ISSN (Print) : 0974-6846
Indian Journal of Science and Technology, *Vol 9(18), DOI: 10.17485/ijst/2016/v9i18/93164, May 2016* ISSN (Online) : 0974-5645

A Neural Network based Model for Determining Overall Aspect Weights in Opinion Mining and Sentiment Analysis

Duc-Hong Pham[1] and Anh-Cuong Le[2*]

[1]VNU University of Engineering and Technology, Hanoicity, Vietnam; hongpd@epu.edu.vn
[2]Faculty of Information Technology, Ton Duc Thang University, Vietnam; leanhcuong@tdt.edu.vn

Abstract

Opinion mining and sentiment analysis has recently attracting many studies in the field of text mining and knowledge discovery. Its task is to identify and analyze subjective information of customers' opinion from social media sources in the Web. The two kinds of objects of this study are products and product's aspects which are given in reviews from sources such as social networks, merchant sites, blogs, forums. This paper focuses on determining the important degree of aspects given a set of reviews. Suppose that each given review is assigned with aspects' ratings and overall rating. Under our opinion, the overall rating of a review is derived from its aspect ratings. This observation inspires us to formulate these factors in a neural network. This proposed model can generate the weights for aspects which reflect aspects' important degrees. Doing experiment for this model, we use a dataset of 397528 reviews of 2558 hotels which are collected from an well known tourist website - *tripadvisor.com*. Five common aspects are used include "cleanliness", "location", "service", "room" and "value". The obtained experimental result shows that our proposed method outperforms some well known studies for the same problem such as the probabilistic rating regression method or the frequency-based method.

Keywords: Aspect-based Analysis, Aspect Weight, Neural Network, Overall Aspect Weights, Sentiment Analysis

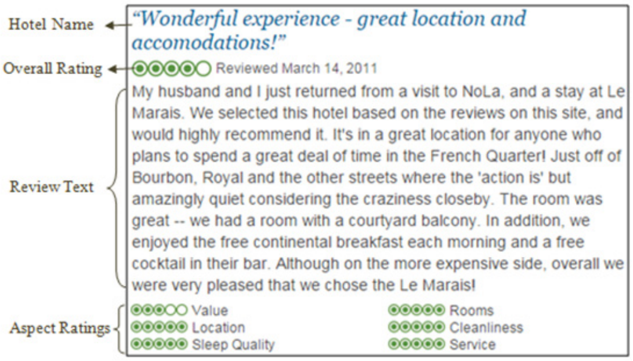
1. Introduction

The Internet has been developing very fast with new advanced features, specially people can give their opinions or sentiments about a product or service they bought or used. These opinions/sentiments can be expressed as posts or comments in social networks, forum, blogs, ect. This kind of information becomes important when a person would like to get helpful information before making decision of buying a product or using a service. It is also very useful for producers to know what should be improved in their products/services. To obtain adviced information from Internet, people usually to look for others' opinions (in general we consider they are expressed under reviews) from various resources in the Web. However, the number of such reviews has been increasing very fast that make more difficult for a person to find enough needed information, and it is also very difficult to understand the overall view.

With its importance, many studies have focused on how to extract and how to understand opinions from the Web. There are some basic tasks for this problem of opinion mining and sentiment analysis. The first one is subjectivity classification which aims at determining whether a review contains opinion (sentiment) or not. It is normally formulated as a classification problem in which a review text will be classified into subjective class or objective class. Many studies have applied statistical machine learning algorithms and extracted effective features to solve this problem, such as[1-4]. The other well known problem is polarity classification which aims to determine whether an opinioned text is positive, negative, or neutral. Polarity classification seems to be the most important task in this field and therefore attracting many

* *Author for correspondence*

studies such as[5-9]. Other studies such as[10-12] considered a different aspect of sentiment classification that is rating a review by assigning it a degree from 1 star to 5 stars as shown in many merchant websites such as *amazon.com*, *tripadvisor.com*. There are also other interesting and important problems: mining comparative opinions, opinion spam detection, and opinion lexicon generation.



"Wonderful experience - great location and accomodations!"

