Just how mad are you? Finding strong and weak opinion clauses

Theresa Wilson
Intelligent Systems Program
University of Pittsburgh
Pittsburgh, PA 15260
twilson@cs.pitt.edu

Janyce Wiebe and **Rebecca Hwa**
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
{wiebe,hwa}@cs.pitt.edu

Abstract

There has been a recent swell of interest in the automatic identification and extraction of opinions and emotions in text. In this paper, we present the first experimental results classifying the strength of opinions and other types of subjectivity and classifying the subjectivity of deeply nested clauses. We use a wide range of features, including new syntactic features developed for opinion recognition. In 10-fold cross-validation experiments using support vector regression, we achieve improvements in mean-squared error over baseline ranging from 57% to 64%.

Introduction

There has been a recent swell of interest in the automatic identification and extraction of attitudes, opinions, and sentiments in text. Strong motivation for this task comes from the desire to provide tools and support for information analysts in government, commercial, and political domains, who want to be able to automatically track attitudes and feelings in the news and on-line forums. How do people feel about the latest camera phone? Is there a change in the support for the new Medicare bill? A system that could automatically identify and extract opinions and emotions from text would be an enormous help to someone sifting through the vast amounts of news and web data, trying to answer these kinds of questions.

Researchers from many different areas of AI have been working on the automatic identification of opinions and related tasks. To date, most such work has focused on classification at the document or sentence level. Document classification tasks include picking out editorials from among news articles and classifying reviews as positive or negative. A common sentence level task is to classify sentences as subjective or objective.

However, for many applications, just identifying opinionated sentences may not be sufficient. In the news, it is not uncommon to find two or more opinions in a single sentence, or to find a sentence containing opinions as well as factual information. An information extraction system trying to distinguish between factual information (which should be extracted) and non-factual information (which should be discarded or labeled uncertain) would find it helpful to be able to pinpoint the particular clauses that contain opinions. This ability would also be important for multi-perspective question answering, which aims to present multiple answers to the user based on opinions derived from different sources, and for multi-document summarization systems, which need to summarize differing opinions and perspectives.

Many applications would benefit from being able to determine not just whether something is opinionated but also the *strength* of the opinion. Flame detection systems want to identify strong rants and emotional tirades, while letting milder opinions pass through. Information analysts need to recognize changes over time in the virulence expressed by persons or groups of interest, and to detect when rhetoric is heating up, or cooling down.

This paper presents the first research in automatic opinion or sentiment classification to classify the clauses of every sentence in the corpus. Also, where other research has focused on distinguishing between subjective and objective or positive and negative language, we address the task of classifying the *strength* of the opinions and emotions being expressed in individual clauses, considering clauses down to four levels deep. A strength of *neutral* corresponds to the absence of opinion and subjectivity, so our strength classification task subsumes the task of classifying language as subjective versus objective.

Because the variety of words and phrases that people use to express opinions is staggering, a system limited to a fixed vocabulary will be unable to identify opinionated language over a broad range of discourse. A broad-coverage approach will require knowledge of subjective language that is truly comprehensive in scope. In this spirit, we use a wide range of features for the experiments in this paper—new syntactic clues that we developed for opinion recognition, as well as a variety of subjectivity clues from the literature. We found that these features can be adapted to the task of strength recognition, and that the best classification results are achieved when all types of features are used.

We present experiments in strength classification using boosting, rule learning, and support vector regression. In 10-fold cross validation experiments, we achieve significant improvements over baseline mean-squared error and accuracy for all algorithms.

Strong and Weak Subjective Expressions

Subjective expressions are words and phrases that express opinions, emotions, sentiments, speculations, etc. A general covering term for such states, from (Quirk et al., 1985), is *private state*, "a state that is not open to objective observation or verification." There are three main ways that private states are expressed in language: direct mentions of private states, speech events expressing private states, and *expressive subjective elements* (Banfield, 1982). An example of a direct private state is "fears" in (1). An example of a speech event expressing a private state is the one referred to by "continued" in (2).

