# UTILITY SCORING OF PRODUCT REVIEWS

Zhu Zhang, Department of
Management Information Systems,
University of Arizona

Balaji Varadarajan,
Department of Computer Science,
University of Arizona

Presented By,
Akarshit Kumar Verma (B19ME003),
Indian Institute of Technology Jodhpur, Rajasthan

## This?

*“Handling: The interface is similar to that in other Canon digital compacts, which helps your learning curve. The case is … daylight conditions.*
*Picture quality: Contrary to other comments, I was not “blown away” by quality of the pictures. The lens produces … low light conditions.*
*Complaints:*

- *Can’t review picture histogram easily (two to three steps)*
- *Noisy operations*
- *No battery level indicator”*

## or This?

*X is the greatest product I’ve ever seen.*

Two different reviews collected for Canon Digital Camera

“

“Utility Score is the score given to a review based on its usefulness from 0 to 1”

# Overview

- Problem Statement
- Datasets
- Approach
- Evaluation Metrics
- Experimental Results
- Conclusions

# 1. Problem Statement

## What's the Problem?

- Online shoppers generally go through other people's reviews of a specific product before actually buying it;
- Manufacturers can also examine product reviews to observe customer opinions and make improvements or predict market trends.
- Now it's undeniable that product reviews aren't equally "useful".

## What we will be doing?

- So our task is a of text sentiment analysis: predicting the utility of product reviews, which is orthogonal to polarity classification and opinion extraction.
- We build regression models and classification models by incorporating a diverse set of features and achieve highly competitive performance for utility scoring and review classification on real-world data sets.

# 2. Datasets

- We downloaded reviews and related data from CNET and AWS(Amazon Web Service) in three different domains: Electronics, Video, and Books.
- For electronics and books, we used keyword-based search to identify Canon products (*Canon*) and engineering books (*Engineering*); for videos, we used the "AudienceRating" field to retrieve PG- 13 movies (*PG-13*).
- For all customer-written reviews, we also collected the votes they get regarding their usefulness ("x out of y people found the following review helpful").

Structure of Dataset:

1.

Reviewer ID - ID of the reviewer, e.g. A2S UAM1J3GNN3B

2. asin - ID of the product, e.g. 0000013714

3. Reviewer Name - name of the reviewer

4.

helpful - helpfulness rating of the review, e.g. ⅔

a.

Numerator: Number of persons upvoted the review

b. Denominator:Number of persons the review

5. Review Text - text of the review

6. overall - rating of the product

7. summary - summary of the review

8.

Unix Review Time - time of the review (unix time)

9. Review Time - time of the review (raw)

# 3. Approach

## What are we trying to get?

We view the problem as one of regression. Formally, given a product review *T*, a number of features *f1(T),...,fj(T),..., fp (T)* can be computed. Our task is to approximate a function

$$u(T) = F(f1,...,fj,...,fp)$$

The output $u \in [0, 1]$ should reflect the real utility of *T* as accurately as possible.

## How we will achieve that?

In this, we have two aspects of the statistical learning framework:

- The learning (regression) algorithms and
- The features.

# 3(a). Learning Algorithms

## ε-Support Vector Regression (ε-SVR)

ε-Support Vector Regression (ε-SVR) implemented in LIBSVM.. Just as SVM represents the state of the art in machine learning, SVR is attractive due to the power of different kernel functions. In this, we use the radial basis kernel function (RBF), which handles the potentially non-linear relationship between the target value and features.

## Simple linear regression (SLR)

Simple linear regression (SLR) implemented in WEKA. SLR is a classical regression tool and has been widely used in numerous applications. It serves as a reasonable baseline.

# 3 (b) Features

## Lexical Similarity Features (LexSim)

- Clearly a review should not be a literal copy or loyal rephrase of the product specification S, and an editorial review E.
- Through this we measure the similarity between customer review and product specification, sim(T,S), and that between customer review and editorial review, sim(T,E), respectively.

## Shallow Syntactic Features (*ShallowSyn*)

- We compute counts of words with the following part-of- speech tags in T, in order to characterize the subjectivity- objectivity mixture of the text at a shallow syntactic level:
  Proper nouns, Numbers, Modal verbs, Interjections, Comparative and superlative adjectives, Comparative and superlative adverbs and Wh-determiners, wh-pronouns, and possessive wh- pronouns, wh-adverbs.

## Lexical Subjectivity Clues (*LexSubj)*

- This is a set of features, which we use to capture the subjectivity-objectivity mixture at a lexical semantic level, by taking advantage of lexical resources created by other researchers.
- Specifically, we calculate counts of words in the subjective adjectives from corpora. clue lists.

# 4. Evaluation Metrics

## What's the Target Value?

- Given a review text *Ti*, we have to operationalize the gold-standard definition of *u(Ti)*.
- Now almost every Amazon customer review comes with a vote regarding its usefulness ("*x* out of *y* people found the following review helpful"). This actually provides a direct and convenient way to approximate the gold-stand utility value of a given review. Formally, we define the utility as

  u=*x*/*y*

- In our experiments, we only use the reviews with at least 10 votes (i.e., $y > 10$), in order to ensure the robustness of the regression model.

