# Product Rating Prediction with Online Reviews Using Support Vector Machines

Chun-Yu Chen, Jyun-Wei Huang, Richard Tzong-Han Tsai∗

Department of Computer Science and Engineering, Yuan Ze University, Taiwan

∗corresponding author

{s951559, s976017}@mail.yzu.edu.tw

thtsai@saturn.yzu.edu.tw

## Abstract

More and more shopping websites allow customers to post online reviews on products, allowing customers to share opinions and information on specific products. Reviews can be expressed in text, ratings, or both. Text-based reviews give detailed information on a product while ratings can be quickly understood. Numerical ratings are especially important when screen size is limited. However, not all customers assign ratings to text reviews, and some text-based reviews are inconsistent with their corresponding numerical ratings. In this paper, we outline a method of mapping text-based reviews to numerical ratings using an SVM classifier. Three linguistic feature types are employed in our SVM-based classifier. Given the very large number of product reviews, only features that can be efficiently extracted are employed. Since it is difficult for customers to distinguish adjacent ratings (e.g. 4 and 5), we have adopted relaxed criteria for evaluating our system precision. According to experimental results, our method achieves a precision of over 76.6% using the relaxed criteria, which is sufficient to automatically annotate text reviews with numerical ratings.

Keywords: support vector machines, product review, product rating, customer feedback

## 1. Introduction

Today most of the biggest online retailers provide feedback and review capabilities to their customers. The websites of large volume sellers, such as Amazon.com and Newegg.com, quickly amass huge numbers of customer product reviews. For popular items, the reviews may number in the hundreds, making it sometimes difficult for consumers to go through them all and arrive at an informed decision. In such a situation, it becomes helpful for consumers to have a summary of all the reviews to consult [1, 2]. Review aggregator websites such as Consumersearch.com collect and summarize reviews manually, but it could potentially save effort if text mining could be put to use to do part or all of this time-consuming and costly process.

One of the effective types of review summarization is directly scoring products. Especially the customers need to make their decisions by reading online comments on their mobile phones within limited time. If ratings are provided, it is more convenient for customers to make buying decision[3-5]. If time is limited, they can firstly pick the products with satisfactory ratings (e.g. over three). Then read the text-based reviews of the selected products. This two level strategy saves the efforts of reading all text-based reviews.

According to our survey of 100 customers, the attitudes of customers toward different ratings are obviously different. In our survey, we assume the rating is from 1 to 5, like Amazon.com and Buy.com. Most of the shopping websites are also rating products range from one to five. One means the lowest rating while five means the highest. We observed that 92% customers do not buy products with average ratings between 1 and 3. 77%

customers will buy products with ratings in 3 to 4 under some special conditions (e.g., special discount). Products with ratings in 4 to 5 are most likely (over 90%) to be bought if there is not any very bad review existed.

In this paper, we investigate the possibility of automatically classifying online comments into finer-grained classes rather than only three. We also analyze the characteristics of comments in different ratings. Through our analysis and experiments, which features apparently influence the ratings and which features are effective for predicting online product ratings are explored.

## 2. Our Approach

### 2.1 System Architecture

Our system uses Conditional Random Fields (CRFs) as the underlying machine learning model to extract opinion phrases from reviews. We generate features based on opinion phrases and other text information in reviews and use these features to train our SVM model. Figure 1 depicts the overall process flow.

