Negative** | *worst, rude, terrible, horrible, bad, awful, disgusting, bland, tasteless, gross, mediocre, overpriced, worse, poor* | **Positive** | *great, best, love(d), delicious, amazing, favorite, perfect, excellent, awesome, friendly, fantastic, fresh, wonderful, incredible, sweet, yum(my)* |
| **Negation** | *no, not* | **Emphatics/ universals** | *very, highly, perfectly, definitely, absolutely, everything, every, always* |
| **1Pl pro** | *we, us, our* | **2 pro** | *you* |
| **3 pro** | *she, he, her, him* | **Articles** | *a, the* |
| **Past verb** | *was, were, asked, told, said, did, charged, waited, left, took* | **Advice** | *try, recommend* |
| **Sequencers** | *after, then* | **Conjunct** | *also, as, well, with, and* |
| **Nouns** | *manager, waitress, waiter, customer, customers, attitude, waste, poisoning, money, bill, minutes* | **Nouns** | *atmosphere, dessert, chocolate, wine, course, menu* |
| **Irrealis modals** | *would, should* | **Auxiliaries** | *is/'s, can, 've, are* |
| **Comp** | *to, that* | **Prep, other** | *in, of, die, city, mouth* |

**Figure 20.10** The top 50 words associated with one–star and five-star restaurant reviews in a Yelp dataset of 900,000 reviews, using the Monroe et al. (2008) method (Jurafsky et al., 2014).

is greater than $\lambda$. If the sentiment lexicon includes positive and negative weights for each word, $\theta_w^+$ and $\theta_w^-$, these can be used as well. Here's a simple such sentiment algorithm:

$$
\begin{aligned}
f^+ &= \sum_{w \text{ s.t. } w \in positivelexicon} \theta_w^+ count(w) \\
f^- &= \sum_{w \text{ s.t. } w \in negativelexicon} \theta_w^- count(w) \\
sentiment &= \begin{cases} + & \text{if } \frac{f^+}{f^-} > \lambda \\ - & \text{if } \frac{f^-}{f^+} > \lambda \\ 0 & \text{otherwise.} \end{cases}
\end{aligned} \tag{20.13}
$$

If supervised training data is available, these counts computed from sentiment lexicons, sometimes weighted or normalized in various ways, can also be used as features in a classifier along with other lexical or non-lexical features. We return to such algorithms in Section 20.7.

## 20.5 Emotion and other classes

**emotion**

One of the most important affective classes is **emotion**, which Scherer (2000) defines as a "relatively brief episode of response to the evaluation of an external or internal event as being of major significance".

Detecting emotion has the potential to improve a number of language processing tasks. Automatically detecting emotions in reviews or customer responses (anger, dissatisfaction, trust) could help businesses recognize specific problem areas or ones that are going well. Emotion recognition could help dialog systems like tutoring systems detect that a student was unhappy, bored, hesitant, confident, and so on.

Emotion can play a role in medical informatics tasks like detecting depression or suicidal intent. Detecting emotions expressed toward characters in novels might play a role in understanding how different social groups were viewed by society at different times.

**basic emotions**

There are two widely-held families of theories of emotion. In one family, emotions are viewed as fixed atomic units, limited in number, and from which others are generated, often called **basic emotions** (Tomkins 1962, Plutchik 1962). Perhaps most well-known of this family of theories are the 6 emotions proposed by (Ekman, 1999) as a set of emotions that is likely to be universally present in all cultures: *surprise, happiness, anger, fear, disgust, sadness*. Another atomic theory is the (Plutchik, 1980) wheel of emotion, consisting of 8 basic emotions in four opposing pairs: *joy–sadness*, *anger–fear*, *trust–disgust*, and *anticipation–surprise*, together with the emotions derived from them, shown in Fig. 20.11.

