Risk-Aware Recommender Systems

Djallel Bouneffouf, Amel Bouzeghoub, and Alda Lopes Gancarski[*]

Department of Computer Science, Télécom SudParis, UMR CNRS Samovar,
91011 Evry Cedex, France
{Djallel.Bouneffouf,Amel.Bouzeghoub,Alda.Gancarski}@it-sudparis.eu

Abstract. Context-Aware Recommender Systems can naturally be modelled as an exploration/exploitation trade-off (exr/exp) problem, where the system has to choose between maximizing its expected rewards dealing with its current knowledge (exploitation) and learning more about the unknown user's preferences to improve its knowledge (exploration). This problem has been addressed by the reinforcement learning community but they do not consider the risk level of the current user's situation, where it may be dangerous to recommend items the user may not desire in her current situation if the risk level is high. We introduce in this paper an algorithm named R-UCB that considers the risk level of the user's situation to adaptively balance between exr and exp. The detailed analysis of the experimental results reveals several important discoveries in the exr/exp behaviour.

1 Introduction

User feedback (e.g., ratings and clicks) and situation (e.g., location, time) have become a crucial source of data when optimizing a Context-Aware Recommender System (CARS). Knowledge about the environment must be accurately learned to avoid making undesired recommendations which may disturb the user in certain situations considered as critical or risky. For this reason, the CARS has to decide, for each new situation, whether so far learned knowledge should be exploited by selecting documents that appear more frequently in the corresponding user feedback, or if never seen documents should be selected in order to explore their impact on the user situation, increasing the knowledge about the environment. On one hand exploration prevents from maximizing the short-term reward since it may yield to negative reward. On the other hand, exploitation based on an uncertain environment prevents from maximizing the long-term reward because document rating values may not be accurate. This challenge is formulated as an exploration/exploitation (exr/exp) dilemma. One smart solution for exr/exp using the "multi-armed bandit problem" is the hybrid approach done by [7]. This approach combines the Upper Confident Bound (UCB) algorithm with the ϵ-greedy algorithm. By introducing randomness into UCB, authors reduce the trouble in estimating confidence intervals. This algorithm estimates both the mean reward of each document and the corresponding confidence interval. With

[*] Corresponding author.

M. Lee et al. (Eds.): ICONIP 2013, Part I, LNCS 8226, pp. 57–65, 2013.

the probability 1-ϵ, this algorithm selects the document that achieves a highest upper confidence bound and, with the probability ϵ, it uniformly chooses any other document. The ϵ parameter essentially controls exr/exp. The problem is that it is difficult to decide in advance the optimal value of ϵ. We introduce in this paper an algorithm, named R-UCB, that computes the optimal value of ϵ by adaptively balancing exr/exp according to the risk of the user situation. We consider as risky or critical, a situation where it is dangerous to recommend uninteresting information for the user; this means that it is not desired, can yield to a trouble, or causes a waste of time for the user when reading a document which is not interesting for him in the current situation. In this case, the exploration-oriented learning should be avoided. R-UCB extends the UCB strategy with an update of exr/exp by selecting suitable user's situations for either exr or exp. We have tested R-UCB in an off-line evaluation with real data. The remaining of the paper is organized as follows. Section 2 reviews related works. Section 3 describes the algorithms involved in the proposed approach. The experimental evaluation is illustrated in Section 4. The last section concludes the paper and points out possible directions for future work.

2 Related Work

We refer, in the following, recent works that address the exr/exp trade-off (bandit algorithm) and the Risk-Aware Decision problem. Existing CARS systems are not considered in this paper, refer to [1] and [2] for further information.

Multi-armed Bandit Problem. Very frequently used in reinforcement learning to study exr/exp, the multi-documented bandit problem was originally described by [9]. Few research works are dedicated to study the contextual bandit problem in recommender systems, considering the user's behaviour as the context. In [7], authors extend UCB by dynamically updating outperforming both beginning and decreasing strategies. In [6], assuming the expected reward of a document is linear, they perform recommendation based on contextual information about the users' documents. To maximize the total number of user's clicks, this work proposes the LINUCB algorithm which is computationally efficient. [7, 6] describe a smart way to balance exr/exp, but do not consider the user's situation and its associated risk during the recommendation.