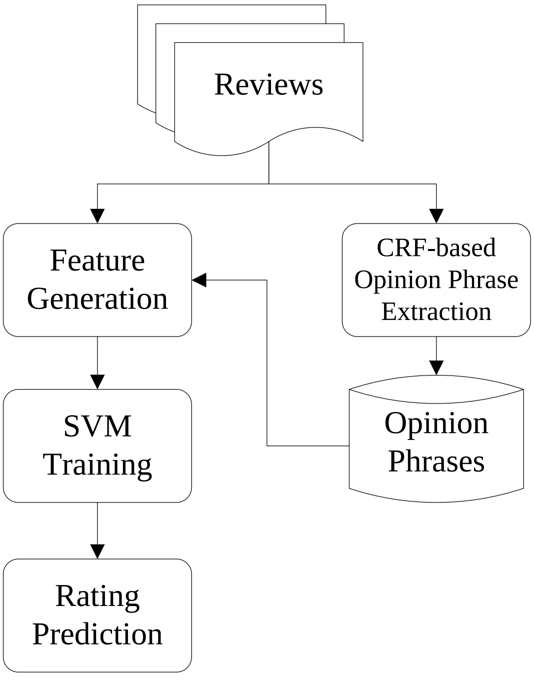

Figure 1. System architecture

### 2.2 CRF-based Opinion Phrase Extraction

One important subtask of rating prediction is to label all opinion phrases. There are two main approaches to extracting opinion phrases: dictionary-based and machine-learning-based. Dictionary-based is easier to implement, but it does not consider contextual information, which can be very useful for determining opinion phrases [6]. Our system to employs a machine-learning-based approach.

Conditional random fields (CRF) [7] is currently the best known sequence tagging machine learning algorithm. The advantages of CRF have been demonstrated in many natural language processing and text mining tasks [8-10]. Therefore, we adopted the CRF model as our underlying ML algorithm.

We constructed a corpus of 600 reviews for training our CRF-based opinion phrase extractor. All opinion phrases are annotated by humans. E.g., the sentence “Nice looking frame, small footprint, quick refresh rate” is annotated as follows:

<OP> Nice </OP> looking frame , <OP> small </OP> footprint , <OP> quick </OP> refresh rate

, where OP is the abbreviation of “Opinion Phrase”.

To compile the training data of the CRF model, the above annotation needs to be converted into the IOB2 format [11]. In IOB2, each word in a sentence is regarded as a token, and each token is associated with a tag that indicates the category of the OP and whether the given token is at the beginning (*B*), or inside (*I*) of the OP. That is, *B_OP* and, *I_OP* denote, respectively, the first token and the subsequent token of an OP. In addition, we use the tag *O* to indicate that a token does not belong to any OP. Once we have tokenized a sentence, we can define the OP extraction problem as the assignment of one of 2+1 tags to each token. For example, the above phrase annotated in XML format is transformed to the following IOB2 format:

“Nice/*B-OP* looking/*O* frame/*O* ,/O small/*B-OP* footprint/*O* ,/*O* quick/*B-OP* refresh/*O* rate/*O*”

## 2.3 Rating Prediction

### 2.3.1 Support Vector Machines

The support vector machine (SVM) model is one of the best known Machine Learning models that can handle sparse high dimension data, which has been proved useful for text classification [12]. It tries to find a maximal-margin separating hyperplane <**w**, $\varphi$(**x**)> - $b$ = 0 to separate the training instances, i.e.,

$$\min \|\mathbf{w}\|^2 + C\sum_i \xi^{(i)} \qquad \text{subject to}$$

$$\gamma^{(i)}\left(< \mathbf{w}, \varphi\left(\mathbf{x}^{(i)}\right) > -b\right) \geq 1 - \xi^{(i)}, \ \forall i$$

where $\mathbf{x}^{(i)}$ is the $i$th training instance which is mapped into a high-dimension space by $\varphi(\cdot)$, $\gamma_i \in \{1, -1\}$ is its label, $\xi^{(i)}$ denotes its training error, and $C$ is the cost factor (penalty of the misclassified data). The mapping function $\varphi(\cdot)$ and the cost factor $C$ are the main parameters of a SVM model.