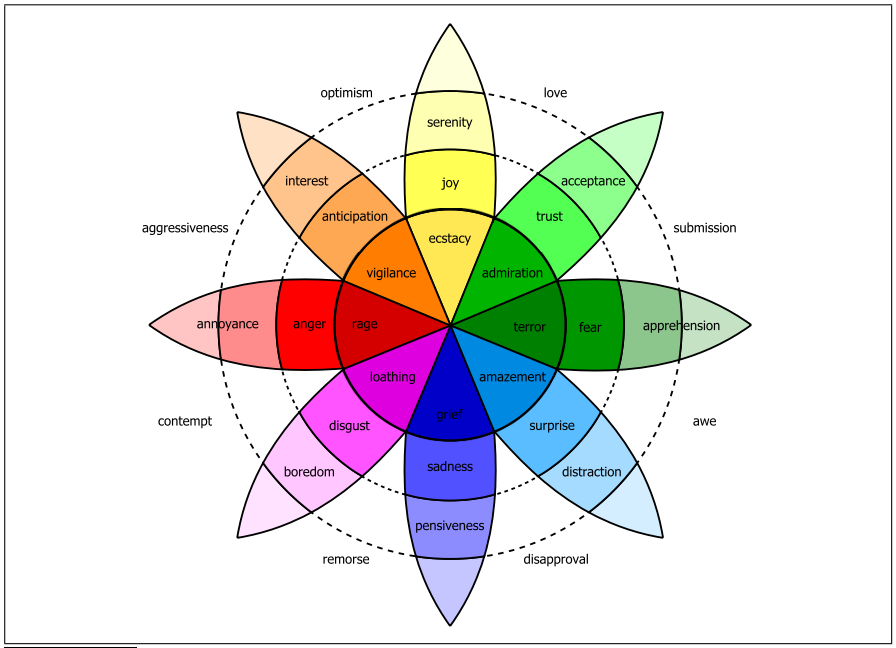


**Figure 20.11** Plutchik wheel of emotion.

The second class of emotion theories views emotion as a space in 2 or 3 dimensions (Russell, 1980). Most models include the two dimensions **valence** and **arousal**, and many add a third, **dominance**. These can be defined as:

**valence:** the pleasantness of the stimulus
**arousal:** the intensity of emotion provoked by the stimulus
**dominance:** the degree of control exerted by the stimulus

Practical lexicons have been built for both kinds of theories of emotion.

### 20.5.1 Lexicons for emotion and other affective states

**crowdsourcing**

While semi-supervised algorithms are the norm in sentiment and polarity, the most common way to build emotional lexicons is to have humans label the words. This is most commonly done using **crowdsourcing**: breaking the task into small pieces and distributing them to a large number of annotators. Let's take a look at one

crowdsourced emotion lexicon from each of the two common theoretical models of emotion.

**EmoLex**

The NRC Word-Emotion Association Lexicon, also called **EmoLex** (Mohammad and Turney, 2013), uses the Plutchik (1980) 8 basic emotions defined above. The lexicon includes around 14,000 words chosen partly from the prior lexicons (the General Inquirer and WordNet Affect Lexicons) and partly from the Macquarie Thesaurus, from which the 200 most frequent words were chosen from four parts of speech: nouns, verbs, adverbs, and adjectives (using frequencies from the Google n-gram count).

In order to ensure that the annotators were judging the correct sense of the word, they first answered a multiple-choice synonym question that primed the correct sense of the word (without requiring the annotator to read a potentially confusing sense definition). These were created automatically using the headwords associated with the thesaurus category of the sense in question in the Macquarie dictionary and the headwords of 3 random distractor categories. An example:

`Which word is closest in meaning (most related) to startle?`

- `automobile`
- `shake`
- `honesty`
- `entertain`

For each word (e.g. *startle*), the annotator was asked to rate how associated that word is with each of the 8 emotions (*joy*, *fear*, *anger*, etc.). The associations were rated on a scale of *not*, *weakly*, *moderately*, and *strongly* associated. Outlier ratings were removed, and then each term was assigned the class chosen by the majority of the annotators, with ties broken by choosing the stronger intensity, and then the 4 levels were mapped into a binary label for each word (no and weak mapped to 0, moderate and strong mapped to 1). Values from the lexicon for some sample words:

| Word | anger | anticipation | disgust | fear | joy | sadness | surprise | trust | positive | negative |
|---|---|---|---|---|---|---|---|---|---|---|
| reward | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| worry | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 |
| tenderness | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| sweetheart | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 |
| suddenly | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| thirst | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| garbage | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

A second lexicon, also built using crowdsourcing, assigns values on three dimensions (valence/arousal/dominance) to 14,000 words (Warriner et al., 2013).

