|  |
| --- |
|  |
| **Formatting Instructions for NIPS 2013** |
|  |

**Alan RidwanCoauthor**

Department of Computer Science Affiliation

University of Washington Address

*hippo@cs.cranberry.lemon.edu* *email*

**Abstract**

This paper tests the ability to use machine-learning algorithms to predict the usefulness of consumer reviews. The primary algorithm used in this experiment is stochastic gradient descent for logistic regression and Least Square Error to measure results. The raw review data was provided by Yelp through Kaggle for a Machine Learning contest that has already ended. The paper will focus on feature selection, kernel testing, and result discussion and future recommendations.

**1 Intro**

Many companies have profited from the increase of available information on the Internet. Recommendations and review which people would once in the past only take from friends now are readily available on websites like Yelp. A huge problem in the market is which reviews to trust because not all of them are going to be useful. The natural process of founding useful reviews takes time, as more and more user read and vote on each one. Unfortunately the market is very competitive and users are not very patient, it be a great advantage to know which reviews are more useful than other even before any user have to read it.

In order to predict usefulness, it is essential to predict what is likely to make a review useful by feature selection. feature selection allow us to select a subset of information that we think is relevant to the problem we are trying to answer. In this case all feature selected are related to the usefulness of reviews. Using our selected features as vectors, we will run the stochastic gradient descent for logistic regression algorithm to test of prediction error.

This paper is organized in section, with section 1 being introductions. Section 2 focus of the data used and the features selected. Section 3 is about experiment and results. Section 4 will be the conclusion and future recommendations.

**2 Data and Features**

The data, Yelp provided on Kaggle, contains information on 11,537 businesses, 8282

checkin sets, 43,873 users and 229,907 reviews. Each of these 4 data types are store in a separate Jason formatted file. Upon getting the data we immediately parsed and stored each file in to a separate but related SQL database. This allows specific data to be easily queried for during feature selection. We mainly focus on two of the 2 data types.

review

{

'type': 'review',

'business\_id': (encrypted business id),

'user\_id': (encrypted user id),

'stars': (star rating, rounded to half-stars),

'text': (review text),

'date': (date, formatted like '2012-03-14'),

'votes': {(vote type): (count)}

}

user

{

'type': 'user',

'user\_id': (encrypted user id),

'name': (first name),

'review\_count': (review count),

'average\_stars': (floating point average, like 4.31),

'votes': {(vote type): (count)}

}

There are a number of different features that we have determined to effect how useful a review maybe. First we assume the length of a review would effect how useful the review is, because there would be more information in longer reviews. Second we assume the rating of a review would effect the usefulness of the review, because more opinionated a review is the more information it would give. Next the number of useful reviews written by a user would show how likely that user would write another useful review. Then the number of reviews a business has would also effect how useful the review is. Aside from the above features we also include Flesch–Kincaid reading level and reading ease as factors for usefulness because that would effect how much a person would read a review..

**3 Headings: first level**

In this experiment we determined if a review had at least 2 vote of usefulness then the review would be considered as useful.

With a Step size of 0.125 the L2 norm of 2088.29, we got the victor weights for features of

1] "Weights of feature vector at step size 0.125"

Base: -56.28003

Length: -81.06665

reading ease: -336.45032

reading level: -1273.93582

rating of review: -305.03348

ratio of useful reviews written: 1586.91766

number of reviews for business -61.15562

Base on the weights, we found that reading ease and ratio of useful reviews written greatly effects the usefulness of a review. When a review the easier to read it is more likely to be useful and users who write useful review is likely to write more useful reviews.

**4 Conclusion:**

This is a very bad algorithm for classifying review usefulness because of the high error. For the final version of the project we will use a different algorithm.