Towards Ranking in On-line Deals for Recommender Systems in Coupons Market

Marcel Caraciolo
Orygens.com
Recife, Brazil
Email: marcel@orygens.com

Bruno Melo
Orygens.com
Recife, Brazil
Email: bjmm@acm.org

***Abstract*—Recommending deals and offers to users, based on their preferences, has long been a favourite domain for recommender systems research. However, different aspects of the deals in coupon systems such as expiration date, initial publishing date, number of coupons available, distribution of the deals per retailer (diversity) or even the popularity of the deal have often been overlooked. In this paper, we propose a ranking model that harnesses these particular characteristics at recommender systems for coupons market as the basis for promoting desirable deals. As study case we use a web coupon's system Favoritoz.com to carry out our experimentation of this technique which based on preliminary evaluations, shows promising results.**

***Index Terms*—recommendation; deals; coupons; market; ranking functions**

I. Introduction

The use of recommender systems in the domain of recommending products, deals and offers has been started for a long time [1], [2]. Recommender systems promise to promote the most relevant deals to a user based on their learned or preferences or even their previous consumption transactions, helping the user in question to receive personalized deals and offers and to save valuable time sifting through less relevant deals. The current approaches for this task such as content-based and collaborative filtering techniques [3] have been used in several coupon market systems such as Yipit (yipit.com) and demonstrate the value of the recommendation techniques when it comes to deliver the most relevant and compelling on-line deals from their retailer partners. For all the sucess of recommennder systems there are stills some aspects in on-line deals, specially in coupons market, that are not well catered for.

For instance, in this short article we will consider the problem of identifying the most popular deals or even take into consideration the importance of factors such as: when the coupons available are running out, that means, the customer needs to be remembered that deal is almost sold out. Other aspects such as the availability period of the offer on-line or the categories or retailers that you are most interested must be added in the final ranking score. Current recommender systems are limited in their ability to identify and consider all those aspects because, typically, they rely on a critical mass of user consumption before such deals can be recognised. We consider a different approach to conventional recommendation approaches by providing a ranking function weighting all those inputs as basis to compute the utility of a on-line coupon deal. To this I describe my initial attempts by analyzing over 200 on-line deals from the web sevice Favoritoz during the period of 9 months, with a view of identifying real-time desirable on-line deals ans suggest them to the users.

In the next section we describe our ranking system, focusing on the ranking model and highlighting how real-time information is extracted from the on-line deals to provice a basis for the recommendation. In the summary, we describe the results of a preliminary user evaluation to compare how the Favoritoz user's responded to those recommendations based on the ranking model strategy.

II. Our Ranking Model

The starting point for this paper is the idea that analyzing further details about the offers. This information could be used as the basis for a novel approach to ranking on-line deals so that the most desirable ones can be effectively promoted. We use Favoritoz, a brazilian on-line marketplace for on-line coupons, where retailers can publish offers and coupons associated with their products by providing discounts or advantages for their customers. There are several web sites that has the same proposal of Favoritoz, however, we developed a unique ranking model where customers would only receive personalized offers based on their interests or the most attractive ones based on the popularity or when the deal is running out reminding the users about the deal. Even the web site was designed to better adapt to our ranking model where the most ranked deals could be listed at the top while the less atractives would be at the bottom of the site. As the Figure 1 presents, the Favoritoz.com shows the main interface where the deals were listed based on our ranking model.

A. The Deal Model

Building our ranking model involved the detection and summing up of all available information provided in the on-line deals. Our model follows [4]'s methodology by incorporating the deals information and the user's preferences. We formally represent the ranking model into distinguished terms as tuple $\langle SD, ED, P, CA, UP, DV \rangle$, where:

- SD represents the *initial date* of the on-line deal, that is, when the on-line deal starts to be available for anyone to purchase. We define SD as the estimated value by

Fig. 1. On-line Deals recommended at Favoritoz.com user interface.

computing the difference in days between the current date and the initial date. Then, it is normalized using a exponential function in the form of

$$SD = exp^{(-1.0*days)/k}, \tag{1}$$

This indicator informs that recently published deals should be ranked higher than older deals.