The annotators marked each word with a value from 1-9 on each of the dimensions, with the scale defined for them as follows:

- valence (the pleasantness of the stimulus)
  9: happy, pleased, satisfied, contented, hopeful
  1: unhappy, annoyed, unsatisfied, melancholic, despaired, or bored
- arousal (the intensity of emotion provoked by the stimulus)
  9: stimulated, excited, frenzied, jittery, wide-awake, or aroused
  1: relaxed, calm, sluggish, dull, sleepy, or unaroused;

- dominance (the degree of control exerted by the stimulus)
  9: in control, influential, important, dominant, autonomous, or controlling
  1: controlled, influenced, cared-for, awed, submissive, or guided

Some examples are shown in Fig. 20.12

| Valence | | Arousal | | Dominance | |
|---|---|---|---|---|---|
| vacation | 8.53 | rampage | 7.56 | self | 7.74 |
| happy | 8.47 | tornado | 7.45 | incredible | 7.74 |
| whistle | 5.7 | zucchini | 4.18 | skillet | 5.33 |
| conscious | 5.53 | dressy | 4.15 | concur | 5.29 |
| torture | 1.4 | dull | 1.67 | earthquake | 2.14 |

**Figure 20.12** Samples of the values of selected words on the three emotional dimensions from Warriner et al. (2013).

There are various other hand-built lexicons of words related in various ways to the emotions. The General Inquirer includes lexicons like strong vs. weak, active vs. passive, overstated vs. understated, as well as lexicons for categories like pleasure, pain, virtue, vice, motivation, and cognitive orientation.

concrete
abstract

Another useful feature for various tasks is the distinction between **concrete** words like *banana* or *bathrobe* and **abstract** words like *belief* and *although*. The lexicon in (Brysbaert et al., 2014) used crowdsourcing to assign a rating from 1 to 5 of the concreteness of 40,000 words, thus assigning *banana*, *bathrobe*, and *bagel* 5, *belief* 1.19, *although* 1.07, and in between words like *brisk* a 2.5.

LIWC

**LIWC**, **Linguistic Inquiry and Word Count**, is another set of 73 lexicons containing over 2300 words (Pennebaker et al., 2007), designed to capture aspects of lexical meaning relevant for social psychological tasks. In addition to sentiment-related lexicons like ones for negative emotion (*bad, weird, hate, problem, tough*) and positive emotion (*love, nice, sweet*), LIWC includes lexicons for categories like anger, sadness, cognitive mechanisms, perception, tentative, and inhibition, shown in Fig. 20.13.

| Positive Emotion | Negative Emotion | Insight | Inhibition | Family | Negate |
|---|---|---|---|---|---|
| appreciat* | anger* | aware* | avoid* | brother* | aren't |
| comfort* | bore* | believe | careful* | cousin* | cannot |
| great | cry | decid* | hesitat* | daughter* | didn't |
| happy | despair* | feel | limit* | family | neither |
| interest | fail* | figur* | oppos* | father* | never |
| joy* | fear | know | prevent* | grandf* | no |
| perfect* | griev* | knew | reluctan* | grandm* | nobod* |
| please* | hate* | means | safe* | husband | none |
| safe* | panic* | notice* | stop | mom | nor |
| terrific | suffers | recogni* | stubborn* | mother | nothing |
| value | terrify | sense | wait | niece* | nowhere |
| wow* | violent* | think | wary | wife | without |

**Figure 20.13** Samples from 5 of the 73 lexical categories in LIWC (Pennebaker et al., 2007). The * means the previous letters are a word prefix and all words with that prefix are included in the category.

## 20.6 Other tasks: Personality

**personality**

Many other kinds of affective meaning can be extracted from text and speech. For example detecting a person's **personality** from their language can be useful for dialog systems (users tend to prefer agents that match their personality), and can play a useful role in computational social science questions like understanding how personality is related to other kinds of behavior.

Many theories of human personality are based around a small number of dimensions, such as various versions of the "Big Five" dimensions (Digman, 1990):

**Extroversion vs. Introversion:** sociable, assertive, playful vs. aloof, reserved, shy

**Emotional stability vs. Neuroticism:** calm, unemotional vs. insecure, anxious

**Agreeableness vs. Disagreeableness:** friendly, cooperative vs. antagonistic, fault-finding

**Conscientiousness vs. Unconscientiousness:** self-disciplined, organized vs. inefficient, careless

**Openness to experience:** intellectual, insightful vs. shallow, unimaginative

A few corpora of text and speech have been labeled for the personality of their author by having the authors take a standard personality test. The essay corpus of Pennebaker and King (1999) consists of 2,479 essays (1.9 million words) from psychology students who were asked to "write whatever comes into your mind" for 20 minutes. The EAR (Electronically Activated Recorder) corpus of Mehl et al. (2006) was created by having volunteers wear a recorder throughout the day, which randomly recorded short snippets of conversation throughout the day, which were then transcribed. The Facebook corpus of (Schwartz et al., 2013) includes 309 million words of Facebook posts from 75,000 volunteers.

