Journal of Engineering and Applied Sciences 12 (Special Issue 9): 8746-8750, 2017
ISSN: 1816-949X

Reinforcement Learning Model for Classification of Youtube Movie

[1]Munir Naveed and [2]Muath Alrammal
[1]Department of Information Technology, Khawarizmi International College,
University Campus, Abu Dhabi, UAE
[2]Laboratoire d'Informatique Fondamentale d'Orleans (LIFO), Univeriste d'Olreans, Olreans, France

Abstract: In this study, a new machine learning algorithm called "RBCM" is presented to generalize the context-awareness problem for high dimensional and partially observable domains. The approach uses a non-linear regression method to build a non-deterministic probability function for generalization. A machine learning model is used to generalize the action schema given in the domain with a context. Monte-Carlo simulations are used to map high dimensional spaces to contextual spaces. A non-linear regression based value-function is applied on contextual spaces to classify the contexts into class labels. The performance of RBCM is measured using Youtube movie benchmark dataset. Youtube movie dataset has links to movies and preferences of users about the movie pairs. The movies are mapped to contexts and then contexts used to predict a user will like a movie or not. The approach is compared with the current sophisticated machine learning models. These models are decision tree (J48), bootstrapping (Ada) and Naive-Bayes. The experimental outcomes reveal that RBCM performs significantly better than its rival models.

Key words: Reinforcement learning, context-awareness, regression, Monte-Carlo simulation, model, domains

INTRODUCTION

The synergy between mobility and e-Commerce is growing rapidly, mainly due to technical sophistication provided by mobility for commerce activities (Efraim and David, 2003). The mobility also adds convenience in performing e-Commerce tasks from anywhere and anytime other than technical sophistication. However, the interaction between mobility and e-Commerce also offers a lot complexities and challenges for the service providers. One of these challenges is the sensitivity of the mobility based e-Commerce applications to the context-awareness.

Traditional e-Commerce applications are designed without context-aware functionality, mainly due to lack of sensor data that can provide contextual information as compared to mobile devices that are equipped with several sensory information, e.g., location and Gyro information. In fact, traditional e-Commerce applications are based on a static and predefined context established by a system designer. However, mobile apps can easily exploit the capabilities of being context aware.

There are several ways a mobile App. can take benefit of being context-aware. Due to this freedom, there have been several studies on the best of context-awareness in computing solutions, e.g., Ubiquitous computing, human machine interaction, information retrieval, recommendation systems, ambient intelligence, mobile computing and sensorial networks (Orsi and Tanca, 2011; Martinenghi and Torlone, 2009; Hong *et al*., 2009; Bolchini *et al*., 2007).

One potential use of context awareness (Huang, 2014) is to build user preferences based on classification. In this study, we explore the use of context-aware methods in classifying the user preferences for movies. We present a new approach called RBCM to generalize the context-awareness problem for high dimensional domains in mobile computing. RBCM is built on a multivariate regression method to generalize a non-deterministic probability function.

Literature review: A context-aware model, presented by Pozveh *et al*. (2009) is applied in a mobile recommendation system. The context-aware model is built in conjunction with a collaborative filtering process. A similarity function (based on trigonometry) is used to search for contents using the current context of the user. The semantics for representation of a context are defined using a parametric based evaluation function. The evaluation function can transform a contextual query into a content search and finds the contents that have highest similarity to the current context. The contents are further refined by using Self-Organizing Maps (SOM). The context-aware model is

Corresponding Author: Munir Naveed, Department of Information Technology, Khawarizmi International College, University Campus, Abu Dhabi, UAE

evaluated empirically using the custom-build datasets. The results demonstrate that the context-aware model given by Pozveh *et al.* (2009) performs better when the size of recommendations escalate.

By Hmouz *et al.* (2009), a context-aware model is presented for categorization of the contextual aspect. The model applies the machine learning algorithms to query the relevant aspects of a context. The machine learning algorithms are tuned for personalization tasks. The model performs two main functionalities: personalization and customization of contents. The model applies classification algorithms to categorize the data into the classes where each class represents a personal context and/or a shared context. The shared context includes the community based features. The hybrid of personal and shared contextual information creates a personalized profile for a user.