- ED represents the *end date* of the deal, that is, the last date of the offer available for purchase. We define ED as the difference in days between the end date and the current date. Therefore, it is normalized using a exponential function in the form of

$$ED = exp^{(-1.0*days)/k}, \tag{2}$$

Unlike the SD, this term becomes higher when the deal is close to end, so it must be shown at the top suggestions to the user.

- P indicates the *popularity* of the deal. Deals purchased by many users provide strong envidence of an desirable deal to be noticed by the user. I consider popularity in this context by the number of purchased coupons of the offer. We normalize it by using the exponential distribution in the form of

$$P = \frac{1.0}{(1.0 + exp^{-coupons_{purchased}/k})}, \tag{3}$$

ranking higher deals which are most popular in the set of deals available.

- CA represents the *number of coupons available*. The less coupons left available more the possibility of interest in that deal. So offers with many coupons available tend not be purchased compared when there aren't many available. The sentiment of urgency becomes critical and may catch the attention of the customer to buy the last coupons. We also normalize it using the exponential distribution in the form of

$$CA = exp^{(-1.0*coupons_{available})/k} \tag{4}$$

, ranking deals with higher scores for this term when there are a few coupons left.

- UP indicates the *user's profile*. We had at our domain some information with the preferences of the user about the category of deals he liked (couisine, shows, movies, arts, sports, etc). This set of informations we ranked using a step function score. When the deal has the category m the user likes, represented by the set $userProfile_i$ it scores 1.0 otherwise 0.0.

$$UP = \begin{cases} 1.0 & if \quad m \in \quad userProfile_i \\ 0 & otherwise \end{cases} \tag{5}$$

- DV is the indicator of *diversity*. In coupons market where there are many retailers offering daily deals, they dispute the top positions to maximize their presence in the deals list. But sometimes it's not interesting for the user to have overcrowded deals from one specific retailer. For the administrator, this factor can influence the diversity of the deals. Therefore, this score ranks deals with higher rates when there are few deal's retailers concentrated. Otherwise, it decreases as there are many deals from the same retailers coming together. DV is represented by the equation

$$DV = exp^{(-1.0*calcstores(deal_{store}))/k} \tag{6}$$

where $calcstores$ is defined by

$$calcstores(deal_{store}) = \sum_{i=1}^{n} \begin{cases} 1 & ifd_{(i)store} == deal_{store}, \\ 0 & otherwise \end{cases} \tag{7}$$

The term k represents a constant factor that adjusts the strengthnes. d represents the deal that belongs to a set D of n items. The terms SD, ED do not require any user information, so we immediately overcome the cold start problem. Regardless of the user's purchase activity, we ensure that s/he will be recommended for deals where the offers are newer ones or the closest to end it. Two terms required in any coupon market systems.

B. *Weighing the Ranking Model*

As introduced in Section II-A, the ranking model comprises a set of weighted terms. Each one has its own importante for the ranking model. For instance, the term SD, corresponding to the start date, better exposes the novelty of the deal rather than the end date term ED because the latter will only bring the deals to top list only when it comes close to expire. In market, the people prefer see newer and interesting deals than the older and repeated ones.

The importance of those terms is not calculated by a mathematical model; instead, it is empirically predefined based on the administrator's common knowledge. In our preliminary results (see Section III), we suggest the appropriate weights respecting the following order of importance $\rho = SD > ED > UP > P > CA > DV$.