For example, here are samples from Pennebaker and King (1999) from an essay written by someone on the neurotic end of the neurotic/emotionally stable scale,

> One of my friends just barged in, and I jumped in my seat. This is crazy. I should tell him not to do that again. I'm not that fastidious actually. But certain things annoy me. The things that would annoy me would actually annoy any normal human being, so I know I'm not a freak.

and someone on the emotionally stable end of the scale:

> I should excel in this sport because I know how to push my body harder than anyone I know, no matter what the test I always push my body harder than everyone else. I want to be the best no matter what the sport or event. I should also be good at this because I love to ride my bike.

**interpersonal stance**

Another kind of affective meaning is what Scherer (2000) calls **interpersonal stance**, the 'affective stance taken toward another person in a specific interaction coloring the interpersonal exchange'. Extracting this kind of meaning means automatically labeling participants for whether they are friendly, supportive, distant. For example Ranganath et al. (2013) studied a corpus of speed-dates, in which participants went on a series of 4-minute romantic dates, wearing microphones. Each participant labeled each other for how flirtatious, friendly, awkward, or assertive they were. Ranganath et al. (2013) then used a combination of lexicons and other features to detect these interpersonal stances from text.

## 20.7 Affect Recognition

Detection of emotion, personality, interactional stance, and the other kinds of affective meaning described by Scherer (2000) can be done by generalizing the algorithms described above for detecting sentiment.

The most common algorithms involve supervised classification: a training set is labeled for the affective meaning to be detected, and a classifier is built using features extracted from the training set. As with sentiment analysis, if the training set is large enough, and the test set is sufficiently similar to the training set, simply using all the words or all the bigrams as features in a powerful classifier like SVM or logistic regression, as described in Fig. **??** in Chapter 4, is an excellent algorithm whose performance is hard to beat. Thus we can treat affective meaning classification of a text sample as simple document classification.

Some modifications are nonetheless often necessary for very large datasets. For example, the Schwartz et al. (2013) study of personality, gender, and age using 700 million words of Facebook posts used only a subset of the n-grams of lengths 1-3. Only words and phrases used by at least 1% of the subjects were included as features, and 2-grams and 3-grams were only kept if they had sufficiently high PMI (PMI greater than $2 * \mathit{length}$, where *length* is the number of words):

$$\text{pmi}(\mathit{phrase}) = \log \frac{p(\mathit{phrase})}{\prod_{w \in \mathit{phrase}} p(w)} \tag{20.14}$$

Various weights can be used for the features, including the raw count in the training set, or some normalized probability or log probability. Schwartz et al. (2013), for example, turn feature counts into phrase likelihoods by normalizing them by each subject's total word use.

$$p(\mathit{phrase}|\mathit{subject}) = \frac{\text{freq}(\mathit{phrase}, \mathit{subject})}{\sum_{\mathit{phrase}' \in \text{vocab}(\mathit{subject})} \text{freq}(\mathit{phrase}', \mathit{subject})} \tag{20.15}$$

If the training data is sparser, or not as similar to the test set, any of the lexicons we've discussed can play a helpful role, either alone or in combination with all the words and n-grams.

Many possible values can be used for lexicon features. The simplest is just an indicator function, in which the value of a feature $f_{\mathscr{L}}$ takes the value 1 if a particular text has any word from the relevant lexicon $\mathscr{L}$. Using the notation of Chapter 4, in which a feature value is defined for a particular output class $c$ and document $x$.

$$f_{\mathscr{L}}(c,x) = \begin{cases} 1 & \text{if } \exists w : w \in \mathscr{L} \ \& \ w \in x \ \& \ \mathit{class} = c \\ 0 & \text{otherwise} \end{cases} \tag{20.16}$$

Alternatively the value of a feature $f_{\mathscr{L}}$ for a particular lexicon $\mathscr{L}$ can be the total number of word *tokens* in the document that occur in $\mathscr{L}$:

$$f_{\mathscr{L}} = \sum_{w \in \mathscr{L}} \mathit{count}(w)$$

For lexica in which each word is associated with a score or weight, the count can be multiplied by a weight $\theta_w^{\mathscr{L}}$:

$$f_{\mathscr{L}} = \sum_{w \in \mathscr{L}} \theta_w^{\mathscr{L}} count(w)$$

Counts can alternatively be logged or normalized per writer as in Eq. 20.15.

