Machine Learning Research
2022; 7(2): 18-23
http://www.sciencepublishinggroup.com/j/mlr
doi: 10.11648/j.mlr.20220702.12
ISSN: 2637-5672 (Print); ISSN: 2637-5680 (Online)

Automatic Indexing of Digital Objects Through Learning from User Data

Clement Leung[1], Yuanxi Li[2]

[1]School of Science and Engineering and Guangdong Provincial Key Laboratory of Future Networks of Intelligence, The Chinese University of Hong Kong, Shenzhen, China

[2]Department of Computer Science, Hong Kong Baptist University, Hong Kong, China

Email address:
clementleung@cuhk.edu.cn (Clement Leung), csyxli@comp.hkbu.edu.hk (Yuanxi Li)

To cite this article:
Clement Leung, Yuanxi Li. Automatic Indexing of Digital Objects Through Learning from User Data. *Machine Learning Research*. Vol. 7, No. 2, 2022, pp. 18-23. doi: 10.11648/j.mlr.20220702.12

Received: December 17, 2022; **Accepted**: January 12, 2023; **Published**: January 31, 2023

Abstract: Digital data objects increasingly take the form of a non-textual nature, and the effective retrieval of these objects using their intrinsic contents largely depends on the underlying indexing mechanism. Since current multimedia objects are created with ever-increasing speed and ease, they often form the bulk of the data contents in large data repositories. In this study, we provide an effective automatic indexing mechanism based on learning reinforcement by systematically exploiting the big data obtained from different user interactions. Such human interaction with the search system is able to encode the human intelligence in assessing the relevance of a data object against user retrieval intentions and expectations. By methodically exploiting the big data and learning from such interactions, we establish an automatic indexing mechanism that allows multimedia data objects to be gradually indexed in the normal course of their usage. The proposed method is especially efficient for the search of multimedia data objects such as music, photographs and movies, where the use of straightforward string-matching algorithms are not applicable. The method also permits the index to respond to change in relation to user feedback, which at the same time avoids the system landing in a local optimum. Through the use of the proposed method, the accuracy of searching and retrieval of multimedia objects and documents may be significantly enhanced.

Keywords: Autonomous Agent, Digital Data Objects, Index Generation, Multimedia Information Search, Probability Generating Function, Reinforcement Learning, Stochastic Modelling

1. Introduction

Data objects increasingly take the form of a non-textual nature, and the effective retrieval of these objects using their intrinsic contents largely depends on the underling indexing mechanism. Since current multimedia objects are created with ever-increasing ease, they often form the bulk of the data contents in large data repositories. The inclusion of users in the information search and retrieval loop improves the overall return [22]. Markov decision process improves the efficiency of locating video frames in a video [23]. The distribution of visual words of multimedia data is probabilistic in relation to the concept relationship formed [24].

Multimedia information retrieval accuracy may be improved using a negative pseudo-relevance feedback approach in the presence of noisy data, and search results may be returned back to the initial retrieval information for refining the search results [19, 26]. Various users allocate the results with scoring metrics. Linear combination of posterior probability is used to refine the search results [25]. Reinforcement learning (RL) approach is suitable for users exposing to raw and high-dimensional information [20]. Instant rewards of the agents improve NDCG in the searching process [21].

In reinforcement learning (RL), an agent learns through the interaction with the dynamic environment to maximize its long-term rewards, in order to act optimally. Most of the time, when modeling real-world problems, the environment involved is non-stationary and noisy [1, 4, 6]. More precisely, the next state results from taking the same action in a specific state may not necessarily be the same but appears to be stochastic [2, 7]. And the exploration strategies adopted in

different categories of RL algorithms provide different levels of control to the exploration of unknown factors, which in turn give various possibilities to the learning results.

As a result, the observed rewards and punishments are often non-deterministic [30, 31, 32]. For example, when one is trying to find a video for cooking a dish, a shortening of the searching time may be regarded as a reward, while a lengthening of the same may be viewed as punishment. Likewise, when one is exploring a new advertising channel, a resultant significant increase in sales may be viewed as a reward, while failure to do so may be regarded as punishment. In situations like these, there are stochastic elements governing the underlying environment. In the new route to work example, whether one receives rewards or punishments depends on a variety of chance factors, such as weather condition, day of the week, and whether there happens to be road works or traffic accidents which may or may not be representative.