DCPE co-training (Xu *et al.*, 2012) is a classifier to categorize data into classes. The classifier is a hybrid of different machines learning algorithms. The algorithms include neural networks, Naive Bayes and decision tree. The machines learning algorithms learn to classify data in a dome and the benefit of using two different algorithms at the same time is to predict better labeled data to be assigned to unlabeled data. DCPE co-trainings promote diversity in both classifiers which results in improved accuracy. However, DCPE can take benefit of being tested in domains with large number of input features to validate the strength of this approach.

Monte-Carlo (MC) Simulations are used for power scheduling by Haroonabadia and Haghifamb (2011) where demand forecasting is performed by using a fuzzy logic based estimation model. MC simulations are run using a data-driven based generative model that can simulate the stochastic behavior of demand and supply process in power scheduling. MC simulator estimates “Loss of Load Expectation (LOLE)”. An error function-based on difference between actual LOLE and estimated LOLE is used to measure performance of MC simulator. MC simulator is compared to a four-layered neural network for the same data set. The results show that performance of MC simulator is comparable to the trained neural network.

Ensemble of classifiers have been explored by Catal *et al.* (2015) to solve the Human Activity Recognition (HAR) problem. Performance of classifier ensemble is determined by using accuracy, area under curve and F-measure. The result shows that Ensemble of classifiers perform better than multi-layered neural network, logistic regression and C4.5 (decision tree) in accuracy. However, multi-layered neural network per-forms better to classify observations better than other models if the observation belongs to either “jogging” or “standing” classes. By Moro *et al.* (2014), the researchers present an application of different machine learning algorithms to predict outcome of telemarketing calls for selling bank long-term deposits. The success of such an approach can give a cost-effective and high profit yielding edge to the organization. The data from 2008-2013 was collected from a Portuguese bank. The dataset has a large number of observations which are unevenly distributed with respect to the class labels. For training the machine learning algorithms, a hand-out validation is used. The authors present a semi-automatic feature selection method to reduce the complexity of the search space. The researchers explored four classification models which are Logistic Regression (LR), Decision Tree (DT), Artificial Neural Networks (ANN) and a flavor of Support Vector Machine (SVM). The authors used the built-in implementation of these algorithms in rMiner library given in R language. The algorithms are compared using receiver operating characteristic (ROC) curve and cumulative lift. The results show that the best performance is achieved at 0.80 of ROC curve and an area of 0.7 for cumulative lift curve.

MATERIALS AND METHODS

Motivations and contributions: The use of a probabilistic model to solve context-awareness problem is an ideal solution because of their agility and adaptive characteristic to the dynamic and partially observable environment. A major of the high performing models for context-awareness are probabilistic, e.g., MC simulations (Naveed *et al.*, 2012), Naive Bayes (Xu *et al.*, 2012; Abbasnejad *et al.*, 2013) and decision tree (Nguyen *et al.*, 2008).

In this research, we formulate the context-awareness problem as a classification problem. We apply a regression technique to construct a probability density function to predict a class for a set of input features. The regression model can provide a mapping between input features and the class labels. The contributions are as the following: we present a novel machine learning algorithm, called RBCM, to solve feature selection and prediction problems. RBCM exploits the strengths of Monte-Carlo simulations and regression.

We also present theoretical aspects of RBCM including the convergence to a global optimal. We provide an example to illustrate the simulation process of RBCM using Monte-Carlo simulation and regression. To the best of our knowledge, we are the first who used the dataset Youtube comedy slam in order to evaluate the prediction capabilities of RBCM for the video preference of users. The study also presents a comparison of RBCM with four state-of-the-art best performing prediction algorithms.

Proposed solution; RBCM: In this study, we first introduce the basic terminologies and fundamental theories related to RBCM. Then, we present the algorithmic details of RBCM. Finally, we present an example to demonstrate the learning mechanism of RBCM.

Formalism:

- Definition 1; Domain has a finite set of input features A
- Definition 2; X is a tuple representing the contextual parameters of the domain
- Definition 3; U is a finite set of preferences for a user. For a given $x \in X$, RBCM predicts a preference $u \in U$
- Definition 4; $P(X|u)$ represents the probability density function for context X and user preference $u \in U$
- Definition 5; I represents a finite set of input features
- Definition 6; O represents a finite set of observations.
- Definition 7; $Q(O, u)$ represents the value function for each observation $o \in O$ for a preference $u \in U$. Q function is used to determine an appropriate preference of an observation
- Definition 8; $R(X, u)$ represents regression coefficient of a linear regression between a preference u and a context X
- Definition 9; $T: O \rightarrow X$ is a transformation from an observation o to a context x
- Definition 10; W is a weight vector that is initialized randomly in a window (0, 1)

