Extracting Appraisal Expressions

Kenneth Bloom and **Navendu Garg** and **Shlomo Argamon**
Computer Science Department
Illinois Institute of Technology
10 W. 31st St.
Chicago, IL 60616
{kbloom1,gargnav,argamon}@iit.edu

Abstract

Sentiment analysis seeks to characterize opinionated or evaluative aspects of natural language text. We suggest here that *appraisal expression extraction* should be viewed as a fundamental task in sentiment analysis. An *appraisal expression* is a textual unit expressing an evaluative stance towards some target. The task is to find and characterize the evaluative attributes of such elements. This paper describes a system for effectively extracting and disambiguating adjectival appraisal expressions in English outputting a generic representation in terms of their evaluative function in the text. Data mining on appraisal expressions gives meaningful and non-obvious insights.

1 Introduction

Sentiment analysis, which seeks to analyze opinion in natural language text, has grown in interest in recent years. Sentiment analysis includes a variety of different problems, including: sentiment classification techniques to classify reviews as positive or negative, based on bag of words (Pang et al., 2002) or positive and negative words (Turney, 2002; Mullen and Collier, 2004); classifying sentences in a document as either subjective or objective (Riloff and Wiebe, 2003; Pang and Lee, 2004); identifying or classifying appraisal targets (Nigam and Hurst, 2004); identifying the source of an opinion in a text (Choi et al., 2005), whether the author is expressing the opinion, or whether he is attributing the opinion to someone else; and developing interactive and visual opinion mining methods (Gamon et al., 2005; Popescu and Etzioni, 2005). Much of this work has utilized the fundamental concept of 'semantic orientation', (Turney, 2002); however, sentiment analysis still lacks a 'unified field theory'.

We propose in this paper that a fundamental task underlying many of these formulations is the extraction and analysis of *appraisal expressions*, defined as those structured textual units which express an evaluation of some object. An appraisal expression has three main components: an *attitude* (which takes an evaluative stance about an object), a *target* (the object of the stance), and a *source* (the person taking the stance) which may be implied.

The idea of appraisal extraction is a generalization of problem formulations developed in earlier works. Mullen and Collier's (2004) notion of classifying appraisal terms using a multidimensional set of attributes is closely tied to the definition of an appraisal expression, which is classified along several dimensions. In previous work (Whitelaw et al., 2005), we presented a related technique of finding opinion phrases, using a multidimensional set of attributes and modeling the semantics of modifiers in these phrases. The use of multiple text classifiers by Wiebe and colleagues (Wilson et al., 2005; Wiebe et al., 2004) for various kinds of sentiment classification can also be viewed as a sentence-level technique for analyzing appraisal expressions. Nigam and Hurst's (2004) work on detecting opinions about a certain topic presages our notion of connecting attitudes to targets, while Popescu and Etzioni's (2005) opinion mining technique also fits well into our framework.

In this paper we describe a system for extracting adjectival appraisal expressions, based on a hand-built lexicon, a combination of heuristic shallow parsing and dependency parsing, and expectation-maximization word sense disambiguation. Each ex-

Proceedings of NAACL HLT 2007, pages 308–315,
Rochester, NY, April 2007. ©2007 Association for Computational Linguistics

tracted appraisal expression is represented as a set of feature values in terms of its evaluative function in the text. We have applied this system to two domains of texts: product reviews, and movie reviews. Manual evaluation of the extraction shows our system to work well, as well as giving some directions for improvement. We also show how straightforward data mining can give users very useful information about public opinion.

2 Appraisal Expressions

We define an *appraisal expression* to be an elementary linguistic unit that conveys an attitude of some kind towards some target. An *appraisal expression* is defined to comprise a *source*, an *attitude*, and a *target*, each represented by various attributes. For example, in 'I found the movie quite monotonous', the speaker (the *Source*) expresses a negative *Attitude* ('quite monotonous') towards 'the movie' (the *Target*). Note that attitudes come in different types; for example, 'monotonous' describes an inherent quality of the Target, while 'loathed' would describe the emotional reaction of the Source.