●●●●○ Reviewed March 14, 2011

My husband and I just returned from a visit to NoLa, and a stay at Le Marais. We selected this hotel based on the reviews on this site, and would highly recommend it. It's in a great location for anyone who plans to spend a great deal of time in the French Quarter! Just off of Bourbon, Royal and the other streets where the 'action is' but amazingly quiet considering the craziness closeby. The room was great -- we had a room with a courtyard balcony. In addition, we enjoyed the free continental breakfast each morning and a free cocktail in their bar. Although on the more expensive side, overall we were very pleased that we chose the Le Marais!

●●●●○ Value	●●●●● Rooms
●●●●● Location	●●●●● Cleanliness
●●●●● Sleep Quality	●●●●● Service

Figure 1. A Sample Hotel Review.

Recently, many researchers focus on aspect-based analysis, such as[13-15]. Actually, a review may mention several aspects of the product or service. Some merchant websites permit customers express not only the overall rating on the product/service but also the rating on separate aspects. You can see in the Figure 1 an example in which the customer has posted a review on a hotel with the overall rating and the separate ratings on its aspects named "cleanliness", "location", "service", "room" and "value". We can see some such typical systems as Amazon[1], Yelp[2] and Tripadvisor[3].

In this paper we also follow the trend of sentiment-based analysis. We will focus on determining the important degree of aspects. This task will answer the question that which aspect is important to customers? This results will help consumers make a right decision when buying. It also help producer to focus on improving the important aspects which doesn't satisfy consumers. Hence, the task of identifying important aspects are significant to both consumers and firms.

For this purpose, in this paper, we study a model for deriving aspect weights which influence their overall rating in reviews. If the weight of an aspect is high then it reflects that the aspect is important. In our proposed method, assuming that we have a training dataset in which each review is given with its overall ratings and aspect ratings, like the example in the Figure 1. We will build a new model using neural network for determining overall aspect weights which are objectives to compute the overall rating for all the reviews.

2. Related Works

In[13], the authors have assumed both the aspect ratings and aspect weights to be latent in reviews which will be determined by analyzing the review content and an observation that aspect ratings and aspect weights will derive the overall rating. An extension of this model is provided by[14] which is an unified generative model called Latent Aspect Rating Analysis. Another study, the paper[15] has proposed a model called Sparse Aspect Coding Model. This model determines users rating for each aspect given a review, which used two latent variables including user intrinsic aspect interest and item intrinsic aspect to discover a set of aspects.

We can see that in many websites a review can give us both overall ratings and aspect ratings. For example, the review in Figure 1, both overall rating and aspect ratings are given by consumers. This paper addresses the problem with this assumption. Recently, several works such as[16,17] are closest to our work. They developed an algorithm for probabilistic aspect ranking which is similar to probabilistic rating regression method in[13]. This study considered aspect frequency and the influence of each aspect over overall rating to determine the aspect importance. Note that, in this aspect ranking algorithm infer aspects' weights for each individual review and then averaging them to generate the overall aspect weight.

Different from previous studies, in this paper we observe that the overall aspect weights are not dependent on aspect weights of each individual review. The overall rating of review is generated based on the combination of the overall aspect weights and the aspect ratings. We use a neural network to determine the overall aspect weights by learning directly from the model with the objective to generating the given overall rating.

3. Problem Definition

We now define some concepts used in this study. Denote $D = \{d_1, d_2, ..., d_{|D|}\}$ as a set reviews represented as a text. Each review d is assigned with an overall rating O_d and a vector r_d of aspect ratings. A dictionary V is required to contain sentiment words to be used. Suppose that there

[1]www.amazon.com, [2]www.yelp.com, [3]www.tripadvisor.com

are *n* words in *V*, which express opinions for all aspects. We alo suppose that there are *k* different aspects to be considered in *D* denoted by a set $\{A_1, A_2, ..., A_k\}$. Each aspect A_i is represented by a set of words in V, that infers a rating factor for aspect A_i in the reviews. Denote $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)$ is the overall aspect weights vector for all reviews. Note t that α_i indicates the degree of importance corresponding to aspect A_i, where we require $0 \le \alpha_i \le 1$ and $\sum_{i=1}^{k} \alpha_i = 1$.