RBCM regression model: The probability distribution of each $x \in X$ is determined by using a weighted regression model as shown in Eq. 1. The probability of each preference $x \in X$ is determined by using a weighted approach where regression of each input feature $o \in O$ to the preference is calculated using a linear regression model R. Each $w_x \in W$ is the initial weight of each context $x \in X$:

$$P(x \in X) = \prod\nolimits_{i=1}^{|U|} (w_x \times R(x, i)) \quad (1)$$

P distribution is used to determine the value function of the preference set, i.e., Q(O, U)) for each generalization. Q(O, U) is used to rank the presence for each observation (of O features). RBCM selects the preference that has the highest Q(O, U) in an observation. The value function is calculated using Eq. 2:

$$Q(O,U) = \sum\nolimits_{k=1}^{|U|} \sum\nolimits_{i=1}^{|X|} P(T(o_i) \,|\, u_k) \times O_i \quad (2)$$

Example: Suppose we have a training dataset of four observations as given in Table 1. The context-aware tuple is X = {C1, C2, C3} and preferences set is U = a, b. The

Table 1: Example dataset

Observation No.	C1	C2	C3	X
1	1	3	3	A
2	1	4	3	B
3	1	6	7	A
4	2	1	4	B

Table 2: Q value for first observation example dataset

Parameters of Q function	Values
$Q(O_1, a)$	0.013956
$Q(O_1, b)$	0.005184

tuple element C1 is set of values of the first contextual parameter, i.e., C1 = {1, 2}. Similarly, other elements of X are C2 = {1, 3, 4, 6} and C3 = {3, 4, 7}. The weight vector $W = \{W_{C1}, W_{C2}, W_{C3}\}$ is for example, $W_{C1} = \{0.01; 0.03; 0.9\}$, $W_{C2} = \{0.02, 0.03, 0.1\}$ and $W_{C3} = \{0.008, 0.06\}$. $P(C1|a) = W_{C1}[1]$ X R(C1[1], a) X $W_{C1}[2]$ XR(C1[2], a) X$W_{C1}(3)$ X R(C1(3), a) = 0:000783. Similarly, P(C1|b) = 0:000294, P(C2|) = 0:000831, P(C3|a) = 0:00356, P(C2|b) = 0:0000987 and P(C3|b) = 0:000643.

RBCM applies the p-values of each preference to build the value function. The Q value of the first observation are given in Table 2. RBCM selects the preference that has the highest Q value. For the example dataset, RBCM will select a as the preference for the first observation as $Q(O_1, a)$ is $>Q(O_1, b)$.

Algorithm details: Algorithm 1 shows a high level description of the proposed algorithm. RBCM uses a predefined limit on learning called $SE_{UpperLimit}$ as given in 1. RBCM reads the observations from the training dataset and stores them in O (Line 1). It builds the context-aware data structure X and extracts a set of preferences (U) from the training dataset. The weight vector is randomly initialized (Line 4, Algorithm 1). RBCM iteratively learns P and Q values. P(X|U) is determined by using the weight vector and linear regression (Line 6). The process for P(X) is repeated several time until the estimated error is less than a predefined threshold.

Algorithm 1; High level design of RBCM:

```
Procedure RBCM(T, X)
Read access SE_UpperLimit
1.  O: = Read training dataset()
2.  X: = ConstructContextAwareTupe(O)
3.  U: = BuildPreferencesSet (O)
4.  W: = InitialWeightVector ()
5.  Repeat
6.    P(X|U): = ΠW×R(X, U)
7.      For each o∈O
8.      For each u∈U
9.    Q(o, u): = MonteCarloSimulation (o, u, P, X, T )
10.   E_o: = ErrorfUnction (Q(o, u), O)
11.   W: = W+E_o
12.    RMSE: = RMSE(E_o)
13.    END FOREACH
14.   END FOREACH
15.   UNTIL RMSE>SE_UpperLimit
End RBCM
```

RBCM determines the preferences by using the learned values of P(X|U). The selection method is trivial as it uses the preference with the highest Q value for an observation. The core of the RBCM is a Monte-Carlo simulation model which is based on Upper bound Confidence interval (UCT) based sampling technique.