Attitude may be expressed through nouns, verbs, adjectives and metaphors. Extracting all of this information accurately for all of these types of appraisal expressions is a very difficult problem. We therefore restrict ourselves for now to *adjectival appraisal expressions* that are each contained in a single sentence. Additionally, we focus here only on extracting and analyzing the attitude and the target, but not the source. Even with these restrictions, we obtain interesting results (Sec. 7).

2.1 Appraisal attributes

Our method is grounded in Appraisal Theory, developed by Martin and White (2005), which analyzes the way opinion is expressed. Following Martin and White, we define:

Attitude type is type of appraisal being expressed—one of *affect*, *appreciation*, or *judgment* (Figure 1). *Affect* refers to an emotional state (e.g., 'happy', 'angry'), and is the most explicitly subjective type of appraisal. The other two types express evaluation of external entities, differentiating between intrinsic *appreciation* of object properties (e.g., 'slender', 'ugly') and social *judgment* (e.g., 'heroic', 'idiotic').

Attitude Type
- Appreciation
  - Composition
    - Balance: *consistent, discordant, ...*
    - Complexity: *elaborate, convoluted, ...*
  - Reaction
    - Impact: *amazing, compelling, dull, ...*
    - Quality: *beautiful, elegant, hideous, ...*
  - Valuation: *innovative, profound, inferior, ...*
- Affect: *happy, joyful, furious, ...*
- Judgment
  - Social Esteem
    - Capacity: *clever, competent, immature, ...*
    - Tenacity: *brave, hard-working, foolhardy, ...*
    - Normality: *famous, lucky, obscure, ...*
  - Social Sanction
    - Propriety: *generous, virtuous, corrupt, ...*
    - Veracity: *honest, sincere, sneaky, ...*

Figure 1: The Attitude Type taxonomy, with examples of adjectives from the lexicon.

Orientation is whether the attitude is *positive* ('good') or *negative* ('bad').

Force describes the intensity of the appraisal. Force is largely expressed via modifiers such as 'very' (increased force), or 'slightly' (decreased force), but may also be expressed lexically, for example 'greatest' vs. 'great' vs. 'good'.

Polarity of an appraisal is *marked* if it is scoped in a polarity marker (such as 'not'), or *unmarked* otherwise. Other attributes of appraisal are affected by negation; e.g., 'not good' also has the opposite orientation from 'good'.

Target type is a domain-dependent semantic type for the target. This attribute takes on values from a domain-dependent taxonomy, representing important (and easily extractable) distinctions between targets in the domain.

2.2 Target taxonomies

Two domain-dependent target type taxonomies are shown in Figure 2. In both, the primary distinction is between a direct naming of a kind of "Thing" or a deictic/pronominal reference (e.g., "those" or "it"), since the system does not currently rely on coreference resolution. References are further divided into references to the writer/reader ('interactants') and to other people or objects.

The Thing subtrees for the two domains differ somewhat. In the movie domain, Things such as 'this movie', 'Nicholas Cage', or 'cinematography', are classified into six main categories: movies (the one being reviewed, or another one), people

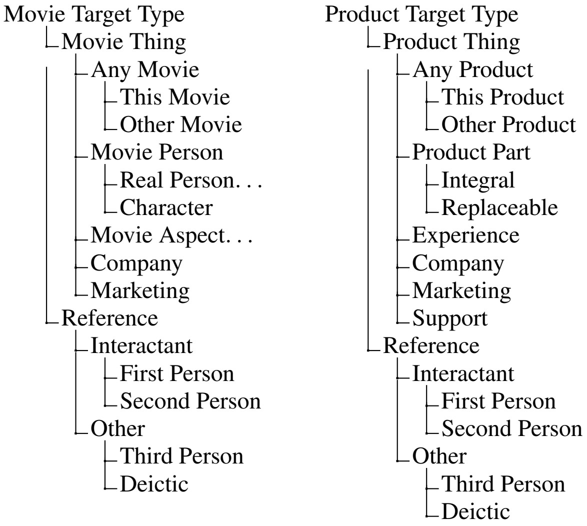

Figure 2: Target taxonomies for movie and product reviews.