When classifying an instance **x**, the decision function f(**x**) indicates that **x** is "above" or "below" the hyperplane. Cristianini [13] shows that the f(**x**) can be converted into an equivalent dual form which can be more easily computed:

$$\text{primal form}: \mathrm{f}(\mathbf{x}) = \mathrm{sign}(< \mathbf{w}, \varphi(\mathbf{x}) > -b)$$

$$\text{dual form}: \mathrm{f}(\mathbf{x}) = \mathrm{sign}(\sum_i \alpha^{(i)}\gamma^{(i)}\,\mathrm{K}\left(\mathbf{x}^{(i)}, \mathbf{x}\right) - b)$$

where $\mathrm{K}(\mathbf{x}^{(i)}, \mathbf{x}) = <\varphi(\mathbf{x}^{(i)}), \varphi(\mathbf{x})>$ is the kernel function and $\alpha^{(i)}$ can be thought of as $\omega$'s transformation.

In our experiment, we also use F-score for selecting features [14] and give a weight to each data point, the cost factor C is chosen to be 1, which is fairly suitable for most problems.

### 2.3.2 Feature Type 1: Pros and Cons Field Word Count (PCWC)

We assume that customers leave more positive than negative comments if they liked the product. Checking the reviews one by one, we found that higher ratings did correlate to higher word counts in the pros field. Likewise, the higher the cons word count was, the lower the product rating. This correlation is the reason we chose pros and cons word counts as our first two features.

### 2.3.3 Feature Type 2: Pros and Cons Field Opinion Phrase Count (PCOPC)

We selected relevant opinion phrases from our 600-review corpus and manually annotated them as positive or negative. We then counted the number of positive and negative opinion phrases appearing in the pros and cons fields, which gave us four features:

(1) Positive phrases in pros field

(2) Positive phrases in cons field

(3) Negative phrases in pros field

(4) Negative phrases in cons field

### 2.3.4 Feature Type 3: Opinion Phrase TFIDF (OPTFIDF)

Opinion phrases are the key information that reveals the reviewer's sentiment or rating. If an opinion phrases has a high TF-IDF value, it may be more helpful for classifying a review. Therefore, we created a feature for each opinion phrase. Each opinion phrase's TFIDF value in the given review is used as its corresponding feature value.

## 3. Experiment and Analysis

### 3.1 Review Data from Newegg.com

We use review information from the popular online computer retailer Newegg.com to build our dataset for these experiments. Newegg website reviews consist of a 5-star ("5-egg") rating plus pros and cons fields for written feedback. The latter function is especially helpful for differentiating positive and negative feedback; some review sites provide only a single comment field. We compiled all reviews on a popular LCD monitor for our dataset.

There are 1706 reviews in the dataset. The 'egg-rating' distribution is displayed in the following table.

Table 1. Dataset

| Rating | Reviews |
|---|---|
| One egg | 225 |
| Two eggs | 210 |
| Three eggs | 384 |
| Four eggs | 467 |
| Five eggs | 420 |

### 3.1 Evaluation Result

We employed three-fold cross-validation to evaluate our system: each fold is used once in turn as the test set, with the remaining two folds being used as the training set. We present the classification results in Table 2.

Table 2. Confusion matrix

| | | Predicted rating | | | | |
|---|---|---|---|---|---|---|
| | | **1** | **2** | **3** | **4** | **5** |
| Gold-standard rating | **1** | **28** | 7 | 95 | 62 | 33 |
| | **2** | 21 | **10** | 86 | 65 | 28 |
| | **3** | 18 | 11 | **145** | 140 | 70 |
| | **4** | 12 | 3 | 79 | **253** | 120 |
| | **5** | 3 | 0 | 21 | 117 | **279** |

We calculated the precision and the recall of each rating class according to the confusion matrix, as shown in Table 3. The precision and recall are not as good as expected, which we explain in more detail in the following paragraphs.

Table 3. Original performance

| Rating | Precision | Recall |
|---|---|---|
| One egg | 42.15% | 12.52% |
| Two eggs | 36.63% | 4.75% |
| Three eggs | 34.02% | 37.73% |
| Four eggs | 39.77% | 54.17% |
| Five eggs | 52.74% | 66.44% |

### Four and Five Eggs

In Table 2, we can see that reviews with five eggs are most often incorrectly classified as four eggs and vice versa. This is highly dependent on users' subjectivity. Some users give a product five eggs even if the product has a small drawback, while others give it four. It is very hard to distinguish reviews with five eggs from those with four. The following example is a five-egg review with a small drawback in the Cons field.