(1) "The US fears a spill-over," said Xirao-Nima.

(2) "The report is full of absurdities," he continued.

Sentence (2) also contains an example of an expressive subjective element, namely "full of absurdities". With expressive subjective elements, sarcasm, emotion, evaluation, etc. are expressed through the way something is described or through particular wording. The subjective *strength* of a word or phrase is the strength of the opinion, emotion, or other private state that it expresses.

An Annotated Corpus of Opinions

In 2003, the Multi-perspective Question Answering (MPQA) corpus of opinion annotations (Wilson and Wiebe, 2003) was released. In the corpus, individual expressions are marked that correspond to explicit mentions of private states, speech events, and expressive subjective elements. A key aspect of the annotation project was that annotators were asked to judge all expressions in context. The result is an amazing variety of annotated expressions. Out of all the subjective expressions marked in the MPQA corpus, fully 53% are unique strings.

Each expression that is marked is characterized by a number of attributes: who is expressing the opinion, who or what is the target of the opinion, the type of attitude expressed by the opinion, and, key for our purposes, its subjective strength. In the annotation scheme, strength is marked as one of four values: *neutral*, *low*, *medium*, and *high*.[1] *Neutral* refers to the absence of opinion. Sentence (3) gives examples of strength annotations in the MPQA corpus.

(3) President Mohammad Khatami of Iran, whose
attempt at reforms have gotten American
<*low*>support</>, <*high*>accused</> the
United States of "<*high*>warmongering</>."

Inter-annotator agreement for strength ratings is challenging. It is not unusual for two annotators to identify the same expression in the text, but to differ in how they mark the boundaries. This in turn affects how they judge the strengths of the annotations. For example, (4) below shows how the same subjective phrase was judged by two annotators.

(4a) <*high*>imperative for harmonious society</>

(4b) <*medium*>imperative</> for <*medium*>harmonious</> society

Also, different people have different mental scales for what they consider strong and weak. *Low* strength to one annotator might be *medium* to another. For the annotations in the MPQA corpus, no specific attempt was made to align the strength scales of the different annotators.

Because of these challenges, as expected, absolute percent agreement for strength judgments on expressions is not high, on average 61%. However, measuring how often two annotators agree in their *ordering* of annotations by strength yields an average pairwise agreement of 95%, computed as follows. Let S_A and S_B be the sets of annotations identified by annotators A and B respectively. If a pair of annotations, $a \in S_A$ and $b \in S_B$, overlap, then a and b are a *matched pair*. Let S_{AB} be all possible combinations of matched pairs ab. Given two matched pairs $(ab)_i$ and $(ab)_j$ from S_{AB}, A and B agree on the ordering if:

$$strength(a_i) \geq strength(a_j) \wedge strength(b_i) \geq strength(b_j)$$

or

$$strength(a_i) \leq strength(a_j) \wedge strength(b_i) \leq strength(b_j)$$

Let M be the number of matched pairs for which A and B agree on the ordering. Then, $agreement = \frac{M}{|S_{AB}|}$.

Exploring Strength

An examination of the annotated data shows not only that a huge variety of expressions have been marked, but that strong subjectivity in particular is expressed in many different ways. We can think of some words that are clearly strong, such as "reckless" and "praise", as well as obvious modifications to these that increase or decrease their strength, as in "not reckless", "very reckless" and "high praise." It is unlikely, though, that expressions like "rhetorical petards" and "hell-bent" readily come to mind, both of which are marked in the annotations.

Expressions marked *high* often contain words that are very infrequent. For example, the word "petards" appears only once in the corpus. Collocations like "at all" add punch to an expression, as in, "at all costs" and "not true at all." It is also important to have knowledge of patterns like "expressed <direct-object>," which can generalize to many different phrases, such as "expressed hope," "expressed concern," "expressed gratitude," and "expressed some understanding." Also, there are syntactic modifications and syntactic patterns that have subjective force. Besides those patterns that merely intensify a subjective word, for example "very <ADJECTIVE>", we find patterns that have a cumulative effect on strength: "terrorist and extremist," and "criticize and condemn." The clues used later in the strength classification experiments contain examples of all these kinds of subjective phenomena.

