Incorporating Guest Preferences into Collaborative Filtering for Hotel Recommendation

Fumiyo Fukumoto[1], Hiroki Sugiyama[2], Yoshimi Suzuki[1] and Suguru Matsuyoshi[1]
[1]*Interdisciplinary Graduate School of Medicine and Engineering, Univ. of Yamanashi, Kofu, Japan*
[2]*Faculty of Engineering, Univ. of Yamanashi, Kofu, Japan*

Keywords: Collaborative Filtering, Latent Dirichlet Allocation, Recommendation System, Transition Probability.

Abstract: Collaborative filtering (CF) has been widely used as a filtering technique because it is not necessary to apply more complicated content analysis. However, it is difficult to take users' preferences/criteria related to the *aspects* of a product/hotel into account. This paper presents a method of hotel recommendation that incorporates different aspects of a product/hotel to improve quality of the score. We used the results of aspect-based sentiment analysis for guest preferences. The empirical evaluation using Rakuten Japanese travel data showed that aspect-based sentiment analysis improves overall performance. Moreover, we found that it is effective for finding hotels that have never been stayed at but share the same neighborhoods.

1 INTRODUCTION

Collaborative filtering (CF) identifies the potential preference of a consumer/guest for a new product/hotel by using only the information collected from other consumers/guests with similar products/hotels in the database. It is a simple technique as it is not necessary to apply more complicated content analysis compared to the content-based filtering framework (Balabanovic and Shoham, 1997). CF has been very successful in both research and practical systems as it has been widely studied (Huang et al., 2004; Yildirim and Krishnamoorthy, 2008; Liu and Yang, 2008; Li et al., 2009; Lathia et al., 2010; Zhao et al., 2013), and many practical systems such as Amazon for book recommendation and Expedia for hotel recommendation have been developed.

Item-based collaborative filtering is one of the major recommendation algorithms (Sarwar et al., 2001; Zhao et al., 2013) because of its simplicity. The algorithms assume that the consumers/guests are likely to prefer product/hotel that are similar to what they have bought/stayed before. Unfortunately, most of them only consider star ratings and leave consumers/guests textual reviews. Several authors focused on the problem, and attempted to improve recommendation results by using the techniques on text analysis such as sentiment analysis, opinion mining, or information extraction (Cane et al., 2006; Niklas et al., 2009; Raghavan et al., 2012). However, major approaches aim at finding the positive/negative opinions for the product/hotel, and do not take users preferences related to the *aspects* of a product/hotel into account. For instance, one guest is interested in a nice restaurant for selecting hotels for her/his vacation, while another guest, *e.g.*, a businessman prefers to the hotel which is close to the station. In this case, the aspect of the former is different from that of the latter.

This paper presents a collaborative filtering method for hotel recommendation incorporating guest preferences. We rank hotels according to scores. The score is obtained by using the analysis of different aspects of guest preferences. The method utilizes a large amount of guest reviews which make it possible to solve the item-based filtering problem of data sparseness, *i.e.*, some items were not assigned a label of users preferences. We used the results of aspect-based sentiment analysis to recommend hotels because whether or not the hotel can be recommended depends on the guest preferences related to the aspects of a hotel. For instance, if one guest stays at hotels for her/his vacation, a room with nice views may be an important factor to select hotels, whereas another guest who stays at hotels for business, hotels with close to the station may be selected.

We parsed all reviews by using syntactic analyzer, and extracted dependency triples which represent the relationship between aspect and its preference. For each aspect of a hotel, we identified the guest opinion, *e.g.*, the aspect, *service* is good or not, based

Fukumoto F., Sugiyama H., Suzuki Y. and Matsuyoshi S..
Incorporating Guest Preferences into Collaborative Filtering for Hotel Recommendation.
DOI: 10.5220/0005034000220030
In *Proceedings of the International Conference on Knowledge Discovery and Information Retrieval* (KDIR-2014), pages 22-30
ISBN: 978-989-758-048-2

on the dependency triples in the guest reviews. The positive/negative opinion on some aspect is used to calculate transitive association between hotels. Finally, we scored hotels by Markov Random Walk (MRW) model, *i.e.*, we used MRW based recommendation technique to explore transitive associations between the hotels. Random Walk based recommendation overcomes the item-based CF problem that the inability to explore transitive associations between the hotels that have never been stayed but share the same neighborhoods (Li et al., 2009).