We define O_d is the overall rating of review *d* and $r_d(r_{d1}, r_{d2}, ..., r_{dk})$ is the *k* - dimensional vector of aspect ratings for all aspects in review *d*. In our model these aspect ratings will affect generating the overall rating. The problem is formulated by that: given *D* which is the set of reviews and for each review d ∈ D which is assigned with an overall rating O_d and assigned with a vector of aspect ratings $r_d(r_{d1}, r_{d2}, ..., r_{dk})$,where the component r_{di} is the aspect rating of aspect A_i. The goal here is to how to generate the aspect weights.

4. The Proposed Method

We assume that the overall rating of each review d ∈ D is generated by the linear combination of the overall aspect weights and the aspect ratings: $\sum_{i=1}^{k} \alpha_i . r_{di}$. This motivates us to apply a neural network for these factors, and thus this model will derive overall aspect weights from the training dataset. Figure 2 shows an illustration of our proposed model.

In this neural network model, let v be a weighted sum, g(v) is an activation function, we choose g(v) = v. We assume the overall aspect weights $\alpha = (\alpha_1, \alpha_2, ..., \alpha_k)$ are the weights of the input and they must satisfy the conditions: $\sum_{i=1}^{k} \alpha_i = 1$, $0 \le \alpha_i \le 1$.

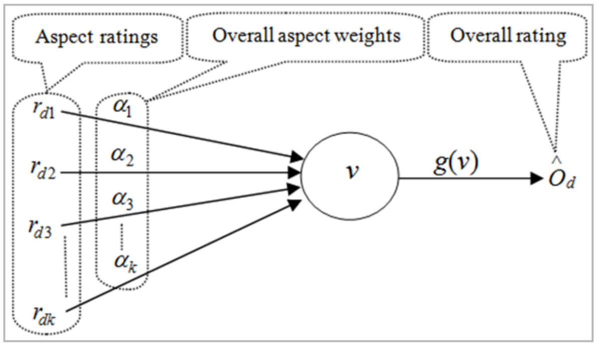


Figure 2. A neural network model for learning the overall aspect weights.

For review d , the input is the *k* - dimensional vector of *k* aspect ratings $r_d(r_{d1}, r_{d2}, ..., r_{dk})$, the output is the overall rating of dwhich is computed by:

$$\hat{O}_d = g(v) = g(\sum_{i=1}^{k} \alpha_i r_{di}) = \sum_{i=1}^{k} \alpha_i r_{di} \quad (1)$$

The basic idea of neural network is to adjust the weights of the network to reduce the deviation (i.e. error) between the output values and the target values in the data set. This work is often based on back-propagation algorithm to determine the weights of the network so that the error is minimum.

Denote O_d is the desired target values of the overall rating of review d then the mean square error function of the data set *D* is defined by:

$$E(\alpha) = \frac{1}{2} \sum_{d \in D} (O_d - \hat{O}_d)^2 \quad (2)$$

In order to support $\sum_{i=1}^{k} \alpha_i = 1$ and $0 \le \alpha_i \le 1$, we replace the aspect weight α_i by the auxiliary aspect weight $\hat{\alpha}_i$ as follows:

$$a_i = \frac{\exp(\hat{\alpha}_i)}{\sum_{l=1}^{k} \exp(\hat{\alpha}_l)} \quad (3)$$

Now, the fuction E (α) is replaced by the function $E(\hat{\alpha})$ as follows:

$$E(\hat{\alpha}) = \frac{1}{2} \sum_{d \in D} s_d(\hat{\alpha})^2 \quad (4)$$

where $s_d(\hat{\alpha}) = O_d - \sum_{i=1}^{k} \frac{\exp(\hat{\alpha}_i)}{\sum_{l=1}^{k} \exp(\hat{\alpha}_l)} . r_{di}$

The objective here is to how to determine the auxiliary aspect weights values $\hat{\alpha}$ subject to minimize the function $E(\hat{\alpha})$. This is the problem of nonlinear square optimization, which is usually solved by an iterative algorithm. We develop a back-propagation algorithm called NNAWs (i.e. Neural Network Aspect Weights) as follows:

At time *t = 0*, according to[17] the important aspects are usually commented on by a large number of consumers which are expressed by frequency of words in each aspect, therefore we initialize $\hat{\alpha}_i \in \hat{\alpha}$ by:

$$\hat{\alpha}_i = \frac{n_i}{\sum_{l=1}^{k} n_l} \quad (5)$$

where $n_i = \sum_{p=1}^{n} n_{ip}$ is the total counts of words in the segnmented text of aspect A_i, and n_{ip} is the frequency of the *p*-th word corresponding to aspect A_i (note that here we will use the aspect segmentation algorithm in[13] to determine the segmented text to the aspect A_i), $\sum_{l=1}^{k} n_l$ is the total counts of words in the text of all aspects. Then the two nested loops, in each iteration of the second loop consists of two phases: propagation and weight update.