As can be seen from earlier example (3), sentences are often complex, with opinions of differing strengths being expressed by perhaps two or more agents. In (3), there is low-strength support being expressed by the United States, as well as high-strength negative accusations coming from Khatami. In the MPQA corpus, 31% of sentences are made up of clauses that differ in strength by two or more strength

[1] *High* actually breaks down into *high* and *extremely-high* ratings. The *extremely-high* ratings are folded into the *high* ratings, rather than being treated as a separate rating, because they are rare.

ratings. This highlights the need to identify opinions at the clause level, as we do in our experiments.

Many other researchers are interested in polarity, another attribute of subjective language. We find some interesting interactions between polarity and strength in the data. The annotators were asked to judge the polarity of expressions that they marked, using an attribute called *attitude-type* that has values *positive*, *negative*, and *other*. The annotations show that annotators are often not comfortable with *positive* and *negative*: 22% of all *attitude-type* labels are *other*. However, the annotations also reveal that the stronger the expression, the clearer the polarity. Only 8% of the high-strength annotations are marked as *other*, while 39% of the low-strength annotations are so marked. In addition to stronger expressions having clearer polarity, stronger expressions of opinions and emotions also tend to be more negative in this corpus. Only 33% of low-strength annotations are negative, compared to 78% of high-strength annotations. These observations lead us to believe that the strength of subjective expressions will be informative for recognizing polarity, and vice versa.

Subjectivity Clues

In this section, we describe the information that we use for automatic strength classification. In addition to a wide variety of previously established subjectivity clues, we introduce a collection of new syntactic clues that are correlated with subjective language.

Previously Established Types of Clues

Previous work in subjectivity identification has supplied the research community with a large stable of subjectivity clues. These clues (**PREV**) include words and phrases culled from manually developed resources, others learned from annotated data, and others learned from unannotated data.

Due to the broad range of clues and their sources, the set of **PREV** clues is not limited to a fixed word list or to words of a particular part of speech. The clues from manually developed resources include entries from (Levin, 1993; Ballmer and Brennenstuhl, 1981), Framenet lemmas with frame element *experiencer* (Baker et al., 1998), adjectives manually annotated for polarity (Hatzivassiloglou and McKeown, 1997), and subjectivity clues listed in (Wiebe, 1990). Clues learned from annotated data include distributionally similar adjectives and verbs (Wiebe, 2000) and n-grams (Dave et al., 2003; Wiebe et al., 2001). From unannotated data, we have extraction patterns and subjective nouns learned using two different bootstrapping algorithms and a set of seed words (Riloff et al., 2003; Riloff and Wiebe, 2003). Finally, low-frequency words, which require no training to identify (Wiebe et al., 2001), are also used as clues.

A few of the **PREV** clues require more explanation. First, extraction patterns are lexico-syntactic patterns typically used by information extraction systems to identify relevant information. Riloff and Wiebe (2003) show that AutoSlog-TS, an algorithm for automatically generating extraction patterns, is able to find extraction patterns that are correlated with subjectivity. An example of a subjective extraction pattern is *<subj> dealt blow*, which matches phrases like "the mistake dealt a stiff blow to his pride."

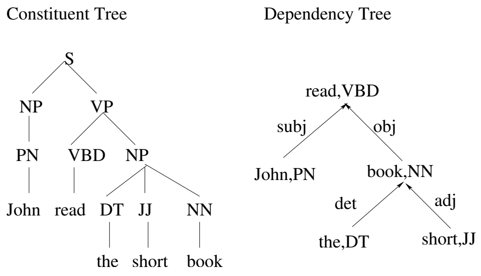

Figure 1: The constituent tree for "John read the short book" is on the left, and the dependency representation is on the right.