1) Ranking Function: To compute the overall score for each deal d, we simply compute the weighted correlation of the importance of each term (term weight) and term itself defined as:

$$W_i = Term_i \cdot w_i, \tag{8}$$

where W_i is the function that calculates the term weight for the term $Term_i$ in the set

$$s = \langle SD, ED, P, CA, UP, DV \rangle$$

with n attributes, and w_i is a constant assigned with the term's predefined importance value. Given the weighted terms W , we evaluate the final score $Score$ as the sum of those weighted terms. Formally:

$$Score(d) = W_1 + W_2 + \ldots + W_i = \sum_{i=1}^{n} W_i \tag{9}$$

Obviously this is a simple scoring mechanism but it does serve to provide a straightforward and justiable starting point. The ranking model is described as follows: for each user $u \in U$, we want to recommend the *unknown* deals $d^{max,u} \in D$, which maximize the personalized function $Score$ described as:

$$\forall u \in U, d^{max,u} = \arg\max_{d \in D} score(u, d) \tag{10}$$

The final score is normalized [0,1] which measures the importance of the corresponding deal to the user.

III. Preliminary Results

We were interested in how well the recommendations produced by this new ranking technique were received by end-users. We implemented this algorithm using the Python programming language. To test this we have carried out an A/B test evaluation over a period of 8 months. For 4 months we had the click-through frequency collected for deals without any recommendation strategy. It was date-based one (The most recent launched was always showing in the top). For the last 4 months we had our ranking model running at our web service. The results shown in Figure 2 represent the average per-user click-throughs for each of the strategies used. As result, we can there is a clear difference in the behaviour of users when our ranking model was running. There's a relative click-throug increase of 333% when we deployed our ranking model.

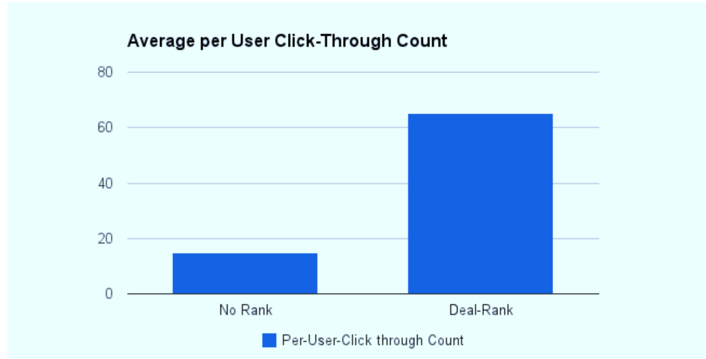

Fig. 2. Average per user click-through for presence of ranking strategy

IV. Conclusions

In this short paper, we have outlined a different approach for recommending on-line deals in coupon market. This technique harnesses particular aspects from the coupons as the basis for ranking and recommending deals. A web system has been developed and deployed using this ranking model to suggest daily deals for their users. Looking to the future, there are many opportunities for further inovation and experimentation for deals recommendation. There are also many ways that the content-based recommendation technique can be used to improve the recommendation ranking. For example, check the attributes of user profile and correlate with the attributes of the candidate deal. We also hope to focus on improving the scoring technique by applying models to adjust automatically the weights based on user's feedback. We believe, this article has the potential to act as a baseline in this research field with a number of opportunities to provide recommendation services such as recommending relevant retailers to follow on Favoritoz or recommending new categories based on what the user purchased.

References

[1] G. Linden, B. Smith, and J. York, "Amazon.com recommendations - item-to-item collaborative filtering," *IEEE Internet Computing*, vol. 7, no. 1, pp. 76–80, 2003.

[2] M. Mandl, A. Felfernig, E. Teppan, and M. Schubert, "Consumer decision making in knowledge-based recommendation," *J. Intell. Inf. Syst.*, vol. 37, no. 1, pp. 1–22, Aug. 2011. [Online]. Available: http://dx.doi.org/10.1007/s10844-010-0134-3

[3] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. on Knowl. and Data Eng.*, vol. 17, no. 6, pp. 734–749, Jun. 2005. [Online]. Available: http://dx.doi.org/10.1109/TKDE.2005.99

[4] P. Brusilovsky, "Adaptive hypermedia," *User Modeling and User-Adapted Interaction*, vol. 11, no. 1-2, pp. 87–110, Mar. 2001.