Phases 1: propagation, the overall rating of review d at time *t* is given by the formula:

$$\hat{O}_d(t) = \sum_{i=1}^{k} \alpha_i(t).r_{di} \qquad (6)$$

Phases 2: weight update, each element of the weight vector $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2, ..., \hat{\alpha}_k)$ is updated at time *t + 1* based on the error between the output values and target values according to the following formula:

$$\hat{\alpha}_i(t+1) = \hat{\alpha}_i(t) + \Delta\hat{\alpha}_i(t); 1 \le i \le k \qquad (7)$$

Where $\Delta\hat{\alpha}_i(t) = -\eta \frac{\partial E(\hat{\alpha})}{\partial \hat{\alpha}_i(t)} = -\eta(O_d - \hat{O}_d(t)).\frac{\partial s_d(\hat{\alpha})}{\partial \hat{\alpha}_i}$, η is the learning rate.

And the gradient of $s_d(\hat{\alpha})$ with respect $\hat{\alpha}_i$ is,

$$\frac{\partial s_d(\hat{\alpha})}{\partial \hat{\alpha}_i} = -\sum_{l=1}^{k}(\delta(\mathrm{i{=}l})\alpha_i(1-\alpha_i)r_{di} - \delta(\mathrm{i} \ne l)\alpha_l\alpha_i r_{dl})$$

where $\delta(y) = \begin{cases} 1; \text{if } y = true \\ 0; \text{if } y = false \end{cases}$

Summary, the algorithm for determining the auxiliary aspect weights is presented in the Algorithm 1 below:

Algorithm 1: The algorithm called NNAWs for determining $\hat{\alpha}$

```
Input: The training set $D=\{(r_d, O_d)\}_{d=1}^{|D|}$, the learning rate η the error threshold ε and the iterative threshold I
Step 0: $t=0$; initialize $\hat{\alpha}$ according to Eq. (5);
Step 1: for $iter=0$ to $I$ do
 for each pair $(r_d, O_d) \in D$ do
1.1. compute $\alpha$ at time $t$ according to Eq. (3);
          1.2. compute $\hat{O}_d$ at time $t$ according to Eq. (6);
          1.3. update $\hat{\alpha}$ at time $t+1$ according to Eq. (7);
Step 2: Step 1 repeated until the mean error
$\frac{1}{|D|}\sum_{d \in D}|O_d - \hat{O}_d(t)|$ less than ε or completed the number of iterations.
Output: $\hat{\alpha}$
```

After obtaining the auxiliary aspect weights $\hat{\alpha}$, we compute each $\alpha_i \in \alpha$ according to Eq. (3), and then the overall aspect weights is estimated.

5. Experiment

5.1 Data

In our experiment, we use the dataset which contains 397528 reviews of 2558 hotels collected from the very famous tourist Website (*tripadvisor.com*). This data is also used in the work[13]. To avoid sparseness and missing aspect descriptions in reviews,[13] has combined all the reviews commenting on the same hotel into a new "review" called "h-review"and average the overall/aspect ratings over them as the ground truth ratings. The data thus includes 2558h-reviews. Table 1 shows an example of h-review.

Table 1. A sample format of h-review in the ground truth ratings

Hotel name	Values	Rooms	Location	Cleanliness	Service	Overall
Prince Conti Hotel	4.347	3.964	4.797	4.266	4.268	4.186
Pitti Palace al Ponte Vecchio	3.900	3.488	4.857	3.884	3.902	3.825
Ho Hotel Banys Orientals	4.155	3.864	4.699	4.179	3.707	3.983

For the dictionary *V* of terms/words which express opinion, we use the dictionary from the work in [13]. It consists of 4000 opinion words. In summary, Table 2 shows the statistics on the data in our experiment.