2 RELATED WORK

CF mainly consists of two procedures, *prediction* and *recommendation* (Sarwar et al., 2001). Here, prediction refers to a numerical value expressing the predicted likeliness of item for user, and recommendation is a list of items that the user will like the most. As the volume of online reviews has drastically increased, sentiment analysis, opinion mining, and information extraction for the process of prediction are a practical problem attracting more and more attention. Several efforts have been made to utilize these techniques to recommend products (Niklas et al., 2009; Faridani, 2011). Cane *et. al* have attempted to elicit user preferences expressed in textual reviews, and map such preferences onto some rating scales that can be understood by existing CF algorithms (Cane et al., 2006). They identified sentiment orientations of opinions by using a relative-frequency-based method that estimates the strength of a word with respect to a certain sentiment class as the relative frequency of its occurrence in the class. The results using movie reviews from the Internet Movie Database (IMDb) for the MovieLens 100k dataset showed the effectiveness of the method, while the sentiment analysis they used is limited, *i.e.*, they used only adjectives or verbs.

Niklas *et. al* have attempted to improve the accuracy of movie recommendations by using the results of opinion extraction from free-text reviews (Niklas et al., 2009). They presented three approaches: (i) manual clustering, (ii) semi-automatic clustering by Explicit Semantic Analysis (ESA), and (iii) fully automatic clustering by Latent Dirichlet Allocation (LDA) (Blei et al., 2003) to extract movie aspects as opinion targets, and used them as features for the collaborative filtering. The results using 100 random users from the IMDb showed that the LDA-based movie aspect extraction yields the best results. Our work is similar to Niklas *et. al* method in the use of LDA. The difference is that our approach applied LDA to the dependency triples while Niklas applied LDA to single words. Raghavan *et. al* have attempted to improve the performance of collaborative filtering in recommender systems by incorporating quality scores to ratings (Raghavan et al., 2012). The quality scores are used to decide the importance given to the individual rating. Their method using quality scores consists of two steps. In the first step, the quality scores of ratings using the review and user data set are estimated. The second step involves rating by using the quality scores as weights. They adapted the probabilistic matrix factorization (PMF) framework. The PMF aims at inferring latent factors of users and items from the available ratings. The experimental evaluation on two product categories of a benchmark data set, *i.e.*, *Book* and *Audio CDs* from Amazon.com showed the efficacy of the method.

In the context of recommendation, several authors have attempted to rank items by using graph-based ranking algorithms (Yin et al., 2010; L.Li et al., 2014). Wijaya *et. al* have attempted to rank items directly from the text of their reviews (Wijaya and Bressan, 2008). They constructed a sentiment graph by using simple contextual relationships such as collocation, negative collocation and coordination by pivot words such as conjunctions and adverbs. They applied PageRank algorithm to the graph to rank items. They reported that the results using 50 movies randomly selected from box office list of Nov. 2007 to Feb. 2008 showed effectiveness of their method while combination of positive and negative orientation did not work well in a linear fashion.

Li *et. al* proposed a basket-sensitive random walk model for personalized recommendation in the grocery shopping domain (Li et al., 2009). The method extends the basic random walk model by calculating the product similarities through a weighted bi-partite network which allows the current shopping behaviors to influence the product ranking. Empirical results using three real-world data sets, LeShop, TaFeng and an anonymous Belgium retailer showed that a performance improvement of the method over other existing collaborative filtering models, the cosine, conditional probability and the bi-partite network based similarities. However, the transition probability from one product node to another product node is computed based on a user's purchase frequency of a product with regardless of the users' positive or negative opinions concerning to the product.

There are three novel aspects in our method. Firstly, we propose a method to incorporate different aspect of a hotel into users preferences/criteria to improve quality of recommendation. Secondly, from a ranking perspective, the MRW model we used is calculated based on the polarities of reviews. Fi-

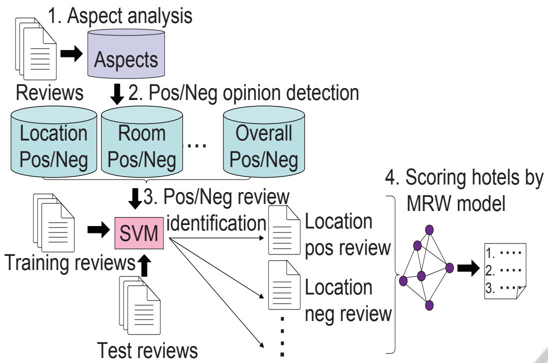

Figure 1: Overview of the method.

nally, from the opinion mining perspective, we propose overcoming with the unknown polarized words by utilizing LDA.

3 SYSTEM DESIGN

Figure 1 illustrates an overview of the method. It consists of four steps: (1) Aspect analysis, (2) Positive/negative opinion detection based on aspect analysis, (3) Positive/negative review identification, and (4) Scoring hotels by MRW model.