The Risk-Aware Decision. To the best of our knowledge, the risk-aware decision is not yet studied in recommender systems. However, it has been studied for a long time in reinforcement learning, where the risk is defined as the reward criteria that takes into account not only the expected reward, but also some additional statistics of the total reward, such as its variance or standard deviation [3]. The risk is measured with two types of uncertainties. The first, named parametric uncertainty, is related to the imperfect knowledge of the problem parameters. For instance, in the context of Markov decision processes, [5] proposes to use the percentile performance criterion to control the risk sensitivity. The second type, termed inherent uncertainty, is related to the stochastic nature of the system, like [4], which consider models where some states are error states representing a catastrophic result. More recently, [10] developed a policy gradient algorithm for criteria that involves both the expected cost and the variance of the cost, and demonstrated the applicability of the algorithm in a portfolio planning problem. However, this work does not consider the risk of the situations in the exr/exp problem. A recent work, [11], treated the risk and proposed the VDBE algorithm to extend ϵ-greedy by introducing a state-dependent exploration probability, instead of hand-tuning. The system makes exploration in situations when the knowledge about the environment is uncertain, which is indicated by fluctuating action values during learning. In contrast, the amount of exploration is reduced as far as the system's knowledge becomes certain, which is indicated by very small or no value differences.

Our Contributions. As shown above, none of the mentioned works tackles the exr/exp problem considering the semantic risk level of the situation. This is precisely what we intend to do by exploiting the following new features: (1) Handling semantic concepts to express situations and their associated risk level. The risk level is associated to a whole situation and/or the concepts composing the situation; (2) Considering the risk level of the situation when managing exr/exp, which helps CARS to adapt them selves to environment dynamically. High exploration (resp. high exploitation) is achieved when the current user situation is "not risky" (resp. "risky"); (3) Assuming that exploring data in non-risky situations is useful for making a safety exploration in risky situations. Our algorithm performs exploration in risky situations by selecting the most interesting documents in non risky situations.

We improve the extension of UCB with ϵ-greedy (called here ϵ-UCB) because it gives the best results in an off-line evaluation done by [7]; however, our amelioration can be applied to any bandit algorithm.

3 The Proposed CARS Model

This section focuses on the proposed model, starting by introducing the key notions used in this paper.

***Situation*:** A situation is an external semantic interpretation of low-level context data, enabling a higher-level specification of human behaviour. More formally, a situation S is a n-dimensional vector, $S = (O_{\delta_1}.c_1, O_{\delta_2}.c_2, ..., O_{\delta_n}.c_n)$ where each c_i is a concept of an ontology O_{δ_i} representing a context data dimension. According to our need, we consider a situation as a 3-dimensional vector $S = (O_{Location}.c_i, O_{Time}.c_j, O_{Social}.c_k)$ where c_i, c_j, c_k are concepts of Location, Time and Social ontologies.

***User preferences*:** User preferences UP are deduced during the user navigation activities. $UP \subseteq D \times A \times V$ where D is a set of documents, A is a set of preference attributes and V a set of values. We focus on the following preference attributes: *click*, *time* and *recom* which respectively correspond to the number of clicks for a document, the time spent reading it and the number of times it was recommended.

***The user model*:** The user model is structured as a case base composed of a set of situations with their corresponding UP, denoted $UM = \{(S^i; UP^i)\}$, where $S^i \in S$ is the user situation and $UP^i \in UP$ its corresponding user preferences.

