

Rating Prediction using Feature Words Extracted from Customer Reviews

Masanao Ochi¹

Makoto Okabe²

Rikio Onai³

The University of Electro-Communications^{1,2,3}, JST PRESTO²
1-5-1, Chofugaoka, Chofu, Tokyo, Japan^{1,2,3}
{ochi, onai}@onailab.com^{1,3} m.o@acm.org²

ABSTRACT

We developed a simple method of improving the accuracy of rating prediction using feature words extracted from customer reviews. Many rating predictors work well for a small and dense dataset of customer reviews. However, a practical dataset tends to be large and sparse, because it often includes too many products for each customer to buy and evaluate. Data sparseness reduces prediction accuracy. To improve accuracy, we reduced the dimension of the feature vector using feature words extracted by analyzing the relationship between ratings and accompanying review comments instead of using ratings. We applied our method to the Pranking algorithm and evaluated it on a corpus of golf course reviews supplied by a Japanese e-commerce company. We found that by successfully reducing data sparseness, our method improves prediction accuracy as measured using RankLoss.

Categories and Subject Descriptors

H.3.3 [Information Storage and Retrieval]: Information Search and Retrieval—*information filtering*

General Terms

Experimentation

Keywords

sentiment analysis, review mining, rating prediction

1. INTRODUCTION

Rating prediction is a practically important problem, because it enables the market to estimate how satisfied a customer will be with a service [1, 2, 5]. Given a dataset of customer reviews, rating predictors (like the Pranking algorithm [1]) can learn and predict the rating of a target customer. These algorithms work well for a small and dense dataset. However, the problem is that in practice a dataset is large and sparse, because it often includes too many products for each customer to buy and evaluate (Figure 1). The dimension of the feature vector increases as fast as the increase in the number of products. For example, the matrix of the EachMovie dataset is reportedly sparse (5% full) [5]. To address this problem, we developed a way to reduce the dimension of a feature vector using feature words extracted by analyzing the relationship between ratings and accompanying review comments. Figure 1 illustrates our approach. We

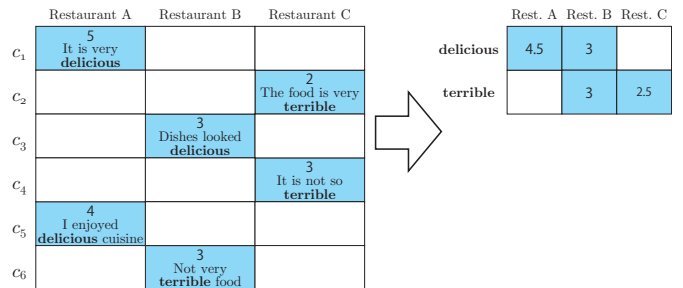


Figure 1: An illustration of our approach in the case of restaurant reviews. Left: The original dataset has ratings only in the blue cells and the whole matrix is sparse. Right: Instead of using customers, we use the feature vector of feature words extracted from the review comments. This reduces the dimension of the feature vector from 6 to 2, so the whole matrix becomes dense. The score of each cell is an average of the scores rated with each word.

reduced the dimension of the feature vector and increased the density of the matrix by extracting “delicious” and “terrible” as feature words. We extracted a word as a feature word when it appeared frequently and was accompanied with a polarized score. The words “delicious” and “terrible” appeared many times with high and low scores and therefore were selected as feature words. “Very” was not selected as a feature word; although it also appeared many times, its accompanying score varied. Our method of analyzing ratings and accompanying review comments is a simple but powerful way of extracting feature words that can capture a characteristic of each rating aspect well. This is often difficult to implement by relying only on natural language-processing techniques or other corpora such as WordNet [3, 4].

2. EFFECT OF DATA DENSITY

As shown in Figure 1, the original dataset is often sparse. Crammer et al. proposed filling in blank cells with an intermediate score [1], which results in low prediction accuracy. We evaluated the relationship between data sparseness and the accuracy of the Pranking algorithm. Given a target customer and a product, the Pranking algorithm predicts the rating that the customer is likely to give the product by learning from the dataset how the customer has rated other products in the past. We used the Book-Crossing Dataset¹, which is a relatively dense dataset because of the

¹<http://www.informatik.uni-freiburg.de/~cziegler/BX/>

Overall		word		average		word		average		Course		word		average		word		average		Cost performance		word		average	
great	4.6	nasty	2.5	Inoue(name)	4.5	lack	3.0	killer price	4.6	relative expense	3.0														
Rope(golf course)	4.5	cheap	2.6	Seiichi(name)	4.5	nasty	3.0	real cheap	4.6	penny-wise	3.1														
praise	4.5	bad	2.7	great	4.4	cheap	3.0	performance	4.5	asking	3.2														
splendid	4.4	asking	2.8	skillful	4.4	weed	3.0	better	4.4	terrible	3.2														
mindful	4.4	apology	2.8	Ishioka(golf course)	4.4	enforced	3.0	cost	4.4	apology	3.2														
beautiful	4.4	anger	2.8	agitate	4.4	river	3.1	CP	4.4	embarrassingly	3.3														
thoughtful	4.4	finally	2.8	motivate	4.3	sand pit	3.1	cheap	4.4	arrogant	3.3														
perfect	4.4	embarrassingly	2.8	rarity	4.3	monotone	3.1	fire-sale	4.3	messy	3.3														
rich	4.4	ass	2.9	variety	4.3	gully	3.1	great	4.3	ass	3.3														
thrilling	4.4	dirty	2.9	championship	4.3	patchy	3.1	outclassing	4.3	complaint	3.3														

Figure 2: Feature words extracted from customer reviews. All reviews were written in Japanese, and we translated each extracted word into English.