RESULTS AND DISCUSSION

Experimental setup: Experiments run on Windows server 2012, Intel(R) Xeon(R) CPU E5-2630 v2 2.6 GHz, RAM 32 GB. RBCM was simulated using Visual studio 2010, C# language. WEKA 3.7 was used to compare our model with the related techniques (Naive-bayes, boot-strapping (Ada) and decision tree (J48)). The learning model of each technique was validated using 10 folds (k folds). Accuracy, precision and F-measure values were measured. In RBCM, the average values of 10 simulations were considered for comparison.

Youtube dataset contains about 1.7 million preference votes. The first 80% of the dataset represent the training part and the remaining 20% represent the testing part. In Youtube dataset, each line represents a pair with first member as a preference. Each video is uniquely identified by its "Youtube video ID". For each observation, there is an anonymous user vote. In every row, there are three columns (separated by comma). The first and second columns are about the IDs of the movies, respectively. The third column is about the preference of the user. If user says "left", it prefers the first video and in case of "right", the choice is for the second video. Due to large size of dataset, all rivals techniques were not scalable and only RBCM could generalize classification task for Youtube. For a fair comparison, we randomly selected 150 k samples from Youtube dataset and run all experiments based on this subset for the Youtube dataset.

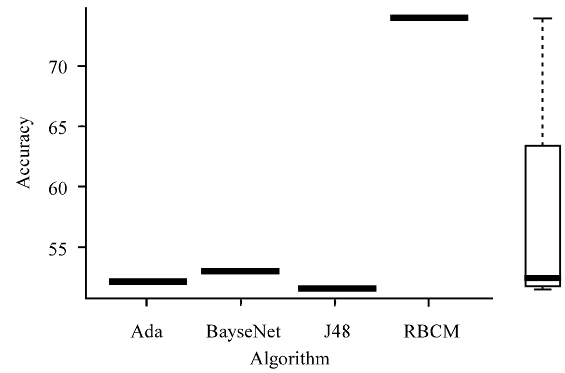

Fig. 1: Accuracy of Youtube dataset

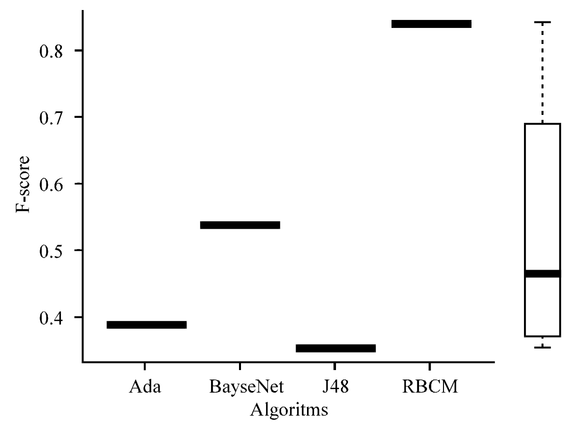

Fig. 2: F-score of Youtube dataset

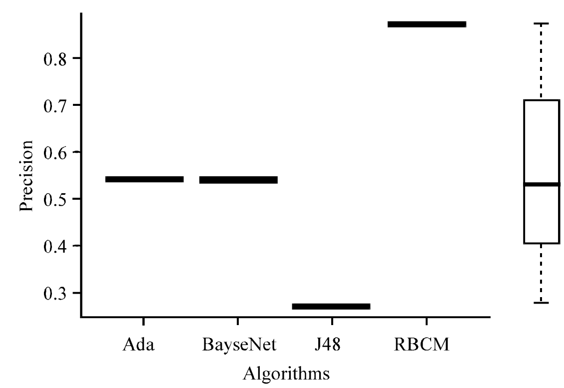

Fig. 3: Precision of Youtube dataset

Comparison with the related work: In our experiments, we were obliged to reduce the number of observations for this Youtube dataset. The reason that competitors approaches could not build their models using 1.7 million observations. RBCM has no such memory problem. RBCM performs significantly better than other algorithms in Youtube dataset. Figure 1 presents a comparison of algorithms in terms of accuracy. Youtube dataset has input features that are based on the complex patterns of alphanumeric values. These features offer a unique challenge for generalization to each of the algorithm. RBCM exploits the complexity of patterns by balancing the trade-off of exploration and exploitation during its simulations. RBCM accuracy, precision F-score and are given in Fig. 1-3, respectively. These figures clearly indicates the success Monte-Carlo sampling techniques for data sets like Youtube. The rival techniques could not generalize the movie recommendation task due to the complex patterns in the input features.