(whether characters, or real people involved in making the film), aspects of the movie itself (its plot, special effects, etc.), the companies involved in making it, or aspects of marketing the movie (such as trailers). For target Things in product reviews, we replace 'Movie Person' and 'Movie Aspect' by 'Product Part' with two subcategories: 'Integral', for parts of the product itself (e.g., wheels or lenses), and 'Replaceable', for parts or supplies meant to be periodically replaced (e.g., batteries or ink cartridges). The categories of 'Support', for references to aspects of customer support, and 'Experience' for things associated with the experience of using the product (such as 'pictures' or 'resolution', were also added.

3 Appraisal Extraction

In our system, appraisal extraction runs in several independent stages. First, the appraisal extractor finds appraisal expressions by finding the chunks of text that express attitudes and targets. Then, it links each attitude group found to a target in the text. Finally, it uses a probabilistic model to determine which attitude type should be assigned when attitude chunks were ambiguous.

3.1 Chunking

The chunker is based on our earlier work (Whitelaw et al., 2005), which finds attitude groups and targets using a hand-built lexicon (Sec. 4). This lexicon contains head adjectives (which specify values for the attributes attitude type, force, polarity, and orientation), and appraisal modifiers (which specify transformations to the four attributes). Some head adjectives are ambiguous, having multiple entries in the lexicon with different attribute values. In all cases, different entries for a given word have different attitude types. If the head adjective is ambiguous, multiple groups are created, to be disambiguated later. See our previous work (Whitelaw et al., 2005) for a discussion of the technique.

Target groups are found by matching phrases in the lexicon with corresponding phrases in the text and assigning the target type listed in the lexicon.

3.2 Linking

After finding attitude groups and candidate targets, the system links each attitude to a target. Each sentence is parsed to a dependency representation, and a ranked list of linkage specifications is used to look for paths in the dependency tree connecting some word in the source to some word in the target. Such linkage specifications are hand-constructed, and manually assigned priorities, so that when two linkage specifications match, only the highest priority specification is used. For example, the two highest priority linkage specifications are:

1. target $\xrightarrow{nsubj} x \xleftarrow{dobj} y \xleftarrow{amod}$ attitude
2. attitude $\xrightarrow{amod}$ target

The first specification selects the subject of a sentence where the appraisal modifies a noun in the predicate, for example 'The Matrix' in 'The Matrix is a good movie'. The second selects the noun modified by an adjective group, for example 'movie' in 'The Matrix is a good movie'.

If no linkage is found connecting an attitude to a candidate target, the system goes through the linkage specifications again, trying to find any word in the sentence connected to the appraisal group by a known linkage. The selected word is assigned the generic category of *movie thing* or *product thing* (depending on the domain of the text). If no linkage is found at all, the system assigns the default category *movie thing* or *product thing*, assuming that there is an appraised thing that couldn't be found using the given linkage specifications.

3.3 Disambiguation

After linkages are made, this information is used to disambiguate multiple senses that may be present in a given appraisal expression. Most cases are unambiguous, but in some cases two, or occasionally even three, senses are possible. We bootstrap from the unambiguous cases, using a probabilistic model, to resolve the ambiguities. The attitude type places some grammatical/semantic constraints on the clause. Two key constraints are the syntactic relation with the target (which can differentiate *affect* from the other types of appraisal), and whether the target type has consciousness (which helps differentiate *judgment* and *affect* from *appreciation*). To capture these constraints, we model the probability of a given attitude type being correct, given the target type and the linkage specification used to connect the attitude to the target, as follows.

The correct attitude type of an appraisal expression is modeled by a random variable A, the set of all attitude types in the system is denoted by $\mathcal{A}$, and a specific attitude type is denoted by a. As described above, other attributes besides attitude type may also vary between word senses, but attitude type always changes between word senses, so when the system assigns a probability to an attitude type, it is assigning that probability to the whole word sense.