Noise in multimedia data is generally numerous and cannot be known or enumerated in a practical sense, and these tend to mask the underlying pattern. Indeed, if stochastic elements are absent, the learning problems involved could be greatly simplified and their presence has motivated early research in the area. As early as 1990s, mainstream research in RL, such as the influential survey assessing existing methods carried out by Kaelbling, *et al.* [2], and the Explicit Explore or Exploit (E^3) Algorithm to solve Markov Decision Process (MDP) in polynomial time [3], adopts the common assumption of a stationary environment within a RL framework. Later on, with further advances in RL, theoretical analyses addressing the concern of non-stationary environment attracted great interests. One of the works by Brafman and Tennenholtz introduces a model-based RL algorithm R-Max to deal with stochastic games [5], and the performance effectiveness of multimedia information search using the epsilon-greedy approach has been exploited in [33]. Such stochastic elements can notably increase the complexity in multi-agent systems and multi-agent tasks, where agents learn to cooperate and compete simultaneously [6, 10]. Autonomous agents are required to learn new behaviors online and predict the behaviors of other agents in multi-agent systems. As other agents adapt and actively adjust their policies, the best policy for each agent would evolve dynamically, giving rise to non-stationarity [8, 9].

In most of the above situations, the cost of a trial or observation to receive either a reward or punishment can be significant, and preferably, one would like to arrive at the correct conclusion by incurring minimum cost. In the case of the advertising example, the cost of advertising can be considerable and one would therefore like to minimize it while acquiring the knowledge whether such advertising channel is effective. Similarly, in RL algorithms, we are always in the hope to rapidly converge to an optimal policy with least volumes of data, calculations, learning iterations, and minimal degree of complexity [11, 12]. To do so, one should explicitly define the stopping rules for specifying the conditions under which learning should terminate and a conclusion drawn as to whether the learning has been successful or not based on the observations so far.

The problem of finding termination conditions, or stopping rules, is an intensive research topic in RL, which is closely linked to the problems of optimal policies and policy convergence [13]. Traditional RL algorithms mainly aim for relatively small-scale problems with finite states and actions. The stopping rules involved are well-defined for each category of algorithms, such as utilizing Bellman Equation in Q-learning [14]. To deal with continuous action spaces or state spaces, new algorithms, such as the Cacla algorithm [15] and CMA-ES algorithm [16], are developed with specific stopping rules. Still, most studies on stopping rules are algorithm-oriented and do not have a unified measurement for general comparison.

In this paper, we present a probabilistic mechanism, which explicitly incorporates the stochastic environment in multimedia information search and retrieval. Section II presents the fundamental model of a predefined general stopping rule. The information search and retrieval success based on the rewards ratio is then studied in Section III. Based on the stochastic model, Section IV analyzes the probability of exceeding cost bounds, and the final conclusions are drawn in Section V.

2. A Stochastic Learning Paradigm

Here, we are dealing with a sequence of iteration feedbacks, and these may represent punishment or reward. The former is referred to as negative feedback, while the latter is referred to as positive feedback. We use the probabilities p and q, with $p + q = 1$, which correspond respectively to those of getting a positive feedback or negative feedback; e.g., for $q < p$, then we would have a successful outcome. An error often committed is that when the first few observations are all negative, one would terminate prematurely and conclude that the multimedia information search and retrieval episode is a failure. Let us consider the stopping criterion:

Criterion A: The process terminates a session after getting s feedbacks which are positive. A session is regarded as successful when the number of positive feedbacks is acceptably more than the number of negative feedbacks. A session is regarded as unsuccessful when the number of negative feedbacks is acceptably more than the number of positive feedbacks.