Table 2. Evaluation Corpus Statistics

Number of reviews	397528
Number of h-reviews	2558
Number of opinion words	4000
Number of aspects	5

5.2 Experimental Result

We set the iterative threshold I = 3000, the learning rate η = 0.015, and the error threshold ε = 0.00001 for the algorithm NNAWs. For initializing the auxiliary aspect weights, we first use the aspect segmentation algorithm in the experimental program[44] in [13] to assign an aspect label for each word in reviews. After aspect segmentation, we compute the frequency of words for each aspect and we use this result to initialize the auxiliary aspect weights according to Eq. (5).

We perform the algorithm NNAWs for determining (i.e. inferring) the overall aspect weights for all reviews. Denote α_{NNAWs} is the vector of the overall aspect weights and it is learned directly from there views by the algorithm NNAWs.

Table 3. Results of computing the overall aspect weights

Values	Rooms	Location	Cleanliness	Service
0.187	0.460	0.028	0.050	0.275

Table 3 shows the overall weights of each aspect. From this results we can see that the aspects rooms and service are the most important aspects. This information is valuable to the hotel managers because it can help them to have an overview of which aspects are important to customers.

Next, we select four groups of hotels, the first is the group of hotels with 5 stars, the second is the group of hotels with 4stars, the third is the group of hotels with 3 stars, the fourth is the group of hotels with 2 stars. We perform the algorithm NNAWs to determine aspect weights for each individual group hotel and the obtained results as shown in Table 4.

Table 4. The aspect weights of four groups of hotels

Aspect	Hotel 5 Stars	Hotel 4 Stars	Hotel 3 Stars	Hotel 2 Stars
Values	0.150	0.337	0.103	0.102
Rooms	0.195	0.348	0.657	0.669
Location	0.117	0.037	0.019	0.013
Cleanliness	0.238	0.017	0.039	0.044
Service	0.300	0.261	0.182	0.172

Result from Table 4 shows that the aspects "service", "cleanliness" are the most important aspects in group of hotels with 5 stars while the aspects " values", "rooms" are the most important aspects in group of hotels with 4 stars, and the aspects "rooms", "service" are the most important aspects in groups of hotels with 3 stars and 2 stars. This information is very interesting and valuable for hotel managers.

5.3 Evaluation

In order to evaluate our method in comparison with previous studies we also conducted the two related methods on the same data. The first one is the Probabilistic Rating Regression presented in[13], we call it as PRR algorithm. The second one is the frequency-based method in[17] which computes the overall aspect weights using aspect frequency. We measure the quality of overall aspect weights through the differences of the predicted ratings with ground-true ratings. All the algorithms are evaluated on the same data set. We implement a 4-fold cross validation and get the average value for evaluation.

In the training phase, we denote α_F is the vector of the overall aspect weights computed by the frequency-based method; denote $\overline{\alpha}_{PRR}$ is the vector of the overall aspect weights and it is computed by PRR algorithm; denote α_{NNAWs} is the vector of the overall aspect weights of reviews and it is learned directly from reviews by the algorithm NNAWs. Note that for all the experimental algorithms, we use the P Rank algorithm in[18] for learning a model for rating aspects (i.e. from 1 star to 5 stars).

In the testing phase, we first determine aspect ratings for each review by using the result of P Rank. We then use the combination of aspect ratings with each different types of overall aspect weights α_F, $\overline{\alpha}_{PRR}$ and α_{NNAWs} to compute the overall rating for each review $d \in D_{test}$ (i.e. denote D_{test} is a set of test data), our goal is to evaluate the quality of each type of overall aspect weights through this combination. According to[14], we compute the overall rating for each review $d \in D_{test}$ by the equation as follows:

$$O_d = \sum_{i=1}^{k} r_{di}\alpha_i \tag{8}$$

Let denote O_{d^*} is the ground-truth rating of review d. $\Delta^2_{Overall}$ stands for the difference between O_{d^*} and O_d, which is defined as follows:

$$\Delta^2_{Overall} = \frac{1}{|D_{test}|}\sum_{d=1}^{|D_{test}|}(O_{d^*} - O_d)^2$$

Table 5 shows experimental results obtained from our method and the two previous works. It first shows that using the overall aspect weights α_{NNAWs} in predicting overall rating we obtained the minimum mean square error, it means the overall aspect weights α_{NNAWs} is the best accuracy. Second, using the overall aspect weights α_F gets the worst result, it indicates the frequency-based