CONCLUSION

RBCM is presented and empirically evaluated using Youtube benchmark. RBCM is based on reinforcement learning algorithm that exploits the capabilities of Monte-Carlo simulations to tune the value function. The algorithm maps the high-dimensional problem space like a movie to a one dimensional contextual representation. The contextual space is used to tune value function for classification. The hybrid of regression and Monte-Carlo simulations strengthen the capabilities of algorithm to generalize the classification task in high dimensional search spaces. The results show that RBCM is easily scalable for large and high dimensional datasets.

RECOMMENDATIONS

In future research, RBCM can be extended to work in collaboration with other learning algorithms in the cases where search space is comprised of low dimensions and input features are dominant by categorical values.

REFERENCES

Abbasnejad, E., S. Sanner, E. Bonilla and P. Poupart, 2013. Learning community-based preferences via Dirichlet process mixtures of gaussian processes. Proceedings of the 23rd AAAI International Joint Conference on Artificial Intelligence, August 3-9, 2013, AAAI Press, Beijing, China,-pp: 1.

Bolchini, C., C.A. Curino, E. Quintarelli, F.A. Schreiber and L. Tanca, 2007. A data-oriented survey of context models. ACM. Sigmod Rec., 36: 19-26.

Catal, C., S. Tufekci, E. Pirmit and G. Kocabag, 2015. On the use of ensemble of classifiers for accelerometer-based activity recognition. Appl. Soft Comput., 37: 1018-1022.

Efraim, T. and K. David, 2003. E-Commerce: Mobile Commerce. Prentice Hall, New Jersey, USA.,.

Haroonabadia, H. and M. Haghifamb, 2011. Generation reliability assessment in power markets using monte carlo simulation and soft computing. Appl. Soft Comput., 11: 5292-5298.

Hmouz, A.A., J. Shen and J. Yan, 2009. A machine learning based framework for adaptive mobile learning. Proceedings of the International Conference on Web-Based Learning, August 19-21, 2009, Springer, Berlin, Germany, ISBN:978-3-642-03425-1, pp: 34-43.

Hong, J.Y., E.H. Suh and S.J. Kim, 2009. Context-aware systems: A literature review and classification. Expert Syst. Appl., 36: 8509-8522.

Huang, G.B., 2014. An insight into extreme learning machines: Random neurons, random features and kernels. Cognit. Comput., 6: 376-390.

Martinenghi, D. and R. Torlone, 2009. Querying context-aware databases. Proceedings of the International Conference on Flexible Query Answering Systems, Vol. 5822, October 26-28, 2009, Springer, Berlin, Germany, ISBN:978-3-642-04956-9, pp: 76-87.

Moro, S., P. Cortez and P. Rita, 2014. A data-driven approach to predict the success of bank telemarketing. Decis. Support Syst., 62: 22-31.

Naveed, M., D. Kitchin, A. Crampton, L. Chrpa and P. Gregory, 2012. A monte-carlo path planner for dynamic and partially observable environments. Proceedings of the IEEE Conference on Computational Intelligence and Games (CIG), September 11-14, 2012, IEEE, Granada, Spain, ISBN:978-1-4673-1193-9, pp: 211-218.

Nguyen, T.V., W. Lim, H .A. Nguyen and D. Choi, 2008. Context awareness framework based on contextual graph. Proceedings of the 5th IFIP International Conference on Wireless and Optical Communications Networks WOCN'08, May 5-7, 2008, IEEE, Surabaya, Indonesia, ISBN:978-1-4244-1979-1, pp: 1-5.

Orsi, G. and L. Tanca, 2011. Context Modelling and Context-Aware Querying. In: Datalog Reloaded, Moor, O.D., G. Gottlob, T. Furche and A. Sellers (Eds.). Springer, Berlin, Germany, pp: 225-244.

Pozveh, M.H., M.A. Nematbakhsh and N. Movahhedinia, 2009. A multidimensional approach for context-aware recommendation in mobile commerce. Intl. J. Comput. Sci. Inf. Secur., 3: 1-6.

Xu, J., H. He and H. Man, 2012. DCPE Co-training for classification. Neurocomputing, 86: 75-85.