3.1 Aspect Analysis

The first step to recommend hotels based on guest preferences is to extract aspects for each hotel from a guest review corpus. All reviews were parsed by the syntactic analyzer CaboCha (Kudo and Matsumoto, 2003), and all the dependency triples (*rel*, *x*, *y*) are extracted. Here, *x* refers to a noun/compound noun word related to the aspect. *y* shows verb or adjective word related to the preference for the aspect. *rel* denotes a grammatical relationship between *x* and *y*. We classified *rel* into 9 types of Japanese particle, "*ga*(*ha*)", "*wo*", "*ni*", "*he*", "*to*", "*de*", "*yori*", "*kara*" and "*made*". For instance, from the sentence "*Cyousyoku (breakfast) ga totemo (very) yokatta (good).*" (The breakfast was very good.), we can obtain the dependency triplet, (*ga*, *cyousyoku*, *yokatta*). The triplet represents positive opinion, "*yokatta*"(*good*) concerning to the aspect, "*Cyousyoku*"(*meal*).

3.2 Positive/Negative Opinion Detection

We identified positive/negative opinion based on the aspects of a hotel. We classified aspects into seven types: "Location", "Room", "Meal", "Spa", "Service", "Amenity", and "Overall". These types are defined by Rakuten travel[1]. For each aspect, we identified positive/negative opinion by using Japanese sentiment polarity dictionary (Kobayashi et al., 2005), *i.e.*, we regarded the extracted dependency triplet as positive/negative opinion if *y* in the triplet (*rel*, *x*, *y*) is classified into positive/negative classes in the dictionary. However, the dictionary makes it nearly impossible to cover all of the words in the review corpus.

For unknown verb or adjective words that were extracted from the review corpus, but did not appear in any of the dictionary classes, we classified them into positive or negative class by using a topic model. Topic models such as probabilistic latent semantic indexing (Hofmann, 1999) and Latent Dirichlet Allocation (LDA) (Blei et al., 2003) are based on the idea that documents are mixtures of topics, where each topic is captured by a distribution over words. The topic probabilities provide an explicit low-dimensional representation of a document. They have been successfully used in many domains such as text modeling and collaborative filtering (Li et al., 2013). We used LDA and classified unknown words into positive/negative classes. LDA presented by (Blei et al., 2003) models each document as a mixture of topics, and generates a discrete probability distribution over words for each topic. The generative process for LDA can be described as follows:

1. For each topic $k = 1, \cdots, K$, generate ϕ_k, multinomial distribution of words specific to the topic k from a Dirichlet distribution with parameter β;
2. For each document $d = 1, \cdots, D$, generate θ_d, multinomial distribution of topics specific to the document d from a Dirichlet distribution with parameter α;
3. For each word $n = 1, \cdots, N_d$ in document d;
   (a) Generate a topic z_{dn} of the n^{th} word in the document d from the multinomial distribution θ_d
   (b) Generate a word w_{dn}, the word associated with the n^{th} word in document d from multinomial ϕ_{zdn}

Like much previous work on LDA, we used Gibbs sampling to estimate ϕ and θ. The sampling probability for topic z_i in document d is given by:

$$P(z_i \mid z_{\setminus i}, W) = \frac{(n^{v}_{\setminus i,j} + \beta)(n^{d}_{\setminus i,j} + \alpha)}{(n^{\cdot}_{\setminus i,j} + W\beta)(n^{d}_{\setminus i,\cdot} + T\alpha)}. \quad (1)$$

$z_{\setminus i}$ refers to a topic set Z, not including the current assignment z_i. $n^{v}_{\setminus i,j}$ is the count of word v in topic j that does not include the current assignment z_i, and

[1] http://rit.rakuten.co.jp/rdr/index.html

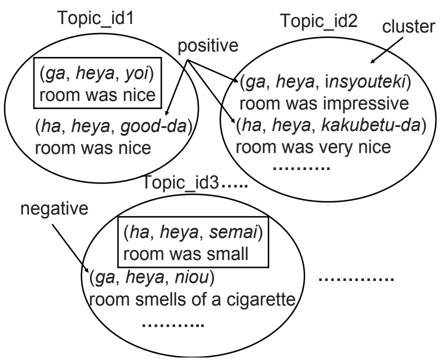

Figure 2: Clusters obtained by LDA.