We denote the linkage type used in a given appraisal expression by L, the set of all possible linkages as $\mathcal{L}$, and a specific linkage type by l. Note that the first attempt with a linkage specification (to find a chunked target) is considered to be different from the second attempt with the same linkage specification (which attempts to find any word). Failure to find an applicable linkage rule is considered as yet another 'linkage' for the probability model. Since our system uses 29 different linkage specifications, there are a total of 59 different possible linkages types.

The target type of a given appraisal expression is denoted by T, the set of all target types by $\mathcal{T}$, and a specific target type by t. We consider an expression to have a given target type $T = t$ only if that is its specific target type; if its target type is a descendant of t, then its target type is not t in the model. $\mathcal{E}$ denotes the set of all extracted appraisal expressions. The term exp denotes a specific expression.

Our goal is to estimate, for each appraisal expression exp in the corpus, the probability of its attitude type being a, given the expression's target type t and linkage type l

$$P(A = a|exp) = P(A = a|T = t, L = l)$$

To do this, we define a model M of this probability, and then estimate the maximum likelihood model using Expectation-Maximization.

We model $P_M(A = a|T = t, L = l)$ by first applying Bayes' theorem:

$$P_M(A = a|T = t, L = l) =$$

$$\frac{P_M(T = t, L = l|A = a)P_M(A = a)}{P_M(T = t, L = l)}$$

Assuming conditional independence of target type and linkage, this becomes:

$$\frac{P_M(T = t|A = a)P_M(L = l|A = a)P_M(A = a)}{P_M(T = t)P_M(L = l)}$$

M's parameters thus represent the conditional and marginal probabilities on this right-hand-side.

Given a set of (possibly ambiguous) appraisal expressions $\mathcal{E}$ identified by chunking and linkage detection, we seek the maximum likelihood model

$$M^* = \arg\max_M \prod_{exp \in \mathcal{E}} \prod_{a \in \mathcal{A}} M(A = a|exp)$$

M^* will be our best estimate of P, given the processed data in a given corpus. The system estimates M^* using an implementation of Expectation-Maximization over the entire corpus. The highest-probability attitude type (hence sense) according to M is then chosen for each appraisal expression.

4 The Lexicon

As noted above, attitude groups were identified via a domain-independent lexicon of appraisal adjectives, adverbs, and adverb modifiers. [1] For the movie domain, appraised things were identified based on a manually constructed lexicon containing generic movie words, as well as automatically constructed lexicons of proper names specific to each movie being reviewed. For each product type considered, we manually constructed a lexicon containing generic product words; we did not find it necessary to construct product-specific lexicons.

[1] All of the lexicons used in the paper can be found at http://lingcog.iit.edu/arc/appraisal_lexicon_2007a.tar.gz

For adjectival attitudes, we used the lexicon developed we developed in our previous work (Whitelaw et al., 2005) on appraisal. We reviewed the entire lexicon to determine its accuracy and made numerous improvements.

Generic target lexicons were constructed by starting with a small sample of the kind of reviews that the lexicon would apply to. We examined these manually to find generic words referring to appraised things to serve as seed terms for the lexicon and used WordNet (Miller, 1995) to suggest additional terms to add to the lexicon.

Since movie reviews often refer to the specific contents of the movie under review by proper names (of actors, the director, etc.), we also automatically constructed a *specific target lexicon* for each movie in the corpus, based on lists of actors, characters, writers, directors, and companies listed for the film at `imdb.com`. Each such specific lexicon was only used for processing reviews of the movie it was generated for, so the system had no specific knowledge of terms related to other movies during processing.