## How to Compare?

Given an estimated function F, we can use the following metrics to evaluate its quality, both of which are standard in regression analysis:

- Squared correlation coefficient:

$$r^2 = \frac{(\sum_{i=1}^{n}(u_i - \bar{u})(\hat{u}_i - \bar{\hat{u}}))^2}{\sum_{i=1}^{n}(u_i - \bar{u})^2 \sum_{i=1}^{n}(\hat{u}_i - \bar{\hat{u}})^2}$$

- Mean Square Error:

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{n}(u_i - \hat{u}_i)^2$$

ui and u^i are the real and predicted utility scores respectively; u¯and u^ represent the mean of the corresponding sample respectively.

# Before Experimental Results

## Length v/s Utility

One might intuitively expect that the utility of a review strongly correlates with its length (in a positive way). However, as we can see from Table , the correlation between the two variables is in fact very weak. Therefore it is necessary to build non-trivial regression models.

| **Collection** | $r^2$ |
|---|---|
| *Canon* | 0.0042 |
| *Engineering* | 0.0997 |
| *PG-13* | 0.0853 |

# Experimental Results

## Learning Algorithm Comparison

- Across all three collections, the results are relatively similar qualitatively. The strongest model for each collection always achieves r^2 > 0.30 and σ^2 < 0.10, and apparently outperforms the length-based baseline.
- Generally speaking, SVR significantly outperforms SLR, mostly due to its regularized model and the non-linear power of its kernel function. Furthermore, since the prediction model only depends on a subset of the training data, SVR is more resistant to outliers, from which SLR sometimes suffer (e.g., producing very large mean squared error).

**(a) Canon Product**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0049 | 0.0957 | 0.0064 | 0.0800 |
| ShallowSync | 0.2726 | 0.0601 | 0.0433 | 0.0772 |
| LexSubj | 0.0448 | 0.0902 | 0.0081 | 0.0806 |
| ALL | **0.3028** | **0.0565** | 0.0892 | 0.0736 |

**(b) Engineering Book**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0216 | 0.0947 | 0.0232 | 0.0874 |
| ShallowSync | 0.31276 | 0.0615 | 0.0895 | 0.0816 |
| LexSubj | 0.0674 | 0.0907 | 0.0424 | 0.0857 |
| ALL | **0.3514** | **0.0581** | 0.1244 | 0.0786 |

**(c) PG-13 Movie**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0014 | 0.1484 | 0.0467 | 0.1347 |
| ShallowSync | 0.4176 | 0.0829 | 0.0905 | 0.1285 |
| LexSubj | 0.0412 | 0.1479 | 0.0244 | 0.1376 |
| ALL | **0.4145** | **0.0826** | 0.1571 | 0.1193 |

# Experimental Results

## Features Comparison

- The set of lexical similarity features play a very minor role in the regression model, this implies instead, the utility is based on inherent properties of the review itself.
- The lexical subjectivity clues have very limited influence on the utility scoring. This means that the perceived “usefulness” of a product review barely correlates with the subjectivity or polarity embedded in the text.
- The shallow syntactic features account for most predicting power of the regression model. This phenomenon demonstrates that high-utility reviews do stand out due to the linguistic styles in which they are written.

**(a) Canon Product**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0049 | 0.0957 | 0.0064 | 0.0800 |
| ShallowSync | 0.2726 | 0.0601 | 0.0433 | 0.0772 |
| LexSubj | 0.0448 | 0.0902 | 0.0081 | 0.0806 |
| ALL | **0.3028** | **0.0565** | 0.0892 | 0.0736 |

**(b) Engineering Book**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0216 | 0.0947 | 0.0232 | 0.0874 |
| ShallowSync | 0.31276 | 0.0615 | 0.0895 | 0.0816 |
| LexSubj | 0.0674 | 0.0907 | 0.0424 | 0.0857 |
| ALL | **0.3514** | **0.0581** | 0.1244 | 0.0786 |

**(c) PG-13 Movie**

| Feature Set | $\epsilon$-SVR | | SLR | |
|---|---|---|---|---|
| | $r^2$ | $\sigma^2$ | $r^2$ | $\sigma^2$ |
| LexSim | 0.0014 | 0.1484 | 0.0467 | 0.1347 |
| ShallowSync | 0.4176 | 0.0829 | 0.0905 | 0.1285 |
| LexSubj | 0.0412 | 0.1479 | 0.0244 | 0.1376 |
| ALL | **0.4145** | **0.0826** | 0.1571 | 0.1193 |

# Conclusions

We viewed the problem as one of regression, and built regression models by incorporating a diverse set of features, which achieved highly competitive performance on three Amazon product review collections. Two major things we observed :

- SVR performed better than SLR in general.
- In particular, the shallow syntactic features turned out to be the most influential predictors.
- Since the description and grammar used by users vary, the accuracies of classifier will depend highly on the data given. Thus, we can see that the predictor with combination of all features gives better accuracy than single feature predictors.

# References


- Support Vector Regression (SVR) — One of the Most Flexible Yet Robust Prediction Algorithms
- New Features for Sentiment Analysis: Do Sentences Matter?
- Sentiment Analysis: A Definitive Guide
- Natural Language Processing, Sentiment Analysis and Clinical Analytics

# Thanks!