Interestingly, low-frequency words are informative for subjectivity recognition. We use low frequency words as clues; we consider a word to have low frequency if it appears $\leq$ 3 times in the document containing it plus a 1-million word corpus of news articles. In addition, we use n-gram clues from (Wiebe et al., 2001) that have fillers matching low-frequency words. When these clues were learned, the fillers matched low frequency words in the training data. When used during testing, the fillers are matched against low-frequency words in the test data. Examples of such n-grams are *<LowFreq-verb> and <LowFeq-verb>* and *so <LowFreq-adj>*.

Syntax Clues

The syntactic clues are developed by using a mostly-supervised learning procedure. The training data is based on both a human annotated (the MPQA) corpus and a large unannotated corpus in which sentences are automatically identified as subjective or objective through a bootstrapping algorithm (Riloff and Wiebe, 2003). The learning procedure consists of three steps.

First, we parse the training sentences in the MPQA corpus with a broad-coverage lexicalized English parser (Collins, 1997). The output constituent trees are automatically converted into their dependency representations (Xia and Palmer, 2001). In a dependency representation, every node in the tree structure is a surface word (i.e., there are no abstract nodes such as NP or VP), but each word may have additional attributes such as its part-of-speech (POS) tag. The parent word is known as the *head*, and its children are its *modifiers*. The edge between a parent and a child node specifies the grammatical relationship between the two words (e.g., *subj*, *obj*, and *adj*). Figure 1 shows the dependency parse tree for a sentence, along with the corresponding constituent representation, for comparison. For this study, we use 48 POS tags and 24 grammatical relationships.

Next, we form five classes of syntactic clues from each word w in every dependency parse tree.

root(w, t**):** word w with POS tag t is the root of a dependency tree (i.e., the main verb of the sentence).

leaf(w, t): word w with POS tag t is a leaf in a dependency tree (i.e., it has no modifiers).

node(w, t): word w with POS tag t.

bilex(w, t, r, w_c, t_c): word w with POS tag t is modified by word w_c with POS tag t_c, and the grammatical relationship between them is r.

allkids($w, t, r_1, w_1, t_1, \ldots, r_n, w_n, t_n$): word w with POS tag t has n children. Each child word w_i has POS tag t_i and modifies w with grammatical relationship r_i, where $1 \leq i \leq n$.

For each class specified above, we also consider less specific variants that back off to only POS tags. For example, $bilex(t, r, t_c)$ considers the cases in which any word with a POS tag t is modified by a word with POS tag t_c with grammatical relationship r.

Finally, we evaluate the collected clues. A clue is considered to be *potentially useful* if more than x% of its occurrences in the MPQA training data are in phrases marked as subjective, where x is a parameter tuned on development data (in our experiments, we chose $x = 70\%$). Potentially useful clues are further categorized into one of three *reliability* levels. First, a useful clue is *highly reliable* if it occurs frequently in the MPQA training data. For those that occur fewer than five times, we check their reliability on the larger corpus of automatically identified subjective and objective sentences. Clues that do not occur in the larger unannotated corpus are considered *not very reliable*. Clues that occur in the subjective set at least y times more than in the objective set are considered *somewhat reliable* (y is tuned on the development data and is set to 4 in our experiments), and the rest are rejected as not useful clues.

Feature Organization

Given the large number of **PREV** and **SYNTAX** clues, we are faced with the question of how best to organize them into features for strength classification. We tried a representation in which each clue is a separate feature, but it gave poor results. Instead, we adopt the strategy from (Riloff et al., 2003) of aggregating clues into sets, and creating one feature per set. The value of each feature is the number of instances in the sentence or clause of all the members of the set.

We define 29 features for the **PREV** clues reflecting how they were presented in the original research. For example, there are two features for the polar adjectives in (Hatzivassiloglou and McKeown, 1997), one for the set of positive adjectives and one for the set of negative adjectives. These 29 features are collectively called **PREV-type** in the experiments below. In addition, we define 15 features for the **SYNTAX** clues. For example, one feature represents the set of highly-reliable *bilex* clues. These features are called **SYNTAX-type**.