However they are defined, these lexicon features are then used in a supervised classifier to predict the desired affective category for the text or document. Once a classifier is trained, we can examine which lexicon features are associated with which classes. For a classifier like logistic regression the feature weight gives an indication of how associated the feature is with the class.

Thus, for example, (Mairesse and Walker, 2008) found that for classifying personality, for the dimension *Agreeable*, the LIWC lexicons *Family* and *Home* were positively associated while the LIWC lexicons *anger* and *swear* were negatively associated. By contrast, Extroversion was positively associated with the *Friend*, *Religion* and *Self* lexicons, and Emotional Stability was positively associated with *Sports* and negatively associated with *Negative Emotion*.

In the situation in which we use all the words and phrases in the document as potential features, we can use the resulting weights from the learned regression classifier as the basis of an affective lexicon. Thus, for example, in the Extroversion/Introversion classifier of Schwartz et al. (2013), ordinary least-squares regression is used to predict the value of a personality dimension from all the words and phrases. The resulting regression coefficient for each word or phrase can be used as an association value with the predicted dimension. The word clouds in Fig. 20.14 show an example of words associated with introversion (a) and extroversion (b).

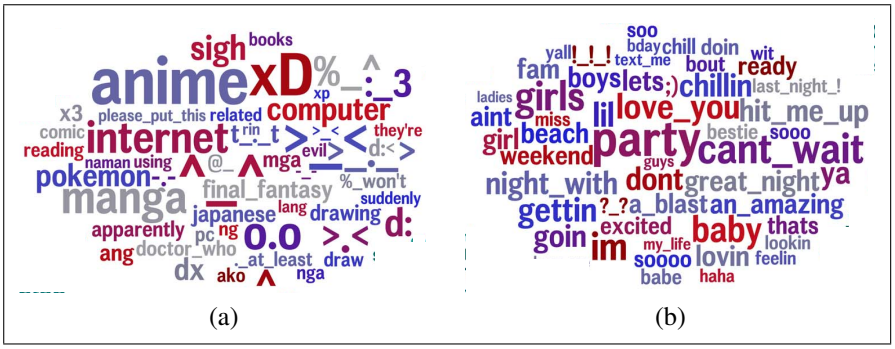


**Figure 20.14** Word clouds from Schwartz et al. (2013), showing words highly associated with introversion (left) or extroversion (right). The size of the word represents the association strength (the regression coefficient), while the color (ranging from cold to hot) represents the relative frequency of the word/phrase (from low to high).

## 20.8 Summary

- Many kinds of affective states can be distinguished, including *emotions*, *moods*, *attitudes* (which include *sentiment*), *interpersonal stance*, and *personality*.
- Words have **connotational** aspects related to these affective states, and this connotational aspect of word meaning can be represented in lexicons.

- Affective lexicons can be built by hand, using **crowd sourcing** to label the affective content of each word.
- Lexicons can be built **semi-supervised**, bootstrapping from seed words using similarity metrics like the frequency two words are conjoined by *and* or *but*, the two words' pointwise mutual information, or their association via WordNet synonymy or antonymy relations.
- Lexicons can be learned in a **fully supervised** manner, when a convenient training signal can be found in the world, such as ratings assigned by users on a review site.
- Words can be assigned weights in a lexicon by using various functions of word counts in training texts, and ratio metrics like **log odds ratio informative Dirichlet prior**.
- **Emotion** can be represented by fixed atomic units often called **basic emotions**, or as points in space defined by dimensions like **valence** and **arousal**.
- Personality is often represented as a point in 5-dimensional space.
- Affect can be detected, just like sentiment, by using standard supervised **text classification** techniques, using all the words or bigrams in a text as features. Additional features can be drawn from counts of words in lexicons.
- Lexicons can also be used to detect affect in a **rule-based classifier** by picking the simple majority sentiment based on counts of words in each lexicon.