We propose CARS to be modelled as a contextual bandit problem including user's situation information. Formally, a bandit algorithm proceeds in discrete trials $t = 1...T$. For each trial t, the algorithm performs the following tasks:

Task 1. Let S^t be the current user's situation, and PS the set of past situations. The system compares S^t with the situations in PS in order to choose the most similar one, S^p:

$$S^p = argmax_{S^i \in PS} sim(S^t, S^i) \tag{1}$$

The semantic similarity metric is computed by:

$$sim(S^t, S^i) = \frac{1}{|\Delta|} \sum_{\delta \in \Delta} sim_\delta(c^t_\delta, c^i_\delta) \tag{2}$$

In Eq. 2, sim_δ is the similarity metric related to dimension δ between two concepts c^t_δ and c^i_δ, and Δ is the set of dimensions (in our case Location, Time and Social). The similarity between two concepts of a dimension δ depends on how closely c^t_δ and c^i_δ are related in the corresponding ontology. To compute sim_δ, we use the same similarity measure as [8]:

$$sim_\delta(c^t_\delta, c^i_\delta) = 2 * \frac{deph(LCS)}{deph(c^t_\delta) + deph(c^i_\delta)} \tag{3}$$

In Eq. 3, LCS is the Least Common Subsumer of c^t_δ and c^i_δ, and $deph$ is the number of nodes in the path from the current node to the ontology root.

Task 2. Let D^p be the set of documents recommended in situation S^p. After retrieving S^p, the system observes rewards in previous trials for each document $d \in D^p$ in order to choose for recommendation the one with the greatest reward, which is the Click Through Rate (CTR) of a document. In Eq. 4, the reward of document d, $r(d)$, is the ratio between the number of clicks (v_i) on d and the number of times d is recommended (v_j).

$\forall d \in D^p, UP^i{=}(d, click, v_i) \in$ UP and $UP^j{=}(d, recom, v_j) \in$ UP we have:

$$r(d) = \frac{v_i}{v_j} \tag{4}$$

Task 3. The algorithm improves its document selection strategy with the new observation: in situation S^t, document d obtains a reward $r(d)$. Depending on the similarity between the current situation S^t and its most similar situation S^p, two scenarios are possible: (1) If sim(S^t, S^p) $\neq 1$: the current situation does not exist in the case base; the system adds this new case composed of the current situation S^t and the current user preferences UP^t; (2) If sim(S^t, S^p) = 1: the situation exists in the case base; the system updates the case having premise the situation S^p with the current user preferences UP^t.

The ϵ-UCB Algorithm. For a given situation, the algorithm recommends a predefined number of documents, specified by parameter N using Eq. 5. Specifically, in trial t, this algorithm computes an index $b(d) = r(d) + c(d)$ for each document d, where: $r(d)$ (Eq. 4) is the mean reward obtained by d and $c(d)$ is the corresponding confidence interval, so that $c(d) = \sqrt{\frac{2 \times log(t)}{v_j}}$ and v_j is the number of times d was recommended. With the probability 1-ϵ, ϵ-UCB selects the document with the highest upper confidence bound $d_t = argmax_{d \in D^p} b(d)$; and with the probability ϵ, it uniformly chooses any other document.

$$d_t = \begin{cases} argmax_{d \in (D^p - RD)} b(d) \quad if\ q > \epsilon \\ Random(D^p - RD)\ otherwise \end{cases} \tag{5}$$

In Eq. 5, D^p is the set of documents included in the user's preferences UP^p corresponding the most similar situation (S^p) to the current one (S^t); RD is the set of documents to recommend; *Random*() is the function returning a random element from a given set; q is a random value uniformly distributed over [0, 1] which controls exr/exp; ϵ is the probability of recommending a random exploratory document.

Semantic Risk Level Computing. In real world, the exr/exp trade-off should be directly related to the risk level of the situation, this is why computing the risk level is indeed indispensable. To consider the semantic risk level of the situation in exr/exp, we add a risk level to each concept in a situation: $S = (O_{\delta_1}.c_1[cv_1], O_{\delta_2}.c_2[cv_2], ..., O_{\delta_n}.c_n[cv_n])$, where $CV = \{cv_1, cv_2, ..., cv_n\}$ is the set of risk levels assigned to concepts c_i $(i = 1..n)$ and $cv_i \in [0, 1]$. $R(S)$ is the risk level of situation S. $R(S) \in [0, 1]$ and situations having $R(S) > th_R$ are considered as *risky* or *critical* situations (CS). The risk threshold th_R is described in Subsection 3. The risk of a concept varies from a domain to another and it is predefined by a domain expert. We conducted a study with professional mobile users, described in detail in Sec 4, where the domain expert is a commercial manager, and we considered, for example, the following set of critical situations: $CS = \{CS1, CS2, CS3\}$, $CS1 = (-, afternoon, manager)$, $CS2 = (company, morning, -)$, $CS3 = (-, -, client)$.