Let us examine the stochastic implications of *Criterion A*. We denote by X the random count of feedbacks before the first positive feedback is received; thus

$$\text{Prob}[X = n] = pq^n, n = 0, 1, 2, 3, \ldots \quad (1)$$

The probability generating function $F(z)$ of X is given by

$$F(z) = \sum_{n=0}^{\infty} \Pr[X = n]\, z^n$$
$$= p \sum_{n=0}^{\infty} q^n z^n = \frac{p}{(1-qz)}. \quad (2)$$

We observe that this is a regenerative process, where the

sequence replicates itself stochastically, so that the number of feedbacks N_s in order to reach s positive feedbacks is,

$$N_s = \sum_{n=1}^{s} X_n, \quad (3)$$

where each X_n has the same stochastic property as X. From [17], the probability generating function of $F_s(z)$ corresponding to N_s may be arrived at by multiplication of the underlying probability generating functions for $s = 1$,

$$F_s(z) = F_1(z)^s = [\frac{p}{(1-qz)}]^s. \quad (4)$$

The statistical properties of N_s may derived easily by differentiating the probability generating function and substituting the argument $z = 1$,

$$\mathrm{E}[N_s] = F_s'(1) = \frac{sq}{p}, \quad (5)$$

$$\mathrm{Variance}[N_s] = F_s''(1) + F_s'(1) - F_s'(1)^2 = \frac{sq}{p^2}. \quad (6)$$

Furthermore, the probability mass function Prob[$X_s = n$] may be derived by undertaking a binomial expansion of equation (4),

$$\mathrm{Prob}[N_s = n] = \binom{-s}{n} p^s (-q)^n, n = 0, 1, \ldots \quad (7)$$

Since N_s is the sum of independent identically distributed random variables, when s is appreciable, it may be approximated by the normal variate by virtue of the Central Limit Theorem [17], so that

$$N_s \sim Gaussian\left(\frac{sq}{p}, \frac{sq}{p^2}\right), \quad (8)$$

where $Gaussian(\mu, \sigma^2)$ denotes the Gaussian distribution with mean μ and variance σ^2. Thus, the probability Pr[$W_r > b$] may be approximated by

$$\mathrm{Prob}[N_s > a] = \int_{\frac{ap-sq}{\sqrt{sq}}}^{\infty} \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}} dt = 1 - \Phi\left(\frac{ap-sq}{\sqrt{sq}}\right), \quad (9)$$

where Φ is the standard Gaussian distribution function having zero mean and unit variance.

3. The Negative Feedback Quotient in a Competitive Framework

We compare the number of the two types of feedbacks, and denote it by ω; i.e.,

$$\omega = \frac{Number\ of\ negative\ feedbacks}{Number\ of\ positive\ feedbacks}.$$

Thus, from this, and making use of the expectation in equation (5), we have,

$$\omega = \frac{E[N_s]}{s} = \frac{1-p}{p}. \quad (10)$$

We shall refer to this as the *negative feedback quotient*. We see from [18] that this can be viewed as the odds of getting a negative feedback.

In this case, we may view the present situation as a competition between an agent and its adversary. The agent would win a point with probability p, while the adversary would win a point with probability q. For a particular experiment, the odds of the adversary winning is given by ω, while the odds of the agent winning is given by 1/ω. It is interesting to determine the optimal strategy for the agent in order to maximize the gain if each experiment carries a stake of one unit. Obviously, the situation is favorable to the agent for ω < 1, and the situation is unfavorable to the agent for ω > 1. The former indicates that the probability of winning is less than the probability of losing for the agent, while the latter indicates that the probability of winning is less than the probability of losing.

In an autonomous agent-based context [27, 28, 29], this may be viewed as a competition as indicated above between the agent and its adversary over a sequence of experiments, the question arises what is the best strategy the agent should adopt in terms of the amount to bet assuming the agent has a total capital of C units, and given that ω < 1. Since ω < 1 implies $q < p$, and so $p > ½$, a simple-minded strategy to maximize gain in each experiment is to bet the full capital in each experiment. While this strategy of betting the full amount seems optimal if the number of experiments is limited to just one, it is far from being so in the long run. This is because by the law of large numbers, given sufficient time, negative feedback is bound to occur, in which case the full capital would be loss, so that the long-term gain of the agent would be zero even in a situation that

$$\omega \approx 0.$$

In a learning context, the capital C corresponds to a situation where the agent is able to sustain a total of C negative feedbacks from the start of the process.

Thus, even in a situation which is highly favorable to the agent, using the expectation maximization will not be optimal. From [34], the amount of capital K to place on a particular experiment should be

$$K = \frac{C[ph-(1-p)]}{h}.$$

where h is the per unit gain upon a win.