# Bibliographical and Historical Notes

The idea of formally representing the subjective meaning of words began with Osgood et al. (1957), the same pioneering study that first proposed the vector space model of meaning described in Chapter 6. Osgood et al. (1957) had participants rate words on various scales, and ran factor analysis on the ratings. The most significant factor they uncovered was the evaluative dimension, which distinguished between pairs like *good/bad*, *valuable/worthless*, *pleasant/unpleasant*. This work influenced the development of early dictionaries of sentiment and affective meaning in the field of **content analysis** (Stone et al., 1966).

**subjectivity**

Wiebe (1994) began an influential line of work on detecting **subjectivity** in text, beginning with the task of identifying subjective sentences and the subjective characters who are described in the text as holding private states, beliefs or attitudes. Learned sentiment lexicons such as the polarity lexicons of (Hatzivassiloglou and McKeown, 1997) were shown to be a useful feature in subjectivity detection (Hatzivassiloglou and Wiebe 2000, Wiebe 2000).

The term **sentiment** seems to have been introduced in 2001 by Das and Chen (2001), to describe the task of measuring market sentiment by looking at the words in stock trading message boards. In the same paper Das and Chen (2001) also proposed the use of a sentiment lexicon. The list of words in the lexicon was created by hand, but each word was assigned weights according to how much it discriminated a particular class (say buy versus sell) by maximizing across-class variation and minimizing within-class variation. The term *sentiment*, and the use of lexicons, caught on quite quickly (e.g., inter alia, Turney 2002). Pang et al. (2002) first showed the power of using all the words without a sentiment lexicon; see also Wang and Manning (2012).

The semi-supervised methods we describe for extending sentiment dictionaries all drew on the early idea that synonyms and antonyms tend to co-occur in the same sentence. (Miller and Charles 1991, Justeson and Katz 1991). Other semi-supervized methods for learning cues to affective meaning rely on information extraction techniques, like the AutoSlog pattern extractors (Riloff and Wiebe, 2003).

For further information on sentiment analysis, including discussion of lexicons, see the useful surveys of Pang and Lee (2008) and Liu (2015).

Baccianella, S., Esuli, A., and Sebastiani, F. (2010). Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining.. In *LREC-10*, pp. 2200–2204.

Brysbaert, M., Warriner, A. B., and Kuperman, V. (2014). Concreteness ratings for 40 thousand generally known english word lemmas. *Behavior Research Methods*, *46*(3), 904–911.

Das, S. R. and Chen, M. Y. (2001). Yahoo! for Amazon: Sentiment parsing from small talk on the web. EFA 2001 Barcelona Meetings. Available at SSRN: http://ssrn.com/abstract=276189.

Digman, J. M. (1990). Personality structure: Emergence of the five-factor model. *Annual Review of Psychology*, *41*(1), 417–440.

Ekman, P. (1999). Basic emotions. In Dalgleish, T. and Power, M. J. (Eds.), *Handbook of Cognition and Emotion*, pp. 45–60. Wiley.

Fano, R. M. (1961). *Transmission of Information: A Statistical Theory of Communications*. MIT Press.

Hatzivassiloglou, V. and McKeown, K. R. (1997). Predicting the semantic orientation of adjectives. In *ACL/EACL-97*, pp. 174–181.

Hatzivassiloglou, V. and Wiebe, J. (2000). Effects of adjective orientation and gradability on sentence subjectivity. In *COLING-00*, pp. 299–305.

Hu, M. and Liu, B. (2004). Mining and summarizing customer reviews. In *SIGKDD-04*.

Jurafsky, D., Chahuneau, V., Routledge, B. R., and Smith, N. A. (2014). Narrative framing of consumer sentiment in online restaurant reviews. *First Monday*, *19*(4).

Justeson, J. S. and Katz, S. M. (1991). Co-occurrences of antonymous adjectives and their contexts. *Computational linguistics*, *17*(1), 1–19.

Kim, S. M. and Hovy, E. H. (2004). Determining the sentiment of opinions. In *COLING-04*.

Liu, B. (2015). *Sentiment Analysis: Mining Opinions, Sentiments, and Emotions*. Cambridge University Press.

Mairesse, F. and Walker, M. A. (2008). Trainable generation of big-five personality styles through data-driven parameter estimation. In *ACL-08*, Columbus.

Mehl, M. R., Gosling, S. D., and Pennebaker, J. W. (2006). Personality in its natural habitat: manifestations and implicit folk theories of personality in daily life.. *Journal of Personality and Social Psychology*, *90*(5).