The risk level $R(S^t)$ of situation S^t is computed as follows:

$$R(S^t) = \begin{cases} R_c(S^t) \text{ if } CS = \emptyset, CV \neq \emptyset \\ R_m(S^t) \text{ if } CS \neq \emptyset, CV = \emptyset \\ \frac{1}{2} \times (\eta R_c(S^t) + \zeta R_m(S^t)) \text{ if } CS \neq \emptyset, CV \neq \emptyset \end{cases} \tag{6}$$

If only CV is known, Eq. 6 returns the risk $R_c(S^t)$ (Eq. 7) inferred from the situation concepts. If only CS is known, Eq. 6 returns the the risk $R_m(S^t)$ (Eq. 9) extracted from the degree of similarity between the current situation S^t and the centroid critical situation S^m (Eq. 10). If CV and CS are both known, Eq. 6 returns the weighted mean between $R_c(S^t)$ and $R_m(S^t)$; η and ζ are respectively the weights associated to $R_c(S^t)$ and $R_m(S^t)$, where $\eta + \zeta = 2$. These weights are related to the application domain: if the domain is itself risky (e.g. healthcare and safety), the system considers that R_c is more important than R_m (during the experimental phase, η and ζ have both a value of 1). $R_c(S^t)$ gives a weighted mean of the risk level of the concepts:

$$R_c(S^t) = \frac{1}{|\Delta|}(\sum_{\delta \in \Delta} \mu_\delta cv_\delta^t) \tag{7}$$

In Eq. 7, cv_δ^t is the risk level of the dimension δ in S^t and μ_δ is the weight associated to dimension δ. μ_δ is set out by using an arithmetic mean as follows:

$$\mu_\delta = \frac{1}{|CS|}(\sum_{S^i \in CS} cv_\delta^i) \tag{8}$$

The idea in Eq. 8 is to get a measure of how risky are, in average, concepts of dimension δ in CS, computing the mean of all the risk levels associated to δ in CS. Being B the similarity threshold (this metric is fixed on the off-line evaluation) and S^m the critical situation centroid, $R_m(S^t)$ is computed as follows:

$$R_m(S^t) = \begin{cases} 1 - B + sim(S^t, S^m) \ if \ sim(S^t, S^m) < B \\ \qquad\qquad 1 \qquad\quad otherwise \end{cases} \tag{9}$$

In Eq. 9, the situation risk level $R_m(S^t)$ increases when the similarity between S^t and S^m increases. The critical situation centroid is selected as follows:

$$S^m = argmax_{S^f \in CS} \frac{1}{|CS|} \sum_{S^e \in CS} sim(S^f, S^e) \tag{10}$$

The R-UCB Algorithm. To improve the adaptation of the ϵ-UCB algorithm to the risk level of the situations, the R-UCB algorithm (Alg. 1) computes the probability of exploration ϵ (line 2), by using the situation risk level $R(S^t)$ as indicated in Eq. 11. In Eq. 11. ϵ_{min} is the minimum exploration allowed in CS and ϵ_{max} is the maximum exploration allowed in non-CS (these two metrics are computed off-line using exponentiated gradient, which gives $\epsilon_{min} = 0.1$ and $\epsilon_{max} = 0.5$).