Although the above sets of subjectivity clues were selected because of their correlation with subjective language, they are not necessarily geared to discriminate between strong and weak subjectivity, and the groupings of clues into sets were not created with strength in mind. We hypothesized that a feature organization that takes into consideration the potential strength of clues would do better for strength classification.

To adapt the clues to strength classification, we use the annotations in the training data to filter the clues and organize them into new sets based on strength. For each clue c and strength rating s, we calculate the $P(strength(c)) = s$ as the probability of c being in an annotation of strength s. For $s = neutral$, this is the probability of c being in a neutral-strength annotation or in no annotation at all. If $P(strength(c)) = s \geq T$, for some threshold T, we put c in the set for strength s. In our experiments, we set $T = (P(strength(word)) = s) + 0.25$ or 0.95 if $(P(strength(word)) = s) + 0.25 \geq 1$. The value 0.25 was determined using experiments on a small amount of development data, held out from the experiment data for parameter tuning. It is possible for a clue to be in more than one set.

When **PREV** and **SYNTAX** clues are used in this feature organization they are called **PREV-strength** and **SYNTAX-strength** .

Experiments in Automatic Strength Classification

It is important to classify the strength of clauses, but pinpointing subjectivity at deeper levels can be challenging because there is less information to use for classification. To study the feasibility of automatically classifying clauses by their subjective strength, we conducted a suite of experiments in which a strength classifier is trained based on the features previously described. We wished to confirm three hypotheses. First, it is possible to classify the strength of clauses, for those that are deeply nested as well as those at the sentence level. Second, classifying the strength of subjectivity depends on a wide variety of features, including both lexical and syntactic clues. Third, organizing features by strength is beneficial.

To test our hypotheses, we performed the experiments under different settings, varying three factors: the learning algorithm used to train the classifiers, the depth of the clauses to be classified, and the types of features used. We vary the learning algorithm to explore its effect on the classification task. In our studies, the three machine learning algorithms are boosting, rule learning, and support vector regression. For boosting, we use BoosTexter (Schapire and Singer, 2000) AdaBoost.HM with 1000 rounds of boosting. For rule learning, we use Ripper (Cohen, 1995). For support vector regression we use SVMlight (Joachims, 1999) and discretize the resulting output into the ordinal strength classes. These algorithms were chosen because they have successfully been used for a number of natural language processing tasks.

We vary the depth of clauses to determine the effect of clausal depth on system performance. In our experiments, clauses are determined based on the non-leaf verbs in the parse tree, parsed using the Collins parser and converted to the dependency representation described earlier. For example, sentence (5) has three clauses, corresponding to the verbs "driven," "refused," and "give."

(5) They were driven out by rival warlord Saif Ullah, who has refused to give up power.

The clause defined for "driven" (level 1) is the entire sentence; the clause for "refused" (level 2) is "has refused to give up power"; and the clause for "give" (level 3) is "to give up power."

The gold standard strength ratings of sentences and clauses are based on the individual expression annotations: the strength of a sentence or clause is defined to be the highest strength rating of any expression in that sentence or clause.

In setting up experiments for classifying nested clauses, either clauses of the same or different levels may be classified in the training and testing phases. In the experiments below, the training examples are always entire sentences regardless of the clause level being classified during testing. Preliminary results showed that this configuration is better than training and testing at the same level.

All experimental results reported are averages over 10-fold cross validation using 9313 sentences from the MPQA corpus. Significance is measured using a 1-tailed t-test. For each experiment, both mean-squared error and classification accuracy are given. Although raw accuracy is important, not all misclassifications should be weighted equally for the task of strength classification. If the true strength of a sentence or clause is *high*, classifying it as *neutral* (off by 3) is a much worse error than classifying it as *medium* (off by 1). Mean-squared error captures this distinction, and, for this task, it is perhaps more important than accuracy as a metric for evaluation. If t_i is the true strength of sentence i, and p_i is the predicted strength of sentence i,

$$\text{mean-squared error (MSE)} = \frac{1}{n}\sum_{i}^{n}(t_i - p_i)^2$$

where n is the number of sentences or clauses being classified.