$n^{\cdot}_{\setminus i,j}$ indicates a summation over that dimension. W refers to a set of documents, and T denotes the total number of unique topics. After a sufficient number of sampling iterations, the approximated posterior can be used to estimate ϕ and θ by examining the counts of word assignments to topics and topic occurrences in documents. The approximated probability of topic k in the document d, $\hat{\theta}_d^k$, and the assignments word w to topic k, $\hat{\phi}_k^w$ are given by:

$$\hat{\theta}_d^k = \frac{N_{dk} + \alpha}{N_d + \alpha K}. \qquad (2)$$

$$\hat{\phi}_k^w = \frac{N_{kw} + \beta}{N_k + \beta V}. \qquad (3)$$

For each aspect, we manually collected reviews and created a review set. We applied LDA to each set of reviews consisted of triples. We need to estimate the number of topics k for the result obtained by LDA. Figure 2 illustrates the result obtained by LDA. The aspect type is "room". The triplet marked with box includes unknown adjective words. We can see from Figure 2 that the result can be regarded as a clustering result: each element of the clusters is positive/negative opinion according to the sentiment polarity dictionary, or unknown words. We estimated the number of topics (clusters) k by using Entropy measure given by:

$$E = -\frac{1}{\log k} \sum_j \frac{N_j}{N} \sum_i P(A_i, C_j) \log P(A_i, C_j). \qquad (4)$$

k refers to the number of clusters. $P(A_i, C_j)$ is a probability that the elements of the cluster C_j assigned to the correct class A_i. N denotes the total number of elements and N_j shows the total number of elements assigned to the cluster C_j. The value of E ranges from 0 to 1, and the smaller value of E indicates better result. We chose the parameter k whose value of E is smallest. For each cluster, if the number of positive opinion is larger than those of negative ones, we regarded a triplet including unknown word in the cluster as positive and vice versa. For example, "*yoi*"(nice) in the Topic_id1 cluster shown in Figure 2 is regarded to a positive as the number of positive and negative were one and zero, respectively.

3.3 Positive/Negative Review Identification

We used the result of positive/negative opinion detection to classify guest reviews into positive or negative related to the aspect. Like much previous work on sentiment analysis based on supervised machine learning techniques (Turney, 2002) or corpus-based statistics, we used Support Vector Machine (SVMs) to annotate automatically (Joachims, 1998). For each aspect, we collected positive/negative opinion (triples) from the results of LDA[2]. Each review in the test data is represented as a vector where each dimension of a vector is positive/negative triplet appeared in the review, and the value of each dimension is a frequency count of the triplet. For each aspect, the classification of each review can be regarded as a two-class problem: positive or negative.

3.4 Scoring Hotels by MRW Model

The final procedure for recommendation is to rank each hotel. We used a ranking algorithm, the MRW model that has been successfully used in Web-link analysis, social networks (Xue et al., 2005), and recommendation (Li et al., 2009; Yin et al., 2010; L.Li et al., 2014). We applied the algorithm to rank hotels. Given a set of hotels H, Gr = (H, E) is a graph reflecting the relationships between hotels in the set. H is the set of nodes, and each node h_i in H refers to the hotel. E is a set of edges, which is a subset of $H \times H$. Each edge e_{ij} in E is associated with an affinity weight $f(i \rightarrow j)$ between hotels h_i and h_j ($i \neq j$). The weight of each edge is a value of transition probability $P(h_j \mid h_i)$ between h_i and h_j, and defined by:

$$P(h_j \mid h_i) = \sum_{k=1}^{|Gr|} \frac{c(g_k, h_j)}{(\sum c(g_k, \cdot))} \cdot \frac{c(g_k, h_i)}{(\sum c(\cdot, h_i))}. \qquad (5)$$

Eq. (5) shows the preference voting for target hotel h_j from all the guests in Gr who stayed at h_i. We note that we classified reviews into positive/negative. We used the results to improve the quality of score. More

[2]We used the clusters that the number of positive and negative words is not equal.

precisely, we used positive review counts to calculate transition probability, *i.e.*, $c(g_k, h_j)$ and $c(g_k, h_i)$ in Eq. (5) refer to the lodging count that the guest g_k reviewed the hotel $h_j(h_i)$ as *positive*. $P(h_j \mid h_i)$ in Eq. (5) is the marginal probability distribution over all the guests. The transition probability obtained by Eq. (5) shows a weight assigned to the edge between hotels h_i and h_j.

We used the row-normalized matrix $U_{ij} = (U_{ij})_{|H|\times|H|}$ to describe Gr with each entry corresponding to the transition probability, where $U_{ij} = p(h_j \mid h_i)$. To make U a stochastic matrix, the rows with all zero elements are replaced by a smoothing vector with all elements set to $\frac{1}{|H|}$. The matrix form of the recommendation score $Score(h_i)$ can be formulated in a recursive form as in the MRW model: $\vec{\lambda} = \mu U^T \vec{\lambda} + \frac{(1-\mu)}{|H|}\vec{e}$, where $\vec{\lambda} = [Score(h_i)]_{|H|\times 1}$ is a vector of saliency scores for the hotels. $\vec{e}$ is a column vector with all elements equal to 1. μ is a damping factor. We set μ to 0.85, as in the PageRank (Brin and Page, 1998). The final transition matrix is given by:

$$M \quad = \quad \mu U^T + \frac{(1-\mu)}{\mid H \mid}\vec{e}\vec{e}^T. \qquad (6)$$

Each score is obtained by the principal eigenvector of the new transition matrix M. We applied the algorithm to the graph. The higher score based on transition probability the hotel has, the more suitable the hotel is recommended. For each aspect, we chose the topmost k hotels according to rank score. For each selected hotel, if the negative review is not included in the hotel reviews, we regarded the hotel as a recommendation hotel.