$$\epsilon = \epsilon_{max} - R(S^t) \times (1 - \epsilon_{min}) \tag{11}$$

Depending on the risk level of the current situation S^t, two scenarios are possible: (1) If $R(S^t) < th_R$, S^t is not critical(Alg. 1, line 3); the ϵ-UCB algorithm is used with $\epsilon > \epsilon_{min}$ (Eq. 11). (2) If $R(S^t) \geq th_R$, S^t is critical (Alg. 1, line 4); the ϵ-UCB algorithm is used with $\epsilon = \epsilon_{min}$, performing a high exploitation (Alg. 1, line 6, instruction 1). Based on our supposition that data in non critical situations can be useful to infer optimal exploration in CS, the algorithm makes a safety exploration in CS by selecting the documents with the highest CTR in D^y (Alg. 1, line 6, instruction 2), where D^y is the set of documents recommended in the most similar situation S^y to S^t, $S^y \notin CS$ and computed using Eq. 1. We still consider random exploration (Alg. 1, line 6, instruction 3) which is indispensable to avoid documents selection in CS becoming less optimal. To summarize, the system makes a low and safety exploration when the current user's situation is critical (Alg. 1, line 6); otherwise (Alg. 1, line 3), the system performs high exploration. In this case, the degree of exploration decreases when the risk level of the situation increases (Eq. 11. To verify if S^t is critical, the risk threshold th_R is computed as indicated in Eq. 12. At the initialization phase, the domain expert may indicate risk levels for a set of concepts and/or situations. If risk levels on concepts are indicated, the expert defines the θ threshold; if risk levels on situations are indicated, the expert defines the B threshold.

$$th_R = \begin{cases} \theta \text{ if } CS = \emptyset, CV \neq \emptyset \\ B \text{ if } CS \neq \emptyset, CV = \emptyset \\ \frac{1}{2} \times (\eta\theta + \zeta B) \text{ if } CS \neq \emptyset, CV \neq \emptyset \end{cases} \tag{12}$$

Algorithm 1. The R-UCB algorithm

```
1: Input: $S^t, D^p, D^y, RD = \emptyset, B, N, \epsilon_{min,\epsilon_{max}}$ Output: $RD$
2: $\epsilon = \epsilon_{max} - R(S^t) \times (1 - \epsilon_{min})$ //$R(S^t)$ is computed as described in Eq.6
3: if $R(S^t) < th_R$ then $RD = \epsilon$-UCB$(\epsilon, D^p, RD, N)$ else
4: for $i = 1$ to $N$ do
5:     $q = Random(0,1); k = Random(0,1)$
6:  $d_i = \begin{cases} argmax_{d \in (D^p - RD)} b(d) & \text{if} \quad q > \epsilon_{min} \\ argmax_{d \in (D^y - RD)} b(d) & \text{if} \quad q \leq k \leq \epsilon_{min} \\ Random(D^p) \; otherwise \end{cases}$
7:     $RD = RD \cup d_i$
8: end for
```

4 Experimental Evaluation

We conducted a diary study with the collaboration of a software company. This company provides a history application, which records time, current location, social and navigation information of its users during their application use. The diary study took 2 months and generated 356 738 diary situation entries. From the diary study, we have obtained a total of 5 518 566 entries concerning the user's navigation (number of clicks and time spent), expressed with an average of 15.47 entries per situation.

Off-line Evaluation. To test the proposed R-UCB algorithm, in our experiments, we have firstly collected the 100 000 cases from the situations case base.

We then evaluate the existing algorithms by confronting them, at each iteration, to a case randomly selected and removed. We calculate the average CTR every 1000 iterations. The number N of documents returned by the CARS for each situation is 10 and we run the simulation during 10000 iterations, where all the tested algorithms have converged. In the first experiment, in addition to a pure exploitation baseline, we compare our algorithm to the ones described in the related work (Sec. 2): VDBE, EG-UCB, ϵ-UCB, beginning-UCB (ϵ-UCB with beginning strategy) and decreasing-UCB (ϵ-UCB with decreasing strategy). In Fig. 1, the horizontal axis represents the number of iterations and the vertical axis is the performance metric.