Viewing the problem from a different perspective, we shall estimate p from the observed negative feedback quotient N/s and to derive an estimate. From equation for ω, and expressing p in terms of the quotient, we have

$$\hat{p} = \frac{1}{1+N/s},$$

and we see that $p > 1/2$ if $N/s < 1$. Another way to the determination of p using interval estimates will not be pursued here and will form a different consideration.

4. Meeting Constraints

The average, however, is often not sufficient as it fails to fully reflect any statistical fluctuations. In many cases, as in

the advertising example, the cost of observation is significant. Let c be the numerical representation of cost associated with an observation. Having specified r, a minimum observation cost of rc must therefore be incurred. What is uncertain is the number of negative feedbacks obtained, and ideally in order to attain the lowest cost, this number should be limited. If we allow up to a cost of bc for observing b negative feedbacks, we can determine the chance P_b that the cost of learning for this element to go above the limit. From (7) above, this is given by

$$P_b = 1 - \sum_{j=0}^{b} \Pr[N_s = j] = 1 - \sum_{j=0}^{b} \binom{-s}{k} s(-q)^j. \quad (11)$$

The computation associated with (11) is somewhat laborious. As indicated above, when the value of r is large, we can make use of the normal approximation of (9). In many RL learning episodes, r tends to be under 100, as a lengthy iteration time is not feasible and most learning algorithms aim to converge in minimum time.

Clearly, the selection of the maximum cost weight b will have a significant impact on P_b. Very often, it is more meaningful to relate b to E[N_s] either additively or multiplicatively. Table 1 tabulates the values of P_b for different values of b. The first part of Table 1 considers b by adding a fixed value d, with d = 5 and d = 10, while the second part considers b by multiplying by a fixed multiple α, with α = 1.2 and α = 1.5; here, b is rounded to the nearest integer. In the first part of Table 1, we see that for either value of r, when p is appreciably greater than q, the probability of exceeding cost bounds tends to be acceptably small, and this is especially so for r = 20. The reason is that, since d is a fixed value, its relative contribution to b increases as p increases, produces a relatively large cost bound weight compared to the average one, and accordingly lowers the probability of exceeding the bound. However, in the second part of Table 1, the difference between E[N_s] and b decreases as E[N_s] decreases, so that P_b tends to be large for higher values of p.

Table 1. *The variation of P_b for different values of b.*

b	r	p	q	E[N_s]	b	P_b	P_b'	Err
		0.5	0.5	20.00	25	0.215	0.186	0.029
	20	0.8	0.2	5.00	10	0.023	0.026	0.003
b = E[N_s] + d		0.9	0.1	2.22	7	0.001	0.004	0.003
(d = 5)		0.5	0.5	50.00	55	0.309	0.279	0.030
	50	0.8	0.2	12.50	17	0.127	0.108	0.019
		0.9	0.1	05.56	11	0.014	0.017	0.003
		0.5	0.5	20.00	30	0.057	0.059	0.002
	20	0.8	0.2	5.00	15	0.000	0.001	0.001
b = E[N_s] + d		0.9	0.1	2.22	12	0.000	0.000	0.000
(d = 10)		0.5	0.5	50.00	60	0.159	0.147	0.012
	50	0.8	0.2	12.50	22	0.008	0.011	0.003
		0.9	0.1	05.56	16	0.000	0.000	0.000
		0.5	0.5	20.00	24	0.264	0.226	0.038
	20	0.8	0.2	5.00	6	0.345	0.253	0.092
b = αE[N_s]		0.9	0.1	2.22	2	0.556	0.380	0.176
(α = 1.2)		0.5	0.5	50.00	50	0.159	0.147	0.012
	50	0.8	0.2	12.50	15	0.264	0.215	0.049
		0.9	0.1	05.56	7	0.280	0.207	0.073
		0.5	0.5	20.00	30	0.057	0.059	0.002
	20	0.8	0.2	5.00	7	0.212	0.156	0.056
b = αE[N_s]		0.9	0.1	2.22	3	0.310	0.193	0.117
(α = 1.5)		0.5	0.5	50.00	75	0.006	0.010	0.004
	50	0.8	0.2	12.50	19	0.050	0.048	0.002
		0.9	0.1	05.56	8	0.163	0.121	0.042

In Table 1, column P_b' gives the exact calculation using (11), while column P_b employs the normal approximation using (9). The absolute error between the exact calculation and the normal approximation is given by column *Err*. We see that the normal approximation is quite acceptable in most cases with absolute error less than 0.1. Note that no matter whether having b additively or multiplicatively related to E[W_r], a higher value of d or α always gives smaller absolute error. We therefore suggest that the approximation should only be used when all of r, d and α are sufficiently large.