4 EXPERIMENTS

4.1 Data

We used Rakuten travel data[3]. It consists of 11,468 hotels, 348,564 reviews submitted from 157,729 guests. We used plda[4] to assign positive/negative tag to the aspects. For each aspect, we estimated the number of topics (clusters) by searching in steps of 100 from 200 to 1,000. Table 1 shows the minimum entropy value and the number of topics for each aspect. As shown in Table 1, the number of topics ranges from 500 to 700. For each of the seven aspects, we used these number of topics in the experiments. We used linear kernel of SVM-Light (Joachims, 1998) and set all parameters to their default values. All reviews were parsed by the syntactic analyzer CaboCha (Kudo and Matsumoto, 2003), and 633,634 dependency triples are extracted. We used them in the experiments.

Table 1: The minimum entropy value and the # of topics.

Aspect	Entropy	Topics
Location	0.209	700
Room	0.460	600
Meal	0.194	700
Spa	0.232	500
Service	0.226	700
Amenity	0.413	600
Overall	0.202	700

We had an experiment to classify reviews into positive or negative. For each aspect, we chose the topmost 300 hotels whose number of reviews are large. We manually annotated these reviews. The evaluation is made by two humans. The classification is determined to be correct if two human judges agree. We obtained 400 reviews consisting 200 positive and 200 negative reviews. 400 reviews are trained by using SVMs for each aspect, and classifiers are obtained. We randomly selected another 100 test reviews from the topmost 300 hotels, and used them as test data. Each of the test data was classified into positive or negative by SVMs classifiers. The process is repeated five times. As a result, the macro-averaged F-score concerning to positive across seven aspects was 0.922, and the F-score for negative was 0.720. For each aspect, we added the reviews classified by SVMs to the original 400 training reviews, and used them as a training data to classify test reviews.

We created the data which is used to test our recommendation method. More precisely, we used the topmost 100 guests staying at a large number of different hotels as recommendation. For each of the 100 guests, we sorted hotels in chronological order. We used these with the latest five hotels as test data. To score hotels by MRW model, we used guest data staying at more than three times. The data is shown in Table 2. “Hotels” and “Different hotels” in Table 2 refer to the total number of hotels, and the number of different hotels that the guests stayed at more than three times, respectively. “Guests” shows the total number of guests who stayed at one of the “Different hotels”. “Reviews” shows the number of reviews with these hotels.

For evaluation measure used in recommendation, we used MAP (Mean-Averaged Precision) (Yates and Neto, 1999). For a given set of guests $G = \{g_1, \cdots, g_n\}$, and $H = \{h_1, \cdots, h_{m_j}\}$ be a set of hotels that

[3]http://rit.rakuten.co.jp/rdr/index.html

[4]http://code.google.com/p/plda

Table 2: Data used in the experiments.

Hotels	30,358
Different hotels	6,387
Guests	23,042
Reviews	116,033

Table 3: Recommendation results.

Method	MAP
Trans. pro. without review	0.257
Content Words	0.304
Without reviews by SVMs	0.356
Without neg review filtering	0.378
Aspect-based SA	0.392

should be recommended for a guest g_j, the *MAP* of G is given by:

$$\mathrm{MAP}(G) = \frac{1}{|G|}\sum_{j=1}^{|G|}\frac{1}{m_j}\sum_{k=1}^{m_j} Precision(R_{jk}). (7)$$

R_{jk} in Eq. (7) refers to the set of ranked retrieval results from the top result until we get hotel h_k. *Precision* indicates a ratio of correct recommendation hotels by the system divided by the total number of recommendation hotels.

4.2 Recommendation Results

We compared the results obtained by our method, aspect-based sentiment analysis (ASA) with the following four approaches to examine how the results of each method affect the overall performance.

1. Transition probabilities without review (TPWoR)
   The probability $P(h_j \mid h_i)$ used in the method is the preference voting for the target hotel h_j from all the guests in a set G who stayed at h_i, regardless of positive or negative review of G.
2. Content Words (CW)
   The difference between content words method and our method, ASA is that the former applies LDA to the content words.
3. Without reviews classified by SVMs (WoR)
   SVMs used in this method classifies test data by using only the original 400 training reviews.
4. Without negative review filtering (WoNRF)
   The method selected the topmost k hotels according to the MRW model, and the method dose not use negative reviews as a filtering.