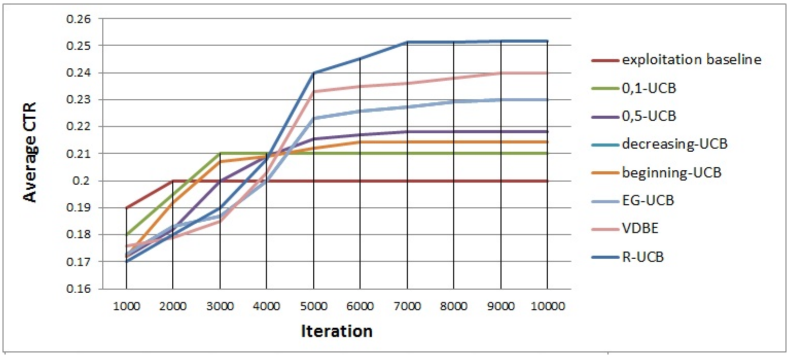

Fig. 1. Average CTR for exr/exp algorithms

R-UCB and VDBE effectively have the best convergence rates; VDBE increases the average CTR by a factor of 1.5 over the baseline and R-UCB, by a factor of 2. The improvement comes from a dynamic exr/exp, controlled by the risk level estimation. These algorithms take full advantage of exploration when the situations are not critic, giving opportunities to establish good results when the situations are critical. Finally, as expected, R-UCB outperforms VDBE, which is explained by the good estimation of the risk.

5 Conclusion

In this paper, we study the problem of exr/exp in CARS and propose a new approach that adaptively balances exr/exp regarding the risk level of the situation. We have validated our work with off-line studies which offered promising results. This study yields to the conclusion that considering the risk level of the situation on the exr/exp strategy significantly increases the performance of the recommender system. In considering these results, we plan to investigate public benchmarks.

References

1. Bouneffouf, D., Bouzeghoub, A., Gançarski, A.L.: A contextual-bandit algorithm for mobile context-aware recommender system. In: Huang, T., Zeng, Z., Li, C., Leung, C.S. (eds.) ICONIP 2012, Part III. LNCS, vol. 7665, pp. 324–331. Springer, Heidelberg (2012)
2. Bouneffouf, D., Bouzeghoub, A., Gançarski, A.L.: Hybrid-ϵ-greedy for mobile context-aware recommender system. In: Tan, P.-N., Chawla, S., Ho, C.K., Bailey, J. (eds.) PAKDD 2012, Part I. LNCS, vol. 7301, pp. 468–479. Springer, Heidelberg (2012)
3. Cherian, J.A.: Investment science: David g. luenberger. Journal of Economic Dynamics and Control 22(4), 645–646 (1998)
4. Geibel, P., Wysotzki, F.: Risk-sensitive reinforcement learning applied to control under constraints. J. Artif. Int. Res. 24(1), 81–108 (2005)
5. Howard, R.A., Matheson, J.E.: Risk-sensitive markov decision processes. Management Science 18(7), 356–369 (1972)
6. Li, L., Chu, W., Langford, J., Schapire, R.E.: A contextual-bandit approach to personalized news article recommendation. In: Proceedings of the 19th International Conference on World Wide Web, WWW 2010, pp. 661–670. ACM, USA (2010)
7. Li, W., Wang, X., Zhang, R., Cui, Y.: Exploitation and exploration in a performance based contextual advertising system. In: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2010, pp. 27–36. ACM, USA (2010)
8. Mladenic, D.: Text-learning and related intelligent agents: A survey. IEEE Intelligent Systems 14(4), 44–54 (1999)
9. Robbins, H.: Some Aspects of the Sequential Design of Experiments. Bulletin of the American Mathematical Society 58, 527–535 (1952)
10. Sehnke, F., Osendorfer, C., Rückstieß, T., Graves, A., Peters, J., Schmidhuber, J.: Policy gradients with parameter-based exploration for control. In: Kůrková, V., Neruda, R., Koutník, J. (eds.) ICANN 2008, Part I. LNCS, vol. 5163, pp. 387–396. Springer, Heidelberg (2008)
11. Tokic, M., Ertle, P., Palm, U., Soffker, D., Voos, H.: Robust Exploration/Exploitation trade-offs in safety-critical applications. In: Proceedings of the 8th International Symposium on Fault Detection, Supervision and Safety of Technical Processes, pp. 660–665. IFAC, Mexico City (2012)