Miller, G. A. and Charles, W. G. (1991). Contextual correlates of semantics similarity. *Language and Cognitive Processes*, *6*(1), 1–28.

Mohammad, S. M. and Turney, P. D. (2013). Crowdsourcing a word-emotion association lexicon. *Computational Intelligence*, *29*(3), 436–465.

Monroe, B. L., Colaresi, M. P., and Quinn, K. M. (2008). Fightin'words: Lexical feature selection and evaluation for identifying the content of political conflict. *Political Analysis*, *16*(4), 372–403.

Osgood, C. E., Suci, G. J., and Tannenbaum, P. H. (1957). *The Measurement of Meaning*. University of Illinois Press.

Pang, B. and Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and trends in information retrieval*, *2*(1-2), 1–135.

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? Sentiment classification using machine learning techniques. In *EMNLP 2002*, pp. 79–86.

Pennebaker, J. W., Booth, R. J., and Francis, M. E. (2007). *Linguistic Inquiry and Word Count: LIWC 2007*. Austin, TX.

Pennebaker, J. W. and King, L. A. (1999). Linguistic styles: language use as an individual difference. *Journal of Personality and Social Psychology*, *77*(6).

Picard, R. W. (1995). Affective computing. Tech. rep. 321, MIT Media Lab Perceputal Computing Technical Report. Revised November 26, 1995.

Plutchik, R. (1962). *The emotions: Facts, theories, and a new model*. Random House.

Plutchik, R. (1980). A general psychoevolutionary theory of emotion. In Plutchik, R. and Kellerman, H. (Eds.), *Emotion: Theory, Research, and Experience, Volume 1*, pp. 3–33. Academic Press.

Potts, C. (2011). On the negativity of negation. In Li, N. and Lutz, D. (Eds.), *Proceedings of Semantics and Linguistic Theory 20*, pp. 636–659. CLC Publications, Ithaca, NY.

Ranganath, R., Jurafsky, D., and McFarland, D. A. (2013). Detecting friendly, flirtatious, awkward, and assertive speech in speed-dates. *Computer Speech and Language*, *27*(1), 89–115.

Riloff, E. and Wiebe, J. (2003). Learning extraction patterns for subjective expressions. In *EMNLP 2003*, Sapporo, Japan.

Russell, J. A. (1980). A circumplex model of affect. *Journal of personality and social psychology*, *39*(6), 1161–1178.

Scherer, K. R. (2000). Psychological models of emotion. In Borod, J. C. (Ed.), *The neuropsychology of emotion*, pp. 137–162. Oxford.

Schwartz, H. A., Eichstaedt, J. C., Kern, M. L., Dziurzynski, L., Ramones, S. M., Agrawal, M., Shah, A., Kosinski, M., Stillwell, D., Seligman, M. E. P., and Ungar, L. H. (2013). Personality, gender, and age in the language of social media: The open-vocabulary approach. *PloS one*, *8*(9), e73791.

Stone, P., Dunphry, D., Smith, M., and Ogilvie, D. (1966). *The General Inquirer: A Computer Approach to Content Analysis*. Cambridge, MA: MIT Press.

Tomkins, S. S. (1962). *Affect, imagery, consciousness: Vol. I. The positive affects*. Springer.

Turney, P. D. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *ACL-02*.

Wang, S. and Manning, C. D. (2012). Baselines and bigrams: Simple, good sentiment and topic classification. In *ACL 2012*, pp. 90–94.

Warriner, A. B., Kuperman, V., and Brysbaert, M. (2013). Norms of valence, arousal, and dominance for 13,915 English lemmas. *Behavior Research Methods*, *45*(4), 1191–1207.

Wiebe, J. (1994). Tracking point of view in narrative. *Computational Linguistics*, *20*(2), 233–287.

Wiebe, J. (2000). Learning subjective adjectives from corpora. In *AAAI-00*, Austin, TX, pp. 735–740.

Wiebe, J., Bruce, R. F., and O'Hara, T. P. (1999). Development and use of a gold-standard data set for subjectivity classifications. In *ACL-99*, pp. 246–253.

Wilson, T., Wiebe, J., and Hoffmann, P. (2005). Recognizing contextual polarity in phrase-level sentiment analysis. In *HLT-EMNLP-05*, pp. 347–354.