Classification Results

Tables 1 and 2 show strength classification results for clauses of depth 1–4. Table 1 gives results for BoosTexter and Table 2 gives results for Ripper and SVMlight.

The first row of Table 1 gives MSE and accuracies for a baseline classifier that chooses the most frequent class. Note that the distribution changes for clauses at different levels, giving higher baseline accuracies for more nested levels. The remaining rows in Table 1 show the BoosTexter results using different sets of features. Row 2 gives the results for a classifier trained using bag-of-words (**BAG**), where the words in each sentence are given to the classification algorithm as features. Rows 3 and 4 give the results for classifiers using **PREV-type + SYNTAX-type** features and **PREV-strength + SYNTAX-strength** features. The next two rows give the results for combining the two feature organizations with bag-of-words. The last row of the table shows the results when the **SYNTAX-strength** features are excluded from the best experiment.

The results for strength classification are promising for clauses at all levels of nesting. In Table 1, all of the improvements over baseline in MSE and accuracy are significant. The experiment in row 6 using **BAG + PREV-strength + SYNTAX-strength** features gives the best results. The improvements in MSE over baseline range from 48% to 60%, and the improvements in accuracy range from 23% to 79%. Table 2 rows 1 and 3 give the results for the same feature set using Ripper and SVMlight. Note that BoosTexter and Ripper are non-ordinal classification algorithms, whereas support vector regression takes into account ordinal values. This difference is reflected in the results. The results are comparable for BoosTexter and Ripper (MSE is not significantly different; BoosTexter has slightly better accuracy). Although accuracies are lower, the regression algorithm achieves much better MSE, improving 10% to 20% over BoosTexter and 57% to 64% over baseline, coming closer to the true strength at all levels[2].

The best experiments (Table 1 row 6 and Table 2 rows 1 and 3) use all the features, supporting our hypothesis that using a wide variety of features is effective. For boosting, the improvements over bag-of-words are significant (compare rows 2 and 6 in Table 1): from 20% to 25% for MSE and from 7% to 12% for accuracy. Results for Ripper and SVMlight (not shown) are similar. The new syntax clues contribute information over and above bag-of-words and the previous clues. For all learning algorithms and all clause levels, removing the syntax clues results in a significant difference in MSE (compare rows 6 and 7 in Table 1, rows 1 and 2 in Table 2, and rows 3 and 4 in Table 2). The differences in accuracy are also significant, with the exception of BoosTexter levels 1 and 2 and Ripper level 4.

Turning to feature organization, we see that organizing features by strength is beneficial. Comparing rows 3 and 4 in Table 1, the strength-based organization shows significant improvements across the row. Row 6 improves over row 5 for all values. All differences except for levels 3 and 4 MSE are significant. Results for Ripper and SVMlight using the strength-based feature organization (not given) show similar improvements.

Related Work

Research in automatic opinion and sentiment recognition includes distinguishing subjective from objective language (Yu and Hatzivassiloglou, 2003; Riloff et al., 2003; Riloff and Wiebe, 2003), distinguishing positive from negative language (Yu and Hatzivassiloglou, 2003; Turney and Littman, 2003; Pang et al., 2002; Dave et al., 2003; Nasukawa and Yi, 2003; Morinaga et al., 2002), and recognizing particular types of attitudes (Gordon et al., 2003; Liu et al., 2003). Ours are the first results to automatically distinguishing between not only subjective and objective (*neutral*) language, but among weak, medium, and strong subjectivity as well. Researchers who have identified opinions below the sentence level have restricted their attention to particular words and phrases (Turney and Littman, 2003; Pang et al., 2002; Dave et al., 2003; Nasukawa and Yi, 2003; Morinaga et al., 2002; Gordon et al., 2003; Liu et al., 2003). In contrast, this paper presents the first work classifying nested clauses in all sentences in the corpus.

