Personalized Recommendation of Coupon Deals by Keywords Association Rules

Yi Huang
Research & Innovation, SAP Asia Pte Ltd, Singapore
Email: yi.huang01@sap.com

***Abstract*—In this paper we focus on personalized recommendation algorithm for coupon deals, which face "cold start" problem at all times because they only have short time validity. As our dataset is from a private source, we first analyzed deal characteristics and found that deals under category "dining", "wellness" and "activities" have a high probability of having the same keywords in the deal names, which suggests a repeated buying pattern. Then we computed the keyword associations from the dataset and found meaningful patterns. Based on the keyword association rules, we proposed a new recommendation algorithm which combines baseline algorithm and keyword association, resulting in significant improvement in percentage of hits, average rank and mean reciprocal rank.**

***Index Terms*—association rules, coupon deals, keywords, personalized recommendation system, repeated buying pattern**

I. Introduction

The boom of E-commerce platforms has taken place in the new century and nowadays more and more transactions are done online. According to [1], in the year 2012 approximately 4% of all Canadian transactions were made online, while 11% for US transactions. Tremendous insights can be drawn from billions of online transactions like ever before and a lot of techniques are developed to improve consumers' shop-ping experiences. One of the techniques is personalized deal recommendation based on individual customers' preferences. Currently most retailers don't differentiate customers when they conduct market campaign, e.g., sending brochures or messages of all sale items to all the customers. This kind of marketing usually ends up spamming customers as most of them may not be interested. Even though some of the customers are interested, the information they focus on may not be very easily spotted among lots of other promotion information. Personalized recommendation is the best way of solving this problem by only recommending relevant products to targeted customers, saving cost for the retailer while helping them acquire more customers. A personalized recommendation engine needs to take into account individual customers' preference and purchase history and find the most relevant products.

Personalized recommendation can be used by all kinds of retailers; however, different business logic will require different types of recommendation algorithms. In this paper we discuss a specific type of deals, coupon deals, which have some particular characteristics. One example of this business model is Groupon, in which all deals are only valid for a period of 1-2 weeks. So we have to deal with the "cold start" problem [2] for all our recommendations. In this paper we discuss the role of keywords playing in recommendation for Groupon-like deals and experiments show that our proposed method outperforms the existing baseline method in terms of both percentage of hits, average rank and mean reciprocal rank.

II. Related Works

A lot of research has been done on recommendation systems for different applications, such as friend recommendation [3], location based deal recommendation [4], movie recommendation [5], music recommendation [6], product recommendation [7] and app recommendation [2]. In most cases, rating information is used for recommendation, e.g., movie, music and Yelp deal [7]. In our case, transactions for coupon deals usually don't have any score or rating. So we can only regard each buying behavior as a "positive" action and there is no "negative" input from users.

In traditional recommendation applications, the items to be recommended are always available, e.g., movies, music and products, however, coupon deals are only valid for a short time. Hence, when recommending coupon deals, referencing deals from history data from other people [7] is not so useful in the case. Collaborative filtering [8] is the most commonly used algorithm for recommendation of general items. It computes the item-to-item or user-to-item similarities and recommends items using combined similarity values. However, we always want to recommend new coupon deals, and collaborative filtering only works well when there are already some transactions for all the deals. According to the formulation of collaborative filtering, only deals with transactions will be recommended and new deals with no transaction will never get recommended.

III. Database Discription and Related Works

In this paper we look into a collection of real-life trans-action data from an online coupon deal marketplace. As

Manuscript received March 16, 2015; revised November 24, 2015.

doi: 10.18178/jiii.4.2.186-190

this data source is not publicly available, so we can only show the statistical information about the dataset here. The description of the dataset is as follows:

1) No. of transactions: 150000+
2) No. of deals: 2000+
3) No. of users who have transactions: 76000+
4) No. of users who have at least 3 transactions: 12000+

Deal categories: dining, wellness, products, activities, travel. These categories are manually defined. Fig. 1 shows the category distribution in transactions.