[4] http://sifaka.cs.uiuc.edu/ wang296/Codes/LARA.zip

method only captures the aspect frequency information, and neglects to consider the impact of aspect ratings on the overall ratings. It may provide information overview about important aspects but do not greatly influence consumer's overall ratings. The $\overline{\alpha}_{PRR}$ is computed by PRR algorithm obtain fewer errors than α_F but it is not as good as α_{NNAWs}. This result indicates that the overall aspect weights are not dependent on the aspect weights in each individual review. The results has shown that our model is experimentally better than both the probabilistic rating regression method and the frequency-based method.

Table 5. The experimental results from our methods and some previous methods

Method	$\Delta^2_{Overall}$
α_F + P Rank	0.421
$\overline{\alpha}_{PRR}$ + P Rank	0.403
α_{NNAWs}+ P Rank	0.389

6. Conclusion

In this paper, we have proposed a new model based on neural network using both known aspect ratings and the overall ratings of reviews to determine the overall aspect weights. From the experimental results, we have demonstrated that the overall aspect weights learned directly from numerous consumer reviews by our proposed model is closer to the ground-true ratings than the results computed by the probabilistic rating regression method or the frequency-based method.

7. Acknowledgments

This work is supported by the Nafosted project 102.01-2014.22

8. References

1. Hatzivassiloglou V and Wiebe J. Effects of adjective orientation and gradability on sentence subjectivity, In Proceedings of the International Conference on Computational Linguistics (COLING). 2000. p. 299–305.
2. Riloff E, Patwardhan S and Wiebe J, Feature subsumption for opinion analysis. Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). 2006. p. 440–8.
3. Riloff E, Wiebe J. Learning extraction patterns for subjective expressions. In Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP). 2003. p. 105–12.
4. Wiebe J, Bruce RF, O'Hara TP. Development and use of a gold standard data set for subjectivity classifications. Proceedings of the Association for Computational Linguistics (ACL). 1999; 246–53.
5. Cui H, Mittal V and Datar M. Comparative experiments on sentiment classification for online product reviews. Twenty-First National Conference on Artificial Intelligence. 2006; 21. p. 1265–70.
6. Dave K, Lawrence S, Pennock DM. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. Proceedings of WWW. 2003; 519–28.
7. Devitt A, Ahmad K. Sentiment polarity identification in financial news: Acohesion-based approach. Proceedings of ACL. 2007; 984–91.
8. Kim S, Hovy E. Determining the sentiment of opinions. Proceedings of COLING. 2004; 4:1367–73.
9. Pang B, Lee L, Vaithyanathan S. Thumbs up? Sentiment classification using machine learning techniques. Proceedings of EMNLP. 2002; 79–86.
10. Pang B, Lee L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In ACL. 2005; 115–24.
11. Gao W, Yoshinaga N, Kaji N, Kitsuregawa M. Modeling user leniency and product popularity for sentiment classification. Proceedings of IJCNLP. 2013; 1107–11.
12. Tang D, Qin B, Liu T. User Modeling with Neural Network for Review Rating Prediction. Proceeding of the 24th International Joint Conference on Artificial Intelligence. 2015. p. 1340–6.
13. Wang H, Lu Y, Zhai C. Latent Aspect Rating Analysis on Review Text Data: A Rating Regression Approach. Proceedings of SIGKDD. 2010; 168–76.
14. Wang H, Lu Y, Zhai C. Latent Aspect Rating Analysis without Aspect Keyword Supervision. Proceedings of SIGKDD. 2011; 618–26.
15. Xu Y, Lin T, Lam W. Latent Aspect Mining via Exploring Sparsity and Intrinsic Information. Proceedings of CIKM. 2014; 879–88.
16. Yu J, Zha Z, Wang M, Chua T. Aspect ranking: Identifying important product aspects from online consumer reviews. Proceedings of the Association for Computational Linguistics (ACL). 2011; 1496–505.
17. Zha Z, Yu J, Tang J, Wang M, Chua T. Product Aspect Ranking and Its Applications. IEEE Transactions on Knowledge and Data Engineering. 2014; 26(5):1211–24.
18. Crammer K, Singer Y. Pranking with ranking. Proceedings of NIPS. 2001; 641–7.