[2] In future work we plan to experiment with more sophisticated ordinal regression algorithms (Herbrich et al., 1999).

	level 1		level 2		level 3		level 4	
	MSE	Acc	MSE	Acc	MSE	Acc	MSE	Acc
(1) baseline	1.921	30.8	2.749	41.8	2.538	45.9	2.507	48.3
(2) BAG	1.234	50.9	1.390	53.1	1.534	53.6	1.613	53.0
(3) PREV-type + SYNTAX-type	1.135	50.2	1.267	53.4	1.339	54.7	1.410	55.5
(4) PREV-strength + SYNTAX-strength	1.060	54.1	1.180	56.9	1.258	57.9	1.269	60.3
(5) BAG + PREV-type + SYNTAX-type	1.069	52.0	1.178	54.8	1.267	55.9	1.321	56.8
(6) **BAG + PREV-strength + SYNTAX-strength**	**0.991**	**55.0**	**1.111**	**57.0**	**1.225**	**57.5**	**1.211**	**59.4**
(7) BAG + PREV-strength	1.081	54.1	1.205	56.0	1.364	55.4	1.363	57.0

Table 1: Classification results using BoosTexter.

		level 1		level 2		level 3		level 4	
		MSE	Acc	MSE	Acc	MSE	Acc	MSE	Acc
Ripper	(1) **BAG + PREV-strength + SYNTAX-strength**	**1.004**	**53.2**	**1.138**	**55.3**	**1.220**	**55.9**	**1.244**	**57.8**
Ripper	(2) BAG + PREV-strength	1.230	50.7	1.347	53.5	1.458	54.3	1.459	56.6
SVMlight	(3) **BAG + PREV-strength + SYNTAX-strength**	**0.793**	**48.3**	**0.979**	**36.3**	**1.071**	**32.1**	**1.084**	**29.4**
SVMlight	(4) BAG + PREV-strength	0.849	43.5	1.164	31.2	1.300	27.4	1.346	25.2

Table 2: Classification results using Ripper and SVMlight.

Automatic opinion extraction is being applied in a number of interesting applications. Tong (2001) tracks sentiment timelines in on-line discussions. Many researchers classify reviews as positive and negative (Turney and Littman, 2003; Pang et al., 2002; Dave et al., 2003; Nasukawa and Yi, 2003; Morinaga et al., 2002). Others perform automatic analyses of product reputations (Morinaga et al., 2002; Nasukawa and Yi, 2003; Yi et al., 2003). Das and Chen (2001) examine the relationship between public sentiment in message boards and stock prices. All such applications would benefit from the rich subjectivity analysis performed by our system.

Conclusions

This paper presents promising results in identifying opinions in deeply nested clauses and classifying their strengths. We use a wide range of features, including new syntactic features. In 10-fold cross-validation experiments using boosting, we achieve improvements over baseline mean-squared error ranging from 48% to 60% and improvements in accuracy ranging from 23% to 79%. Experiments using support vector regression show even stronger mean-squared error results, with improvements ranging from 57% to 64% over baseline. Applications such as question answering and summarization will benefit from the rich subjectivity analysis performed by our system.

Acknowledgments

We would like to thank the reviewers for their helpful comments, Paul Hoffmann for programming support, and Adam Lopez for providing his dependency grammatical relationship labeling program. This work was supported in part by the National Science Foundation under grant IIS-0208798.

References

C. Baker, C. Fillmore, and J. Lowe. 1998. The Berkeley framenet project. In *Proceedings of the COLING-ACL*.

T. Ballmer and W. Brennenstuhl. 1981. *Speech Act Classification: A Study in the Lexical Analysis of English Speech Activity Verbs*. Springer-Verlag.

Ann Banfield. 1982. *Unspeakable Sentences*. Routledge and Kegan Paul, Boston.

William Cohen. 1995. Learning trees and rules with set-valued features. In *Proceedings of the Twelfth International Conference on Machine Learning (ICML-95)*.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, pages 16–23.