5 Corpora

We evaluated our appraisal extraction system on two corpora. The first is the standard publicly available collection of movie reviews constructed by Pang and Lee (2004). This standard testbed consists of 1000 positive and 1000 negative reviews, taken from the IMDb movie review archives[2]. Reviews with 'neutral' scores (such as three stars out of five) were removed by Pang and Lee, giving a data set with only clearly positive and negative reviews. The average document length in this corpus is 764 words, and 1107 different movies are reviewed.

The second corpus is a collection of user product reviews taken from `epinions.com` supplied in 2004 for research purposes by Amir Ashkenazi of Shopping.Com. The base collection contains reviews for three types of products: baby strollers, digital cameras, and printers. Each review has a numerical rating (1–5); based on this, we labeled positive and negative reviews in the same way as Pang and Lee did for the movie reviews corpus. The products corpus has 15162 documents, averaging 442 words long. This comprises 11769 positive documents, 1420 neutral documents, and 1973 negative documents. There are 905 reviews of strollers, 5778 reviews of ink-jet printers and 8479 reviews of digital cameras, covering 516 individual products.

Each document in each corpus was preprocessed into individual sentences, lower-cased, and tokenized. We used an implementation of Brill's (1992) part-of-speech tagger to find adjectives and modifiers; for parsing, we used the Stanford dependency parser (Klein and Manning, 2003).

6 Evaluating Extraction

We performed two manual evaluations on the system. The first was to evaluate the overall accuracy of the entire system. The second was to specifically evaluate the accuracy of the probabilistic disambiguator.

6.1 Evaluating Accuracy

We evaluated randomly selected appraisal expressions for extraction accuracy on a number of binary measures. This manual evaluation was performed by the first author.We evaluated interrater reliability between this rater and another author on 200 randomly selected appraisal expressions (100 on each corpus). The first rater rated an additional 120 expressions (60 for each corpus), and combined these with his ratings for interrater reliability to compute system accuracy, for a total of 320 expressions (160 for each corpus). The (binary) rating criteria were as follows. Relating to the appraisal group:

APP Does the expression express appraisal at all?

ARM If so, does the appraisal group have all relevant modifiers?

HEM Does the appraisal group include extra modifiers? (Results are shown negated, so that higher numbers are better.)

Relating to the target:

HT If there is appraisal, is there an identifiable target (even if the system missed it)?

FT If there is appraisal, did the system identify some target? (Determined automatically.)

RT If so, is it the correct one?

Relating to the expression's attribute values (if it expresses appraisal):

Att Is the attitude type assigned correct?

Ori Is the orientation assigned correct?

Pol Is the polarity assigned correct?

Tar Is the target type assigned correct?

Pre Is the target type the most precise value in the taxonomy for this target?

[2] See `http://www.cs.cornell.edu/people/pabo/movie-review-data/`

Table 1: System accuracy at evaluated tasks. 95% confidence one-proportion z-intervals are reported.

Measure	Movies	Products	Combined
APP	86% ± 3%	81% ± 3%	83% ± 2%
ARM	94% ± 2%	95% ± 2%	95% ± 1%
¬ HEM	99% ± 1%	100%	99.6% ± 0.4%
HT	91% ± 2%	97% ± 2%	94% ± 1%
FT	96% ± 2%	94% ± 2%	95% ± 1%
RT	77% ± 4%	73% ± 4%	75% ± 3%
Att	78% ± 4%	80% ± 4%	79% ± 2%
Ori	95% ± 2%	95% ± 2%	94% ± 1%
Pol	97% ± 1%	96% ± 2%	97% ± 1%
Tar	84% ± 3%	86% ± 3%	85% ± 2%
Pre	70% ± 4%	77% ± 4%	73% ± 3%

Table 2: Interrater reliability of manual evaluation. 95% confidence intervals are reported.