TABLE I. SAMPLE DEALS OF FIVE CATEGORIES

Category	Deal Name
Dining	$5.50 for Dlifrance All Time Favourite Classic Sandwich & Beverage Set (Worth up to $10.40) at 27 Locations
Dining	9-Course Chinese Cuisine Set: $138 for 6 Pax at Owen Seafood Restaurant at The Grandstand in Turf City (Worth $298)
Wellness	$24 for Classic Manicure and Pedicure with Foot Soak + Foot Scrub at The Professionails in Orchard (Worth $60)
Wellness	3 Hour Spa Indulgence: $40 for Body Massage + Diamond Peel Facial + More at Z Beauty Boutique in Clementi (Worth $380)
Products	One 1TB Portable 2.5" USB HDD Set at $87.90 (Worth $154). 4 Colours Available.
Products	Eurobed Cambridge Pocketed Mattress, with Delivery by Four Star Industries at $199 (Worth $399). More Options Available.
Activities	Sentosa: $29 for Admission to Underwater World + Dolphin Lagoon + Merlion Tower + Songs of the Sea (Worth $47.90)
Activities	$25 for Four 60 min Yoga Classes at Real Yoga in 4 Locations (Worth $180)
Travel	Phuket: $310 per pax for 3D2N 4-Star U Sunsuri Phuket Stay + Tiger Airways Flight + Airport Transfer (Worth $550)
Travel	Ho Chi Minh: $189 per pax for 3D2N Hotel Stay with Breakfast + Vietnam Airlines Flight + Airport Transfer (Worth $355)

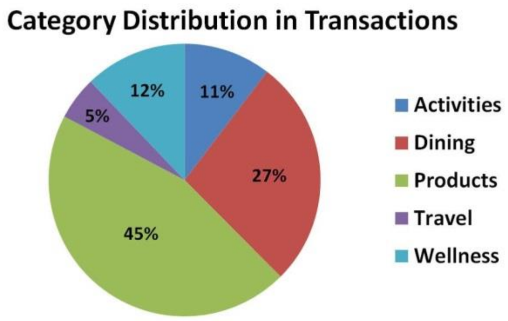

Figure 1. Category distribution in 150000+ transactions. We can see products has the highest share of 45%. Dining, wellness and activities comprises 50% altogether

To give an idea of how the deals look like, Table. I lists some similar examples from Groupon for each category. The detailed description of the dataset can be found in [9]. In [9] the authors also conducted research on if people are buying similar deals repeatedly, e.g., buying facial/body massage deals again and again. Note that when we try to recommend deals to users, we always face the "cold start" problem [10], in which we don't have enough transaction history for new deals every day. Hence [9] proposed to use keywords to link the new deals with old deals and perform recommendation based on the keyword mapping between deals. In this paper, we follow [9] and propose the new recommendation algorithm based on keywords association rules.

IV. PROPOSED SOLUTION

A. Association Rules between Keywords

Traditional Apriori algorithm looks at association rules between deals, e.g., beer and dippers are usually purchased together [11]. However in our context, as deals expire very quickly, our algorithm cannot recommend expired deals based on association rules. So instead of computing the relationship between deals, we compute the relationship between keywords that are in the description of deals which are bought together. We preprocess the deal descriptions and only keep the nouns as described in [9].

Define $U = \{u_1, \dots u_l, \dots, u_L\}(1 \le l \le L)$ as the set of users, with L as the total number of users. $D = \{d_1, \dots d_n, \dots, d_N\}(1 \le n \le N)$ is the set of all deals and N is the total number of deals. Also define $W_n = \{\omega_1^n, \dots \omega_2^n, \dots \omega_{\theta(n)}^n\}$ as the set of words in the description of deal d_n and $\theta(n)$ is the number of words for deal d_n. We first compute all word associations from all transactions using Algorithm 1.

Algorithm 1 Compute the keyword association rules

1: Select $D_l = \{d_1^l, \dots d_2^l, \dots, d_{\delta(l)}^l\}$ $(1 \le l \le L)$ as the list of historical deals that user u_l has purchsed until now. $\delta(l)$ is the number deals bought buy user u_l. Note the list is sorted by date ascendently, which means $d_{\delta(l)}^l$ is the most recent purchase and d_1^l is the oldest purchase.

2: Compute the set of all possible deal pairs $p_{(s,t)}^l = \{d_s^l, d_t^l\}$ in D_l where$1 \le s < t \le \delta(l)$, bought by user u_l. Note that in every deal pair the first deal d_s^l is always bought before the second deal d_t^l by ensuring $s < t$.

3: For every deal pair $p_{(s,t)}^l$, compute all possible word pairs $p_{(k,m)}^{(s,t)} = \{\omega_k^s, \omega_m^t\}$ where ω_k^s belongs to deal d_s^l and ω_m^t belongs to deal d_t^l. Note that as different deals may contain the same words, ω_k^s and ω_m^t can be the same.