S. R. Das and M. Y. Chen. 2001. Yahoo! for Amazon: Opinion extraction from small talk on the web. In *Proceedings of the 8th Asia Pacific Finance Association Annual Conference*.

Kushal Dave, Steve Lawrence, and David M. Pennock. 2003. Mining the peanut gallery: Opinion extraction and semantic classification of produce reviews. In *Proceedings of the 12th International World Wide Web Conference (WWW2003)*. Web Proceedings.

Andrew Gordon, Abe Kazemzadeh, Anish Nair, and Milena Petrova. 2003. Recognizing expressions of commonsense psychology in English text. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (ACL-03)*, pages 208–215.

Vasileios Hatzivassiloglou and Kathy McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, pages 174–181.

Ralf Herbrich, Thore Graepel, and Klaus Obermayer. 1999. Support vector learning for ordinal regression. In *Proceedings of the 9th International Conference on Artificial Neural Networks*.

T. Joachims. 1999. Making large-scale SVM learning practical. In B. Scholkopf, C. Burgess, and A. Smola, editors, *Advances in Kernel Methods – Support Vector Learning*. MIT-Press.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago.

H. Liu, H. Lieberman, and T. Selker. 2003. A model of textual affect sensing using real-world knowledge. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI-2003)*.

S. Morinaga, K. Yamanishi, K. Tateishi, and T. Fukushima. 2002. Mining product reputations on the web. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2002)*.

T. Nasukawa and J. Yi. 2003. Sentiment analysis: Capturing favorability using natural language processing. In *Proceedings of the 2nd International Conference on Knowledge Capture (K-CAP 2003)*.

Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. 2002. Thumbs up? Sentiment classification using machine learning techniques. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)*, pages 79–86.

Randolph Quirk, Sidney Greenbaum, Geoffry Leech, and Jan Svartvik. 1985. *A Comprehensive Grammar of the English Language.* Longman, New York.

Ellen Riloff and Janyce Wiebe. 2003. Learning extraction patterns for subjective expressions. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, pages 105–112.

Ellen Riloff, Janyce Wiebe, and Theresa Wilson. 2003. Learning subjective nouns using extraction pattern bootstrapping. In *Proceedings of the 7th Conference on Natural Language Learning (CoNLL-2003)*, pages 25–32.

Robert E. Schapire and Yoram Singer. 2000. BoosTexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

Richard Tong. 2001. An operational system for detecting and tracking opinions in on-line discussions. In *Working Notes of the SIGIR Workshop on Operational Text Classification*, pages 1–6.

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

Janyce Wiebe, Theresa Wilson, and Matthew Bell. 2001. Identifying collocations for recognizing opinions. In *Proceedings of the ACL-01 Workshop on Collocation: Computational Extraction, Analysis, and Exploitation*, pages 24–31.

Janyce Wiebe. 1990. *Recognizing Subjective Sentences: A Computational Investigation of Narrative Text.* Ph.D. thesis, State University of New York at Buffalo.

Janyce Wiebe. 2000. Learning subjective adjectives from corpora. In *Proceedings of the Seventeenth National Conference on Artificial Intelligence (AAAI-2000)*, pages 735–740.

Theresa Wilson and Janyce Wiebe. 2003. Annotating opinions in the world press. In *Proceedings of the 4th ACL SIGdial Workshop on Discourse and Dialogue (SIGdial-03)*, pages 13–22.

F. Xia and M. Palmer. 2001. Converting dependency structures to phrase structures. In *Proceedings of the Human Language Technology Conference (HLT-2001)*.

J. Yi, T. Nasukawa, R. Bunescu, and W. Niblack. 2003. Sentiment analyzer: Extracting sentiments about a given topic using natural language processing techniques. In *Proceedings of the 3rd IEEE International Conference on Data Mining (ICDM-2003)*.

Hong Yu and Vasileios Hatzivassiloglou. 2003. Towards answering opinion questions: Separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP-2003)*, pages 129–136.