Table 3 shows averaged MAP across seven aspects. As we can see from Table 3 that aspect-based sentiment analysis was the best among four baselines, and

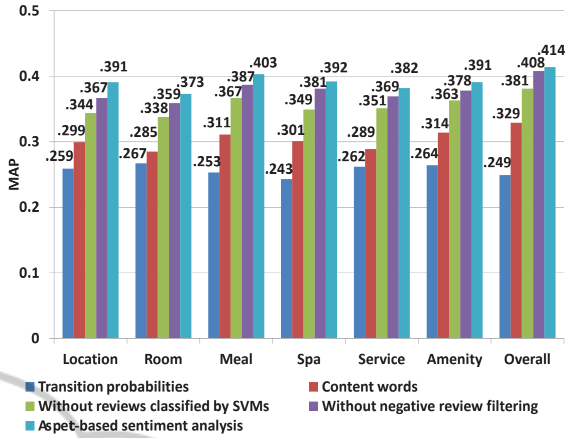

Figure 3: The results against each aspect.

MAP score attained at 0.392. The result obtained by transition probability without review was worse than any other results. This shows that the use of guest review information is effective for recommendation. Table 3 shows that the result obtained by content words method was worse than the result obtained by aspect-based sentiment analysis, and even worse than the results without reviews classified by SVMs (WoR) and without negative review filtering (WoNRF). Furthermore, we can see from Table 3 that negative review filtering was a small contribution, *i.e.*, the improvement was 0.014 as the result without negative review filtering was 0.378 and aspect-based SA was 0.392. One reason is that the accuracy of negative review identification. The macro-averaged F-score concerning to negative across seven aspects was 0.720, while the F-score for positive was 0.922. Negative review filtering depends on the performance of negative review identification. Therefore, it will be necessary to examine features other than word triples to improve negative review identification.

Table 4 shows sample clusters regarded as positive for three aspects, “location”, “room”, and “meal” obtained by LDA. Each cluster shows the top 5 triples and content words. We observed that the extracted triples show positive opinion for each aspect. This indicates that aspect extraction contributes to improve overall performance. In contrast, some words such as *yoi* (be good) and *manzoku* (satisfy) in content word based clusters appear across aspects. Similarly, some words such as *ricchi* (location) and *cyousyoku* (breakfast) which appeared in negative cluster are an obstacle to identify positive/negative reviews in SVMs classification.

It is very important to compare the results of our method with four baselines against each aspect. Figure 3 shows MAP against each aspect. The results obtained by aspect-based sentiment analysis were statis-

Table 4: Top 5 triples and content words.

Aspects			Content Words		
Location	Room	Meal	Location	Room	Meal
(*ni*, *eki*, *chikai*) be near to the station	(*ga*, *heya*, *yoi*) room was nice	(*ga*, *shokuji*, *yoi*) breakfast was nice	*ricchi* location	*heya* room	*syokuji* meal
(*ha*, *hotel*, *chikai*) the hotel is close	(*ha*, *heya*, *hiroi*) the room is wide	(*ha*, *shokuji*, *yoi*) meal was nice	*eki* station	*hiroi* be wide	**yoi** be good
(*ni*, *hotel*, *chikai*) be near to the hotel	(*ga*, *heya*, *kirei*) A room is clean	(*ha*, *restaurant*, *good*) a restaurant is good	**yoi** be good	*kirei* be clean	*cyousyoku* breakfast
(*ni*, *parking*, *chikai*) be near to the parking	(*de* , *sugoseru*, *heya*) can spend in the room	(*ha*, *restaurant*, *yoi*) restaurant is nice	*mise* store	**manzoku** satisfy	*oishii* be delicious
(*ga*, *konbini*, *aru*) be near to the convenience store	(*ha*, *heya*, *jyuubun*) a room is enough good	(*ha*, *buffet*, *yoi*) Buffet is delicious	*subarashii* *be great*	**yoi** be good	**manzoku** satisfy

Table 5: Recommendation list for user ID 2037.

R	TPWoR	CW	WoR	WoNRF	ASA
1	2349	2203	2614	3022	**449**
2	604	2349	554	604	30142
3	12869	30142	30142	**449**	**18848**
4	90	604	604	30142	531
5	666	12869	3022	**18848**	**769**
6	2149	39502	531	531	**2223**
7	38126	**449**	**449**	**769**	15204
8	**449**	31209	**18848**	**2223**	**20428**

tically significant compared to other methods except for the aspects "spa" and "overall in "without negative review filtering" method.