Measure	Movies	Products	Combined
APP	71% ± 9%	87% ± 7%	79% ± 6%
ARM	95% ± 5%	91% ± 6%	93% ± 4%
¬ HEM	98% ± 3%	100%	99% ± 1%
HT	97% ± 4%	99% ± 3%	98% ± 3%
FT	N/A	N/A	N/A
RT	94% ± 6%	97% ± 4%	96% ± 4%
Att	79% ± 10%	86% ± 8%	83% ± 6%
Ori	93% ± 6%	94% ± 5%	93% ± 4%
Pol	96% ± 4%	94% ± 5%	95% ± 4%
Tar	94% ± 6%	90% ± 7%	91% ± 5%
Pre	86% ± 10%	90% ± 8%	88% ± 6%

Results are given in Table 1, and interrater reliability is given in Table 2. In nearly all cases agreement percentages are above 80%, indicating good inter-rater consensus. Regarding precision, we note that most aspects of extraction seem to work quite well. The area of most concern in the system is precision of target classification. This may be improved with further development of the target lexicons to classify more terms to specific leaves in the target type hierarchy. The other area of concern is the **APP** test, which encountered difficulties when a word could be used as appraisal in some contexts, but not in others, particularly when an appraisal word appeared as a nominal classifier.

6.2 Evaluating Disambiguation

The second experiment evaluated the accuracy of EM in disambiguating the attitude type of appraisal expressions. We evaluated the same number of expressions as used for the overall accuracy experiment (100 used for interrater reliability and accuracy, plus 60 used only for accuracy on each corpus), each having two or more word senses, presenting all of the attitude types possible for each appraisal expression, as well as a 'none of the above' and a 'not appraisal' option, asking the rater to select which one applied to the selected expression in context.

Baseline disambiguator accuracy, if the computer were to simply pick randomly from the choices specified in the lexicon is 48% for both corpora. Interrater agreement was 80% for movies and 73% for products (taken over 100 expressions from each corpus.)

Considering just those appraisal expressions which the raters decided were appraisal, the disambiguator achieved 58% accuracy on appraisal expressions from the movies corpus and 56% accuracy on the products corpus. Further analysis of the results of the disambiguator shows that most of the errors occur when the target type is the generic category *thing* which occurs when the target is not in the target lexicon. Performance on words recognized as having more specific target types is better: 68% for movies, and 59% for products. This indicates that specific target type is an important indicator of attitude type.

7 Opinion Mining

We (briefly) demonstrate the usefulness of appraisal expression extraction by using it for opinion mining. In opinion mining, we find large numbers of reviews and perform data mining to determine which aspects of a product people like or dislike, and in which ways. To do this, we search for association rules describing the appraisal features that can be found in a single appraisal expression. We generally look for rules that contain attitude type, orientation, thing type, and a product name, when these rules occur more frequently than expected.

The idea is similar to Agrawal and Srikant's (1995) notion of generalized association rules. We treat each appraisal expression as a transaction, with the attributes of attitude type, orientation, polarity, force, and thing type, as well as the document attributes product name, product type, and document classification (based on the number of stars the reviewer gave the product). We use CLOSET+ (Wang et al., 2003) to find all of the frequent closed itemsets in the data, with a support greater than or equal to 20 occurrences. Let $\langle b, a_1, a_2, \ldots a_n \rangle$ or $\langle b, A \rangle$ denote the contents of an itemset, and $c(\langle b, A \rangle)$ denote the support for this itemset. For a given item b, $\pi(b)$ denotes its immediate parent its value taxonomy, or 'root' for flat sets.

Table 3: The most interesting specific rules for products.

	b		A				Doc.
Int.	**Product Name**		**Attitude**	**Target Type**	**Orientation**	**Polarity**	**class**
45.7	**Peg Perego Pliko Matic (1)**	⇐	quality	this-product	positive	unmarked	
42.8	Lexmark Color JetPrinter 1100	⇐	reaction	this-product	negative	unmarked	neg
41.9	Peg Perego Milano XL	⇐	reaction	this-product	positive	unmarked	pos
41.1	Peg Perego Pliko Matic	⇐	reaction	this-product	positive	unmarked	
40.8	Peg Perego Milano XL	⇐	quality	this-product	positive	unmarked	pos
37.5	Peg Perego Milano XL	⇐	reaction	this-product	positive	unmarked	
37.1	Peg Perego Milano XL	⇐	quality	this-product	positive	unmarked	
36.3	**Agfa ePhoto Smile (2)**	⇐	reaction	experience	negative	unmarked	neg
36.0	**Agfa ePhoto Smile (2)**	⇐	reaction	experience	negative		neg
33.9	KB Gear KG-JC3S Jamcam	⇐	quality	experience	negative		neg