4: Aggregate all word pairs into matrix M with every row being each word pair. So M is a $R \times 2$ matrix where $R > 3{,}000{,}000$ in our experiments.

5: We select distinct word pairs in M as $\overline{M}$ and count word pair's occurrence as o_v, representing the v-th row in the matrix M.

6: We also denote $\overline{M}_{v,1}$ and $\overline{M}_{v,2}$ as the first and second words in the v-th row. Then we define the count of $\overline{M}_{v,1}$ in the whole matrix as $c_{(v,1)}$. Note here we count the occurrence in the first column of M instead of $\overline{M}$.

7: Compute the support value for word $\overline{M}_{v,1}$ as

$$supp = \frac{c_{(v,1)}}{R} \tag{1}$$

8: Compute the confidence value for word $\overline{M}_{v,2}$ given the word $\overline{M}_{v,1}$ as

$$conf = \frac{o_v}{c_{(v,1)}} \tag{2}$$

9: Select the top 5% word pairs in $\overline{M}$ with highest values in terms of $supp \times conf$. This results in 160,000+ word pairs.

TABLE II. EXAMPLES OF MOST FREQUENT WORD PAIRS

WORD1	WORD2	SUPPORT	CONFIDENCE
buffet	buffet	0.001097	0.041530
dinner	buffet	0.000273	0.043138
hotel	buffet	0.000114	0.038511
massage	buffet	0.000137	0.020783
massage	massage	0.000124	0.018894
body	massage	0.000103	0.019071
swarovski	swarovski	0.000052	0.018473
treatment	body	0.000060	0.016164
playground	playtime	0.000035	0.024618
beauty	massage	0.000044	0.019258
spa	body	0.000053	0.014788
facial	massage	0.000026	0.019929
necklace	swarovski	0.000022	0.019577
coach	coach	0.000022	0.009390
therapy	hair	0.000017	0.007879

We selected some typical word pairs with both high support values and confidence values in Table II and we can see some interesting behaviors. We can interpret the results as if people bought a deal which has "WORD1" in description; they also tend to buy another deal which has "WORD2". For example, the most frequent word pair in Table II is {buffet, buffet}. It indicates: 1) there are a lot of deals in the transactions having "buffet" in the description; 2) people who previously bought buffet deals tend to buy another different buffet deal. Words "buffet", "dinner", "hotel" and "massage" appear with "buffet" most, which means that people who bought deals about these words tend to buy buffet deals in the future. This also makes sense because other buffet deals also have these words in their description. Additionally we can see people who like to buy massage deals also like to buy buffet deals. The same observation holds for "treatment", "body", "facial", "hair", "spa" and "massage", e.g., people who bought body massage deals will buy facial spa deals later, meaning people buy beauty deals repeatedly. Interestingly we find not only for service deals (e.g., dining, beauty), but also some products deals, like Swarovski and Coach Products are repeated in people's buying patterns. This means we can recommend deals to customers who bought the same brand before. In light of these findings, our proposed recommendation method will recommend new deals which have keywords associated with customers' purchase history.

B. Recommendation Based on Keyword Association Rules

We design a recommendation algorithm to recommend to customers new deals based on the deal descriptions in their purchase history and the prior knowledge of word pair association rules computed in Section IV-A.

We already defined $D_l = \{d_1^l, \dots d_2^l, \dots, d_{\delta(l)}^l\}$ $(1 \leq l \leq L)$ as the list of historical deals that user u_l has purchased until now and it is sorted by date ascendingly. We can also define $D_T = \{d_1^T, \dots d_2^T, \dots, d_{\delta(T)}^T\}$ as all the deals that are available at the time of query, represented by T. The purpose of our algorithm is to sort D_T for particular user u_l so that the top deals will be of most interest to u_l. Based on Eq. (1) and Eq. (2), we define $C(\omega, \omega')$ as the confidence value of word pair $\{\omega, \omega'\}$. Algorithm 2 shows the steps of computing the weights for each deal in D_T.

Algorithm 2 Compute weights for deals to be recommended for user u_l.