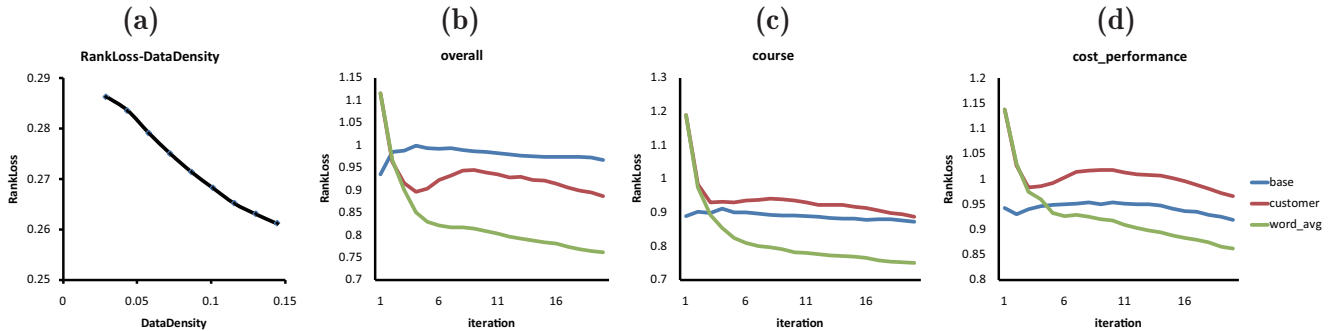


Figure 3: (a) The performance of a rating predictor depends on the data density. (b)-(d) The RankLoss for each aspect. Base means prediction of an intermediate score, 3 in this case. Customer means customer ratings are used as the feature vector. Word_avg means feature words are used as the feature vector.

nature of the service. The original matrix of this dataset is 15% full. We generated a sparse dataset by removing data from randomly selected cells and applied it to the Pranking algorithm. We evaluated prediction accuracy using RankLoss [1], $RankLoss = \sum_t^T \frac{|\hat{y}_t - y_t|}{T}$. T is the number of products, \hat{y}_t is the t -th predicted output score, and y_t is the t -th desired output score. As shown in Figure 3 (a), the sparser the dataset becomes, the lower the prediction accuracy becomes. This result motivated us to solve the problem of data sparseness by reducing the dimension of the feature vector.

3. EXPERIMENT AND RESULTS

We applied our technique to the Pranking algorithm and evaluated it on a corpus of golf course reviews supplied by a Japanese e-commerce company². Each review was accompanied by a set of three aspects, each on a scale of 1-5: overall experience, golf course, and cost performance. To extract feature words related to each aspect, we computed the score for each word as the average of all rated scores accompanied by a review comment that included the word. The score is shown as “average” in Figure 2. As a result, the score of a word irrelevant to the aspect was close to the total average score, and a word that captured the characteristic of the aspect had a high or low average score. Figure 2 shows that our method successfully extracts interesting feature words. Note that our method extracts not only positive and negative words but also words that explain the semantic context of the aspect. For example, “Inoue” and “Seiichi,” shown in the Course table, refer to a famous golf course designer who has designed many golf courses in Japan. The negative side of the Course table includes words such as “weed,” “river,” and “sand pit,” from which one learns that a customer’s low

rating is caused by complaints about the conditions of a golf course.

We used the extracted feature words as the feature vector of the Pranking algorithm and reduced the dimension to increase the density of the data. The original dataset had 85,000 customers and was sparse (0.5% full). We selected 100 feature words with high and low average scores, thereby reducing the dimension of the feature vector from 85,000 to 100. As a result, data density was increased (30% full). We evaluated the RankLoss (Figure 3 (b)-(d)), and our results outperformed the results of the original sparse dataset.

4. CONCLUSIONS

We have demonstrated that data density is important for prediction accuracy. We have developed a novel feature vector using feature words that successfully improves prediction accuracy as measured using RankLoss.

5. REFERENCES

- [1] K. Crammer. Pranking with ranking. *NIPS’01*, pages 641–647, 2001.
- [2] E. F. Harrington. Online ranking/collaborative filtering using the perceptron algorithm. *ICML’03*, pages 250–257, 2003.
- [3] M. Hu and B. Liu. Mining and summarizing customer reviews. *KDD’04*, pages 168–177, 2004.
- [4] J. Kamps, M. Marx, R. J. Mokken, and M. de Rijke. Using wordnet to measure semantic orientations of adjectives. *LREC’04*, pages 1115–11118, 2004.
- [5] A. Sashua and A. Levin. Ranking with large margin principle: Two approaches. *NIPS’03*, pages 937–944, 2002.

²<http://rit.rakuten.co.jp/rdr/index.html>