Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

Using Contextual Bandits with Behavioral Constraints for Constrained Online Movie Recommendation

Avinash Balakrishnan, Djallel Bouneffouf, Nicholas Mattei, and **Francesca Rossi**
IBM Research
Yorktown Heights, NY 10598
{avinash.bala, djallel.bouneffouf, n.mattei, francesca.rossi2}@ibm.com

Abstract

AI systems that learn through reward feedback about the actions they take are increasingly deployed in domains that have significant impact on our daily life. In many cases the rewards should not be the only guiding criteria, as there are additional constraints and/or priorities imposed by regulations, values, preferences, or ethical principles. We detail a novel online system, based on an extension of the contextual bandits framework, that learns a set of behavioral constraints by observation and uses these constraints as a guide when making decisions in an online setting while still being reactive to reward feedback. In addition, our system can highlight features of the context which are more predicted to be more rewarding and/or are in line with the behavioral constraints. We demonstrate the system by building an interactive interface for an online movie recommendation agent and show that our system is able to act within a set of behavior constraints without significantly degrading overall performance.

1 Overview and Related Work

In online decision settings an agent must select one out of several possible actions, e.g., recommending a movie to a particular user, or proposing a treatment to a patient in a clinical trial. Each of these actions is associated with a context, e.g., a user profile, and feedback, e.g., the reward or rating, is only observed for the chosen option. In these online decision settings the agent must learn the inherent trade-off between exploration, identifying and understanding the reward from an action, and exploitation, gathering as much reward as possible from an action. These decision problems are traditionally modeled for online settings as a *multi-armed bandit (MAB)* problem [Mary *et al.*, 2015; Villar *et al.*, 2015] and is used to solve many real-world problems [Sutton and Barto, 2017; Mnih *et al.*, 2013]. In the MAB setting there are K *arms*, each associated with a fixed but unknown reward probability distribution [Lai and Robbins, 1985; Auer *et al.*, 2002]. At each time step, an agent plays an arm, i.e., recommends an item to a user, and receives a reward that follows the selected arm's probability distribution, independent of the previous actions. In the generalized contextual multi-armed bandit (CMAB) problem the agent observes a d-dimensional *feature vector*, or *context* before making a decision. Over time, the agent learns the relationship between contexts and rewards; LINUCB [Li *et al.*, 2010; Chu *et al.*, 2011] and Contextual Thompson Sampling [Agrawal and Goyal, 2013] are the most successful algorithms for this domain.

We consider cases where the behavior of the online agent may need to be restricted in its choice of arm for a given context by laws, values, preferences, or ethical principles [Sen, 1974]. Constrained or ethical decision making has become popular in CS and AI [Briggs and Scheutz, 2015; Armstrong, 2015] with most of the work focused on teaching a computer system to act within ethical or legal guidelines. We apply a *behavioral constraint* to the agent that is independent of the reward. For instance, a parent or guardian group may want the agent to not recommend certain types of movies to children, even if this recommendation could lead to a high reward. There is recent work on combining contextual bandits with a budget of arm pulls and/or time constraints [Wu *et al.*, 2015; Agrawal and Goyal, 2016] but none use policy constraints. These exogenous behavioral constraints are guidelines that should be followed by the agent and not updated online. Hence, they should either be explicitly given or learnt from examples, before the online agent begins taking actions. In many settings we may not have access to the constraints as an explicit set of rules or function, we may only have access to examples of constrained behavior from a doctor, or a set of decisions made by a parent, e.g., when setting up a new online account.

Contribution. We build an online agent using novel extensions to the CMAB algorithms who is able to learn constraints demonstrated as examples of appropriate actions and apply these constraints in an online recommendation setting while accruing as much reward as possible. For flexibility, we expose a parameter of the algorithm called σ_{online} that allows the system designer to transition between $\sigma_{online} = 0.0$ where the agent is only following the learned constraints and is insensitive to the online reward to $\sigma_{online} = 1.0$ where the agent is only following the online rewards. We demonstrate an example constrained movie recommendation system using real-world data.