> ***Pros****: I ordered this monitor to upgrade from my 17" CRT, and all i can say is this monitor is AMAZING. 0 Dead Pixels, fast shipping, awesome color, and best of all its pretty cheap compared to most monitors.*
>
> ***Cons****: No HDMI ports.The stand is a little flimsy.*
>
> ***Product Rating:*** ●●●●●

### Three Eggs

Products rated as three eggs tend to be incorrectly classified as four or five eggs. This may be because they usually contain several positive statements. The boundary that separates three-egg and four-egg review is not very clear. The following example is a three-egg review misclassified as a four:

> ***Pros****: Right price for this. Nice large screen. Great color, contrast, and resolution.*
>
> ***Cons****: No height adjustment. No tilt adjustment.*
>
> ***Product Rating:*** ●●●○○

### Two and One Egg

There are several reasons why the one-egg and two-egg classes have the lowest performance. Firstly, there are usually fewer of these reviews and, therefore, less training data. Secondly, due to the presence of a Pros field, most of these reviews contain at least one or two positive phrases, even if these are used in a sarcastic manner (e.g. "Great product… until it died"). Many people are also in the habit of prefacing negative reviews with one or two positive comments, to make their evaluation seem fair or well-reasoned. The reverse does not seem to apply to glowing reviews to the same degree. Lastly, the negative reviews tend to contain fewer total words, making it difficult for the learning model to classify.

Example of situation 1: sarcastic manner

> ***Pros****: Great picture, when it worked*
>
> ***Cons****: Decided to not recieve a video signal after 2 months*
>
> ***Product Rating:*** ●●○○○

Example of situation 2: negative reviews with one or two positive comments

> ***Pros**: Same monitor that I had before. Incredibly saturated colors, large, relatively cheap. HDCP works, unlike the previous one I RMA.*
>
> ***Cons**: Nice little clump of 6 dead black pixels in a 2x3 rectangle.*
>
> ***Product Rating:*** ●●○○○

Example of situation 3: negative reviews tend to contain fewer total words

> ***Pros**: GREAT PRICE*
>
> ***Cons**: BAD PRODUCT*
>
> ***Product Rating:*** ●○○○○

### 3.2 Relaxed Evaluation

After analyzing our first set of evaluation results, we observed that reviews with close ratings (i.e., 1 and 2, 2 and 3, 3 and 4, 4 and 5) look very similar. Therefore, we adopted more meaningful relaxed evaluation criteria: reviews classified as their gold-standard ratings plus or minus 1 (e.g., three eggs classified as two or four eggs) are considered correct. The results of the relaxed evaluation are shown in Table 4.

Table 4. Performance measured in relaxed evaluation

| Rating | Precision |
|---|---|
| One egg | 67.69% |
| Two eggs | 86.54% |
| Three eggs | 73.16% |
| Four eggs | 80.13% |
| Five eggs | 75.36% |

### 3.3 Other Errors

Sometimes negative reviews with many words in the Pros field are incorrectly classified as three eggs. Below is an example of a one-egg review in which the customer complained in the Pros field about the product seemingly misunderstanding the field's purpose. Predictably, this leads to the wrong classification. In addition, we also found that there are some typos in the review. This also influences our model's classification.