Table 4: The most interesting oppositional rules for products.

	b		A				Doc.
Int.	**Product Name**		**Attitude**	**Target**	**Orient.**	**Polarity**	**class**
31.6	**Lexmark Color JetPrinter 1100 (3)**	⇐	reaction	this-product	positive		neg
31.5	Lexmark Color JetPrinter 1100	⇐	quality	this-product	positive		neg
29.5	Lexmark Color JetPrinter 1100	⇐	reaction	this-product	positive	unmarked	neg
29.2	Lexmark Color JetPrinter 1100	⇐	quality	this-product	positive	unmarked	neg
28.9	Lexmark Color JetPrinter 1100	⇐	appreciation	this-product	positive		neg

For each item set, we collect rules $\langle b, A\rangle$ and compute their interestingness relative to the itemset $\langle \pi(b), A\rangle$. Interestingness is defined as follows:

$$Int = \frac{P(A|b)}{P(A|\pi(b))} = \frac{c(\langle b, A\rangle) \times c(\langle \pi(b)\rangle)}{c(\langle \pi(b), A\rangle) \times c(\langle b\rangle)}$$

Int is the relative probability of finding the child itemset in an appraisal expression, compared to finding it in a parent itemset. Values greater than 1 indicate that the child itemset appears more frequently than we would expect.

We applied two simple filters to the output, to help find more meaningful results. **Specificity** requires that b be a product name, and that attitude type and thing type be sufficiently deep nodes in the hierarchy to describe something specific. (For example, 'product thing' gives no real information about what part of the product is being appraised.) **Opposition** chooses rules with a different rating than the review as a whole, that is, document classification is the opposite of appraisal orientation. The filter also ensures that thing type is sufficiently specific, as with specificity, and requires that b be a product name.

We present the ten most 'interesting' rules from each filter, for the products corpus. Rules from the specificity filter are shown in Table 3 and rules from the opposition filter are shown in Table 4. We consider the meaning of some of these rules.

The first specificity rule **(1)** describes a typical example of users who like the product very well overall. An example sentence that created this rule says 'Not only is it an *excellent stroller*, because of it's [sic] size it even doubled for us as a portable crib.'

The specificity rules for the Agfa ePhoto Smile Digital Camera **(2)** are an example of the kind of rule we expect to see when bad user experience contributes to bad reviews. The text of the reviews that gave these rules quite clearly convey that users were not happy specifically with the photo quality.

In the oppositional rules for the Lexmark Color JetPrinter 1100 **(3)**, we see that users made positive comments about the product overall, while nevertheless giving the product a negative review. Drilling down into the text, we can see some examples of reviews like 'On the surface it looks like a *good printer* but it has many flaws that cause it to be frustrating.'

8 Conclusions

We have presented a new task, *appraisal expression extraction*, which, we suggest, is a fundamental tasks for sentiment analysis. Shallow parsing based on a set of appraisal lexicons, together with sparse use of syntactic dependencies, can be used to effectively address the subtask of extracting adjectival appraisal expressions. Indeed, straightforward data mining applied to appraisal expressions can yield insights into public opinion as expressed in patterns of evaluative language in a corpus of product reviews.

Immediate future work includes extending the approach to include other types of appraisal expressions, such as where an attitude is expressed via a noun or a verb. In this regard, we will be examining extension of existing methods for automatically building lexicons of positive/negative words (Turney, 2002; Esuli and Sebastiani, 2005) to the more complex task of estimating also attitude type and force. As well, a key problem is the fact that evaluative language is often *context-dependent*, and so proper interpretation must consider interactions between a given phrase and its larger textual context.