2 Behavior Constrained Contextual Bandits

In the standard contextual bandit problem, at each time $t \in \{1, ..., T\}$ the agent is presented with a *context vector* $c(t) \in \mathbf{R}^N$ and must choose an arm $k \in A = \{1, ..., K\}$ to play, observing reward $r_k(t)$ for pulling arm k at time t [Langford and Zhang, 2008]. The purpose of a contextual bandit algorithm is to minimize the cumulative regret $R(T) = \sum_{t=1}^{T} r^*(c(t)) - r_k(c(t))$ where $r^*(c(t))$ is the maximum reward at time t and $r_k(c(t))$ is the actual reward observed. We assume that the expected reward for arm k is a linear function of the context, i.e. $E[r_k(t)|c(t)] = \mu_k^T c(t)$, where μ_k is an unknown weight vector (to be learned from the data) associated with the arm k. Informally μ_k models the agents' belief over what features of the context vector are most rewarding. We build off the popular Contextual Thompson Sampling Algorithm (CTS) [Agrawal and Goyal, 2013].

We introduce a new setting we call *Behavior Constrained Contextual Bandits (BCCB)* and use it to demonstrate a system for online movie recommendation that is able to learn and follow behavioral constraints. We first train the agent during the teaching phase with samples of (c, k), i.e., context and subsequent arm pulled, which are examples of the teacher agent who demonstrates the desired constraints. We use the CTS algorithm to learn a policy μ^e from these examples that captures the constraints we want our agent to follow during the online recommendation phase.

In the recommendation phase the agent needs to maximize both $r_k(t) \in [0, 1]$, the reward of the action k at time t, and $r_k^e(t) \in [0, 1]$, which models whether or not the pulling of arm k violates the behavior constraints. The agent does not observe $r_k^e(t)$ online as the labeler may not be around to always provide this feedback. Our algorithm uses Thompson sampling to estimate the expected rewards of the online policy for each arm μ_k and uses the learned policy μ_k^e to estimate whether or not pulling arm k is consistent with the behavioral constraints. It makes a decision based on a weighted combination of $\mu_k^e(t)$ and $\mu_k(t)$ for each arm using σ_{online} as weight given by the user, this weight balances between following a reward driven policy and constrained policy. The system can display the features of the context vector which are most aligned with either the constraints, i.e., more important in $\mu_k^e(t)$, or most aligned for online reward, i.e., more important in $\mu_k(t)$. The agent then obtains the reward $r_k(t)$ for arm k and updates the parameters of the distribution.

3 Constrained Online Recommendation

To study the effect of imposing exogenous constraints on an online decision making agent and to demonstrate the soundness and flexibility of our techniques we built an agent and interface for online movie recommendation [Mary *et al.*, 2015]. This system learns online what movies to recommend to users but obeys guidelines set by parents during the teaching phase, e.g., young people should not be recommended movies with too much violence, and reward is given when the user reviews the movie. We want the agent to accrue as much reward as possible given that he does stray *too* far from the learned constraints by controlling his behavior using σ_{online}. A screenshot of our interactive system is shown in Figure 1.

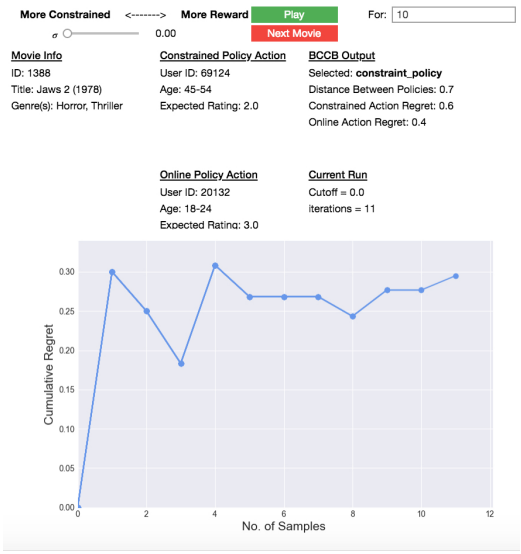

Figure 1: Screenshot of the demonstration system.

Data. We take the top 1000 users and movies from the MovieLense 20m dataset [Harper and Konstan, 2016]. The genre information, which defines the context vector for our agent, is one or more of the categories: Action, Adventure, Comedy, Drama, Fantasy, Horror, Romance, Sci-Fi, Thriller. We impute ages to the users by commonly used advertising demographics: 12-17, 18-24, 25-34, etc. [Jobber and Ellis-Chadwick, 2012] proportionally according to the US Census.