> ***Pros:** I bought this monitor, over 5 months ago, It wast nice at first arrived with no dead pixels, I even checked it to make sure. It made my games look nice, Then i noticed a black dot on the bottom of the screen, Then a few days later another. Then i seen a hair on my screen, went to brush it away, It did not move, Then i relaized it was behind the screen, Then i noticed the monitors flaw. If you look at its shiny black face plate, You will notice the seam lines up perfectly with the lcd screens first layer. Allowing just enough of a gap to let particles in. This flaw is what is causeing my dead pixels. Right now has i write this i have over 15 dead pixels from this prob. I wrote to acer about it. But they play dumb and act like its the user's fault. In a way im kicking my self for buying this becasue my gf warned me about a laptop she got from im. Wich dyed out before it was even 5 months old. Looks like i will have to suck it up and buy another brand of lcd*
>
> ***Cons:** Acer support is not very good, There dead pixel policy is not very good, 4 dead pixels and 1 got to be in the middle before they even consider looking at it. It will cost you money to even have im do that*
>
> ***Product Rating:*** ●○○○○

***Typos:***

- *No "complaints" at all.*
- *"sooo" nice.*
- *Person with high tech knowledge "recommended".*
- *"Lightbleeding" on the top and bottom*

## 4. Conclusion

In this paper we have presented an approach to predicting numerical customer product review ratings from text-based reviews with an SVM classifier using three linguistic feature types. Given the difficulty of distinguishing adjacent ratings (customer subjectivity), we have used more reasonable relaxed criteria to evaluate system precision. Our experimental results using these criteria show that our method achieves a precision of over 76.6%, which is sufficient to automatically annotate text reviews with numerical ratings. In the future, we hope to improve prediction of negative reviews and develop other tools to detect and filter inconsistent or noise reviews based on the techniques proposed here.

## References

[1] Y.-T. L. a. H.-H. C. L.-W. Ku, "Opinion Extraction, Summarization and Tracking in News and Blog Corpora," in *AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs* Stanford University, California, 2006, pp. 100-107.

[2] M. Hu and B. Liu, "Mining and summarizing customer reviews," in *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining* Seattle, WA, USA: ACM, 2004.

[3] D. F. Cox and S. U. Rich, "Perceived Risk and Consumer Decision-Making: The Case of Telephone Shopping," 1964.

[4] J. Arndt, "Role of product-related conversations in the diffusion of a new product," *Journal of Marketing Research,* vol. 4, pp. 291-295, 1967.

[5] J. J. Brown and P. H. Reingen, "Social Ties and Word-of-Mouth Referral Behavior," *Journal of Consumer Research,* vol. 14, p. 350, 1987.

[6] R. T.-H. Tsai, S.-H. Wu, and W.-L. Hsu, "Mencius: A Chinese Named Entity Recognizer Based on a Maximum Entropy Framework," *Computational Linguistics and Chinese Language Processing,* vol. 9, pp. 65-82, 2004.

[7] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of the Eighteenth International Conference on Machine Learning*: Morgan Kaufmann Publishers Inc., 2001.

[8] A. McCallum and W. Li, "Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons," in *Proceedings of the seventh conference on Natural language learning at HLT-NAACL 2003 - Volume 4* Edmonton, Canada: Association for Computational Linguistics, 2003.

[9] F. Sha and F. Pereira, "Shallow parsing with conditional random fields," in *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1* Edmonton, Canada: Association for Computational Linguistics, 2003.

[10] R. Tsai, C.-L. Sung, H.-J. Dai, H.-C. Hung, T.-Y. Sung, and W.-L. Hsu, "NERBio: using selected word conjunctions, term normalization, and global patterns to improve biomedical named entity recognition," *BMC Bioinformatics,* vol. 7, 2006.

[11] E. F. T. K. Sang and J. Veenstra, "Representing text chunks," in *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics* Bergen, Norway: Association for Computational Linguistics, 1999.

[12] T. Joachims, "Text Categorization with Suport Vector Machines: Learning with Many Relevant Features," in *Proceedings of*

*the 10th European Conference on Machine Learning*: Springer-Verlag, 1998.

[13] Cristianini N and S.-T. J, *An Introduction to Support Vector Machines*: Cambridge University Press, 2000.

[14] Y. W. C. a. C. J. Lin, *Combining SVMs with various feature selection strategies* vol. In Feature extraction, foundations and applications: Springer-Verlag, Berlin, 2006.