References

Rakesh Agrawal and Ramakrishnan Srikant. 1995. Mining generalized association rules. In Umeshwar Dayal, Peter M. D. Gray, and Shojiro Nishio, editors, *Proc. 21st Int. Conf. Very Large Data Bases, VLDB*, pages 407–419. Morgan Kaufmann, 11–15 September.

Eric Brill. 1992. A simple rule-based part of speech tagger. In *Proc. of ACL Conference on Applied Natural Language Processing*.

Yejin Choi, Claire Cardie, Ellen Riloff, and Siddharth Patwardhan. 2005. Identifying sources of opinions with conditional random fields and extraction patterns. *Human Language Technology Conference/Conference on Empirical Methods in Natural Language Processing (HLT-EMNLP 2005)*, October.

Andrea Esuli and Fabrizio Sebastiani. 2005. Determining the semantic orientation of terms through gloss analysis. In *Proceedings of CIKM-05, the ACM SIGIR Conference on Information and Knowledge Management*, Bremen, DE.

Michael Gamon, Anthony Aue, Simon Corston-Oliver, and Eric Ringger. 2005. Pulse: Mining customer opinions from free text. In *Proceedings of IDA-05, the 6th International Symposium on Intelligent Data Analysis*, Lecture Notes in Computer Science, Madrid, ES. Springer-Verlag.

Dan Klein and Christopher D. Manning. 2003. Accurate unlexicalized parsing. In *Proceedings of the 41st Meeting of the Association for Computational Linguistics*.

J. R. Martin and P. R. R. White. 2005. *The Language of Evaluation: Appraisal in English*. Palgrave, London. (`http://grammatics.com/appraisal/`).

George A. Miller. 1995. Wordnet: A lexical database for English. *Commun. ACM*, 38(11):39–41.

Tony Mullen and Nigel Collier. 2004. Sentiment analysis using support vector machines with diverse information sources. In *Proceedings of EMNLP-04, 9th Conference on Empirical Methods in Natural Language Processing*, Barcelon, ES.

Kamal Nigam and Matthew Hurst. 2004. Towards a robust metric of opinion. In *Proceedings of the AAAI Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications*, Standford, US.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proc. 42nd ACL*, pages 271–278, Barcelona, Spain, July.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? sentiment classification using machine learning techniques. In *Proceedings of EMNLP*.

Ana-Maria Popescu and Oren Etzioni. 2005. Extracting product features and opinions from reviews. In *Proceedings of HLT-EMNLP-05, the Human Language Technology Conference/Conference on Empirical Methods in Natural Language Processing*, Vancouver, CA.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of EMNLP*.

Peter D. Turney. 2002. Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. In *Proceedings 40th Annual Meeting of the ACL (ACL'02)*, pages 417–424, Philadelphia, Pennsylvania.

Jianyong Wang, Jiawei Han, and Jian Pei. 2003. CLOSET+: searching for the best strategies for mining frequent closed itemsets. In Pedro Domingos, Christos Faloutsos, Ted Senator, Hillol Kargupta, and Lise Getoor, editors, *Proceedings of the ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-03)*, pages 236–245, New York, August 24–27. ACM Press.

Casey Whitelaw, Navendu Garg, and Shlomo Argamon. 2005. Using appraisal taxonomies for sentiment analysis. In *Proceedings of CIKM-05, the ACM SIGIR Conference on Information and Knowledge Management*, Bremen, DE.

Janyce Wiebe, Theresa Wilson, Rebecca Bruce, Matthew Bell, and Melanie Martin. 2004. Learning subjective language. *Computational Linguistics*, 30(3).

Theresa Wilson, Janyce Wiebe, and Paul Hoffmann. 2005. Recognizing contextual polarity in phrase-level sentiment analysis. In *Proceedings of Human Language Technologies Conference/Conference on Empirical Methods in Natural Language Processing (HLT/EMNLP 2005)*, Vancouver, CA.