1: For every deal $d_n \in D_T (1 \leq n \leq \delta(T))$, select all the words in its description, denoted as $W_n = \{\omega_1^n, \dots \omega_2^n, \dots \omega_{\theta(n)}^n\}$

2: Denote all words in D_l as $W_l = \left\{\omega_1^1, \dots \omega_{\theta(1)}^1, \omega_1^2, \dots \omega_{\theta(2)}^2, \dots \omega_1^{\delta(l)}, \dots \omega_{\theta(\delta(l))}^{\delta(l)}\right\} = \{\omega_1^l, \dots \omega_{\overline{\theta}}^l\}$, and the number of elements in W_l is $\overline{\theta} = \sum_{i=1}^{\delta(l)} \theta(i)$, which is the collection of words in purchase history for user u_l.

3: Weight for deal d_n is then computed by:

$$\alpha_n = \sum_{r=1}^{\theta(n)} \sum_{q=1}^{\overline{\theta}} C(\omega_r^n, \omega_q^l) \tag{3}$$

4: Normalize α_n to [0, 1] by dividing it by the maximum α_{max} in D_T.

C. Recommendation by Quantity Sold

The simplest and efficient recommendation method used by e-commerce websites is to just sort all deals by quantity of coupons sold. We define β_n as the number of coupons sold for deal $d_n \in D_T$. We also normalize β_n values for each deal by dividing the maximum value β_{max} in D_T.

We combine the weights from keyword association rules and quantity sold by:

$$\gamma_n = \alpha_n + \beta_n \tag{4}$$

and sort all the deals in D_T by γ_n values descendently. The top deals are the most relevant deals for user u_l in terms of keywords association and overall popularity.

D. Experiments

We conduct experiments to test our proposed algorithm with the baseline algorithm on the coupon deal database described in Section III. For every deal bought by the user, we regard it as unknown, and we recommend a list of deals based on the previous purchase history on the date of query, and see if the deal in question is among our recommended deals. If the deal is in the recommended list, we call it a hit. In our experiments we report the percentage of hits for all deals. As our algorithm depends on the purchase history, we skip the first purchase bought by the users and only recommend deals from the second purchase on. It's obvious that the higher the hit rate is, the better the recommendation is. We also compute the average rank for each hit, rank = 1 means the actual purchased deal by the user is at the top position of the recommended list. So lower average rank means better recommendation. We also compute the average Mean Reciprocal Rank (MRR) [12] based on the following equation:

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|}\frac{1}{rank_i} \quad (5)$$

where Q is the set of all hit deals. It's easy to see that the higher the MRR value, the better the recommendation.

Based on the findings in [9], we know that words are repeated more often in dining, wellness and activities deals because they are mostly service deals, while for products and travel deals, words are less likely to have association. We also observe that in this database people tend to buy deals in the same category very often. For example, if a customer buys a dining deals, so that his next purchase is probably also dining deal. Hence the procedure for our experiments is as follows:

1) For every user, get the purchase history at the query date.

2) If the last purchase was in category dining, wellness, activities, recommend deals by Eq. (4), else recommend deals by values of n.

3) Compute the percentage of hits, the average ranking of all hits, and the MRR.

We report the percentage of hits, average rank and mean reciprocal rank for the top 5, top 10 and top 20 recommendations. From the results in Fig. 2 we can see that our proposed algorithm which integrates quantity and keyword association outperforms quantity algorithm in all experiments in terms of percentage of hits, average rank and mean reciprocal rank for all cases. The percentage of hits increased from 12.9% to 13.9% for top 5 recommendation, from 21.1% to 21.6% for top 10 recommendation, from 35.7% to 36.9% for top 20 recommendation, yielding a 7.8%, 2.4%, 3.4% improvement for different cases. For average rank of each hit, our algorithm always has lower rank, which means better performance for all cases. Lastly, we also achieve higher mean reciprocal rank in all cases than the baseline, yielding an increase of 14.1%, 9.3%, and 11.9% for top 5, top 10 and top 20 recommendations separately.