Table 5 shows a ranked list of the hotels for one guest (guest ID: 2037) obtained by using each method. The aspect is "meal", and each number shows hotel ID. Bold font in Table 5 refers to the correct hotel, *i.e.*, the latest five hotels that the guest stayed at. As can be seen clearly from Table 5, the result obtained by our method includes all of the five correct hotels within the topmost eight hotels, while without negative review filtering (WoNRF) was four. TPWoR, CW, and WoR did not work well as the number of correct hotel was no more than three, and these were ranked seventh and eighth.

It is interesting to note that some recommended hotels are very similar to the correct hotels, while most of the eight hotels did not exactly match these correct hotels except for the result obtained by aspect-based sentiment analysis method. If these hotels were similar to the correct hotels, the method is effective for finding transitive associations between the hotels that have never been stayed but share the same neighborhoods. Therefore, we examined how these hotels are similar to the correct hotels. To this end, we calculated distance between correct hotels and other hotels within the rank for each method by using seven preferences. The preferences have star rating, *i.e.*, each has been scored from 1 to 5, where 1(bad) is lowest, and 5(good) is the best score. We represented each ranked hotel as a vector where each dimension of a vector is these seven preferences and the value of each dimension is its score value. The distance between correct hotel and other hotels within the rank for each method X is defined as:

Table 6: Distance between correct hotel and another hotel.

Method	Dis
Trans. pro. without review	3.067
Content Words	2.859
Without reviews by SVMs	2.721
Without neg review filtering	2.532
Aspect-based SA	2.396

$$\mathrm{Dis}(X) = \frac{1}{\mid G \mid} \sum_{i=1}^{|G|} \operatorname*{argmin}_{j,k} d(R\_h_{ij}, C\_h_{ik}). \qquad (8)$$

$\mid G \mid$ refers to the number of guests. R_h_{ij} refers to a vector of the j-th ranked hotels except for the correct hotels. Similarly, C_h_{ik} stands for a vector representation of the k-th correct hotel. d refers to Euclidean distance. Eq. (8) shows that for each guest, we obtained the minimum value of Euclidean distance between R_h_{ij} and C_h_{ik}. We calculated the averaged summation of the 100 guests. The results are shown in Table 6.

The value of "Dis" in Table 6 shows that the smaller value indicates a better result. We can see from Table 6 that the hotels except for the correct hotels obtained by our method are more similar to the correct hotels than those obtained by four baselines. The results show that our method is effective for finding hotels that have never been stayed at but share the same neighborhoods.

5 CONCLUSIONS

We proposed a method for recommending hotels by incorporating different aspects of a hotel to improve quality of score. We used the results of aspect-based sentiment analysis for guest preferences. We parsed all reviews by the syntactic analyzer, and extracted dependency triples. For each aspect, we identified the guest opinion to positive or negative, by using dependency triples in the guest review. We calculated transitive association between hotels based on the positive/negative opinion. Finally, we scored hotels by Markov Random Walk model. The comparative results using Rakuten travel data showed that aspect analysis of guest preferences improves overall performance, and it is effective for finding hotels that have never been stayed at but share the same neighborhoods.

There are a number of directions for future work. In the aspect-based sentiment analysis for guest preferences, we should be able to obtain further advantages in efficacy by overcoming the lack of sufficient reviews in data sets by incorporating transfer learning approaches (Blitzer et al., 2007; Dai et al., 2007). We used Rakuten Japanese travel data in the experiments, while the method is applicable to other textual reviews. To evaluate the robustness of the method, experimental evaluation by using other data such as grocery stores: LeShop[5] and movie data: movieLens[6] can be explored in future. Finally, comparison to other recommendation methods, *e.g.*, matrix factorization methods (MF) (Koren et al., 2009) and combination of MF and the topic modeling (Wang and Blei, 2011) will also be considered in the future.

ACKNOWLEDGEMENTS

The authors would like to thank the referees for their valuable comments on the earlier version of this paper. We also thank Rakuten Institute of Technology for providing Japanese travel data. This work was supposed by the Grant-in-aid for the Japan Society for the Promotion of Science (No. 25330255).

REFERENCES

Balabanovic, M. and Shoham, Y. (1997). Fab Content-based Collaborative Recommendation. *In Communications of the ACM*, 40:66–72.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003). Latent Dirichlet Allocation. *Machine Learning*, 3:993–1022.

Blitzer, J., Dredze, M., and Pereira, F. (2007). Biographies, Bollywood, Boom-boxes and Blenders: Domain Adaptation for Sentiment Classification. In *Proc. of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 187–295.

Brin, S. and Page, L. (1998). The Anatomy of a Large-scale Hypertextual Web Search Engine. *Computer Networks*, 30(1-7):107–117.

Cane, W. L., Stephen, C. C., and Fu-lai, C. (2006). Integrating Collaborative Filtering and Sentiment Analysis. In *Proc. of the ECAI 2006 Workshop on Recommender Systems*, pages 62–66.