Methodology. To learn the behaviorally constrained policy μ^e we construct a learning environment for the teaching phase using a disjoint subset of movies (contexts), but the same arms (users).We create a *behaviorally constrained reward training set* that is derived from an *behavior constraint matrix*. While in the real world the we assume the behavior constraint matrix does not exist, in order to test our algorithms we use it as a method of generating data. The behavior constraint matrix is a $\{0, 1\}$ matrix representing the frequency we can recommend a movie with genre type d to someone of age type a. We enforce a constraint in the most restricted sense, i.e., if a movie has multiple features then if any feature is restricted the movie is as well. We run the CTS algorithm over a 200 movie subset for variable numbers of examples, modeling a parent who answers a few queries as to what is appropriate. We allow users to add examples to update μ^e. In the recommendation phase we show the movies to the agent and allow the user to adjust σ_{online}. For each movie the system explains its decision by highlighting the features of μ^e and μ_k it thinks most rewarding or constraining.

Discussion. The agent is able to learn a high quality constrained policy described only by examples. The agent is then able to use this policy to guide the actions it takes while learning online a reward-based policy. This agent is able to make decisions that very rarely or never violate the constrained policies, if necessary, and achieves performance on par with an agent that is given the behavioral constraints explicitly. Understanding the interaction between the behavioral constraints and online rewards is important and data dependent.

References

[Agrawal and Goyal, 2013] Shipra Agrawal and Navin Goyal. Thompson sampling for contextual bandits with linear payoffs. In *ICML (3)*, pages 127–135, 2013.

[Agrawal and Goyal, 2016] Shipra Agrawal and Navin Goyal. Linear contextual bandits with knapsacks. In *Proceedings of Advances in Neural Information Processing Systems (NIPS 2016)*, pages 3450–3458, 2016.

[Armstrong, 2015] Stuart Armstrong. Motivated value selection for artificial agents. In *Workshops of the 29th AAAI: AI, Ethics, and Society*, 2015.

[Auer *et al.*, 2002] Peter Auer, Nicolò Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 47(2-3):235–256, 2002.

[Briggs and Scheutz, 2015] Gordon Briggs and Matthias Scheutz. "Sorry, I can't do that": Developing mechanisms to appropriately reject directives in human-robot interactions. In *AAAI Fall Symposium Series: AI for Human-Robot Interaction*, 2015.

[Chu *et al.*, 2011] Wei Chu, Lihong Li, Lev Reyzin, and Robert E. Schapire. Contextual bandits with linear payoff functions. In *Proc. AISTATS*, pages 208–214, 2011.

[Harper and Konstan, 2016] F Maxwell Harper and Joseph A Konstan. The Movielens datasets: History and context. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 5(4):19, 2016.

[Jobber and Ellis-Chadwick, 2012] David Jobber and Fiona Ellis-Chadwick. *Principles and Practice of Marketing*. McGraw-Hill Higher Education, 2012.

[Lai and Robbins, 1985] Tallai Andherbertrobbins Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

[Langford and Zhang, 2008] John Langford and Tong Zhang. The Epoch-Greedy Algorithm for Contextual Multi-armed Bandits. In *Proc. 21st NIPS*, 2008.

[Li *et al.*, 2010] Lihong Li, Wei Chu, John Langford, and Robert E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proc. 19th WWW*, pages 661–670, USA, 2010.

[Mary *et al.*, 2015] Jérémie Mary, Romaric Gaudel, and Philippe Preux. Bandits and recommender systems. In *Machine Learning, Optimization, and Big Data*, pages 325–336, 2015.

[Mnih *et al.*, 2013] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *CoRR*, 2013.

[Sen, 1974] Amartya Sen. *Choice, Ordering and Morality*. Blackwell, Oxford, 1974.

[Sutton and Barto, 2017] Richard S. Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. MIT Press, 2nd edition, 2017.

[Villar *et al.*, 2015] Sofia S. Villar, Jack Bowden, and James Wason. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical Science*, 30(2):199, 2015.

[Wu *et al.*, 2015] Huasen Wu, R. Srikant, Xin Liu, and Chong Jiang. Algorithms with logarithmic or sublinear regret for constrained contextual bandits. In *Advances in Neural Information Processing Systems (NIPS)*, pages 433–441, 2015.