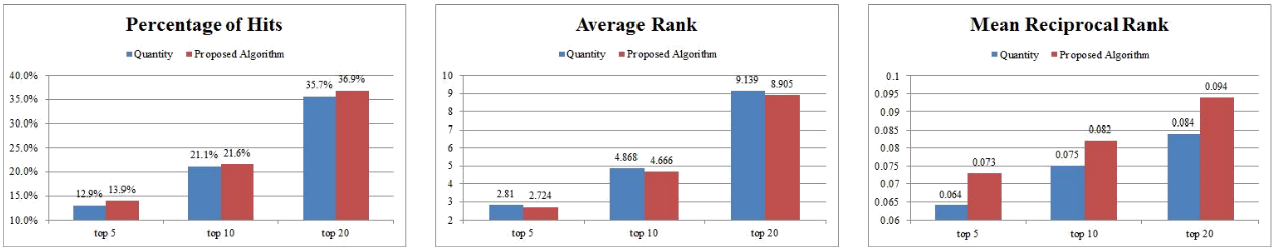

Figure 2. Experimental results for quantity sold only and our proposed algorithm which combines word association rules and quantity sold for top 5/10/20 recommendation separately.

E. Future Work

In the future, we want to look more closely into the transactions to find which deals are more suitable to use keyword association rules for recommendation. We also plan to integrate our keyword association rules into collaborative filtering algorithms to achieve better performance.

V. Conclusion

This paper proposed a new algorithm to recommend coupon deals based on word association rules to handle the "cold start" problem which are not so obvious in traditional recommendation cases. Based on previous findings, it makes a lot of sense to apply the algorithm on dining, wellness, and activity deals. Keyword association rules are computed and used to compute weights for every deal to be recommended and the weights are tailored for specific user personally. We conduct experiments of our proposed algorithm and compare the results with the baseline quantity sold algorithm and achieved significant improvement for top 5/10/20 recommendations in terms of percentage of hits, average rank and mean reciprocal rank.

Acknowledgment

This work was supported in part by the Singapore Economic Development Board (EDB) and National Research Foundation (NRF).

References

[1] N. S. A. Service. (May 2013). Four percent of Canadian transactions were made online in 2012. [Online]. Available: https://www.npdgroup.ca/wps/portal/npd/ca/news/press-releases/four-percent-of-canadian-transactions-were-made-online-in-2012/

[2] J. Lin, K. Sugiyama, M. Y. Kan, and T. S. Chua, "Addressing cold-start in app recommendation: Latent user models constructed from twitter followers," in *Proc. 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2013, pp. 283–292.

[3] F. E. Walter, S. Battiston, and F. Schweitzer, "A model of a trust-based recommendation system on a social network," *Autonomous Agents and Multi-Agent Systems*, vol. 16, no. 1, pp. 57–74, 2008.

[4] M. H. Park, J. H. Hong, and S. B. Cho, "Location-based recommendation system using Bayesian user's preference model

in mobile devices," in *Ubiquitous Intelligence and Computing*, Springer, 2007, pp. 1130–1139.
[5] S. Debnath, N. Ganguly, and P. Mitra, "Feature weighting in content based recommendation system using social network analysis," in *Proc. 17th International Conference on World Wide Web. ACM*, 2008, pp. 1041–1042.
[6] H. C. Chen and A. L. Chen, "A music recommendation system based on music data grouping and user interests," in *Proc. CIKM*, 2001, vol. 1, pp. 231–238.
[7] S. Ebrahimi, N. M. Villegas, H. A. Muller, and A. Thomo, "Smarterdeals: A context-aware deal recommendation system based on the smartercontext engine," in *Proc. Conference of the Center for Advanced Studies on Collaborative Research. IBM Corp*, 2012, pp. 116–130.
[8] Wikipedia. (Nov 2013). Collaborative Filtering. [Online]. Available: http://en.wikipedia.org/wiki/Collaborative filtering/
[9] Y. Huang, "Can keywords help personalized recommendation for coupon deals?" in *Proc. International Conference on Progress in Informatics and Computing*, May 2014, pp. 656–659.
[10] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. in Artif. Intell.*, Jan. 2009.
[11] J. Han, *Data Mining: Concepts and Techniques*, San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2005.
[12] E. Voorhees, "Trec-8 question answering track report," in *Proc. 8th Text Retrieval Conference*, 1999, pp. 77–82.

Yi Huang received her bachelor's degree from School of Software Engineering, Chongqing University, Chongqing, China in 2006. She received her PhD degree from School of Computer Engineering, Nanyang Technological University, Singapore in 2011.
She joined Barclays Capital as an IT ANALYST in 2011 and later joined SAP Asia as a RESEARCHER in 2013. Her research interests include data mining, image processing, computer vision and machine learning. She is also interested in applications such as recommendation systems for E-commerce platforms.
Ms. Huang was the recipient of the Microsoft Research Asia Fellowship in 2008.