Dai, W., Yang, Q., Xue, G., and Yu, Y. (2007). Boosting for Transfer Learning. In *Proc. of the 24th International Conference on Machine Learning*, pages 193–200.

Faridani, S. (2011). Using Canonical Correlation Analysis for Generalized Sentiment Analysis Product Recommendation and Search. In *Proc. of 5th ACM Conference on Recommender Systems*, pages 23–27.

Hofmann, T. (1999). Probabilistic Latent Semantic Indexing. In *Proc. of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 50–57.

Huang, Z., Chen, H., and Zeng, D. (2004). Applying Associative Retrieval Techniques to Alleviate the Sparsity Problem in Collaborative Filtering. *ACM Transactions on Information Systems*, 22(1):116–142.

Joachims, T. (1998). SVM Light Support Vector Machine. In *Dept. of Computer Science Cornell University*.

Kobayashi, N., Inui, K., Matsumoto, Y., Tateishi, K., and Fukushima, S. (2005). Collecting Evaluative Expressions for Opinion Extraction. *Journal of Natural Language Processing*, 12(3):203–222.

Koren, Y., Bell, R. M., and Volinsky, C. (2009). Matrix Factorization Techniques for Recommender Systems. *In IEEE Computer*, 42(8):30–37.

Kudo, T. and Matsumoto, Y. (2003). Fast Method for Kernel-based Text Analysis. In *Proc. of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 24–31.

Lathia, N., Hailes, S., Capra, L., and Amatriain, X. (2010). Temporal Diversity in Recommender Systems. In *Proc. of the 33rd ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 210–217.

Li, M., Dias, B., and Jarman, I. (2009). Grocery Shopping Recommendations based on Basket-Sensitive Random Walk. In *Proc. of the 15th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 1215–1223.

Li, Y., Yang, M., and Zhang, Z. (2013). Scientific Articles Recommendation. In *Proc. of the ACM International Conference on Information and Knowledge Management CIKM 2013*, pages 1147–1156.

Liu, N. N. and Yang, Q. (2008). A Ranking-Oriented Approach to Collaborative Filtering. In *Proc. of the 31st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 83–90.

[5] www.beshop.ch

[6] http://www.grouplens.org/node/73

L.Li, Zheng, L., Fan, Y., and Li, T. (2014). Modeling and Broadening Temporal User Interest in Personalized News Recommendation. *In Expert system with Applications*, 41(7):3163–3177.

Niklas, J., Stefan, H. W., c. M. Mark, and Iryna, G. (2009). Beyond the Stars: Exploiting Free-Text User Reviews to Improve the Accuracy of Movie Recommendations. In *Proc. of the 1st International CIKM workshop on Topic-Sentiment Analysis for Mass Opinion*, pages 57–64.

Raghavan, S., Gunasekar, S., and Ghosh, J. (2012). Review Quality Aware Collaborative Filtering. In *Pro.c of the 6th ACM Conference on Recommender Systems*, pages 123–130.

Sarwar, B., Karypis, G., Konstan, J., and Reidl, J. (2001). Automatic Multimedia Cross-Model Correlation Discovery. In *Proc. of the 10th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 653–658.

Turney, P. D. (2002). Thumbs Up or Thumbs Down? Semantic Orientation Applied to Un-supervised Classification of Reviews. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 417–424.

Wang, C. and Blei, D. M. (2011). Collaborative Topic Modeling for Recommending Scientific Articles. In *Proc. of the 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 448–456.

Wijaya, D. T. and Bressan, S. (2008). A Random Walk on the Red Carpet: Rating Movies with User Reviews and PageRank. In *Proc. of the ACM International Conference on Information and Knowledge Management CIKM 2008*, pages 951–960.

Xue, G. R., Yang, Q., Zeng, H. J., Yu, Y., and Chen, Z. (2005). Exploiting the Hierarchical Structure for Link Analysis. In *Proc. of the 28th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 186–193.

Yates, B. and Neto, R. (1999). *Modern Information Retrieval*. Addison Wesley.

Yildirim, H. and Krishnamoorthy, M. S. (2008). A Random Walk Method for Alleviating the Sparsity Problem in Collaborative Filtering. In *Proc. of the 3rd ACM Conference on Recommender Systems*, pages 131–138.

Yin, Z., Gupta, M., Weninger, T., and Han, J. (2010). A Unified Framework for Link Recommendation Using Random Walks. In *Proc. of the Advances in Social Networks Analysis and Mining*, pages 152–159.

Zhao, X., Zhang, W., and Wang, J. (2013). Interactive Collaborative Filtering. In *Proc. of the 22nd ACM Conference on Information and Knowledge Management*, pages 1411–1420.