5. Conclusion and Future Work

Data objects increasingly take the form of a non-textual nature, and the effective retrieval of these objects using their intrinsic contents largely depends on the underling indexing mechanism. Since current multimedia objects are created with ever-increasing ease, they often form the bulk of the data contents in large data repositories. Moreover, the operating environments in which multimedia information is deployed are frequently noisy and probabilistic, and the use of stochastic models in learning is thus useful and effective.

In the present study, we examine a scenario where the total positive feedbacks required is given, which constitute the criterion for terminating the process. By considering the positive to negative feedback quotient, a decision of either success or failure of the process may be obtained. In addition, each experiments also attracts a cost and this is also taken into

consideration.

We also examine a competitive framework where the negative and positive feedbacks are handed out by the user and its adversary Therefore, the eventual conclusion is viewed as a competitive game, and the concluding condition is governed by the manner in which the competition is won. The chances of success and failure have been derived. Closed-form expressions of other relevant measures of interest are derived. So far we have made use of the stochastically independence framework. Other frameworks which eliminate this restriction such as Markovian dependence should be useful in extending the model.

Acknowledgements

This research was supported in part by the Shenzhen Research Institute of Big Data and the Guangdong Provincial Key Laboratory of Future Networks of Intelligence.

References

[1] B. D. Ziebart, A. L. Maas, J. A. Bagnell, and A. K. Dey, "Maximum Entropy Inverse Reinforcement Learning," *Proc. Twenty-Third AAAI Conference on Artificial Intelligence (AAAI 08)*, vol. 8, pp. 1433-1438, 2008.

[2] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey" *Journal of artificial intelligence research*, vol. 4, pp. 237-285, 1996.

[3] M. Kearns, and S. Singh, "Near-optimal reinforcement learning in polynomial time," In *Int. Conf. on Machine Learning*, 1998.

[4] H. Santana, G. Ramalho, V. Corruble, and B. Ratitch, "Multi-agent patrolling with reinforcement learning," *Proc. Third International Joint Conference on Autonomous Agents and Multiagent Systems*, vol. 3, pp. 1122-1129, IEEE Computer Society, 2004.

[5] R. I. Brafman, and M. Tennenholtz, "R-max-a general polynomial time algorithm for near-optimal reinforcement learning," *Journal of Machine Learning Research*, vol. 3, pp. 213-231, 2002.

[6] L. Panait, and S. Luke, "Cooperative multi-agent learning: The state of the art," *Autonomous agents and multi-agent systems*, vol. 11, no. 3, pp. 387-434, 2005.

[7] E. Ipek, O. Mutlu, J. F. Martínez, and R. Caruana, "Self-optimizing memory controllers: A reinforcement learning approach," *ACM SIGARCH Computer Architecture News,* vol. 36, no. 3, IEEE Computer Society, 2008.

[8] L. Busoniu, R. Babuska, and B. De Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Transactions on Systems, Man, And Cybernetics-Part C: Applications and Reviews*, vol. 38, no. 2 2, 2008.

[9] S. V. Albrecht, and P. Stone, "Autonomous agents modelling other agents: A comprehensive survey and open problems," *Artificial Intelligence* 258, pp. 66-95, 2018.

[10] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, and R. Vicente, "Multiagent cooperation and competition with deep reinforcement learning," *PloS one*, vol. 12, no. 4: e0172395, 2017.

[11] A. W. Moore, and C. G. Atkeson, "Prioritized sweeping: Reinforcement learning with less data and less time," *Machine learning*, vol. 13, no. 1, pp. 103-130, 1993.

[12] E. Brochu, V. M. Cora, and N. De Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning,", unpublished.

[13] Q. Wei, F. L. Lewis, Q. Sun, P. Yan, and R. Song, "Discrete-time deterministic Q-learning: A novel convergence analysis," *IEEE transactions on cybernetics*, vol. 47, no. 5, pp. 1224-1237, 2017.

[14] C. J. Watkins, and P. Dayan, "Q-learning," Machine learning 8.3-4 pp. 279-292, 1992.

[15] H. Van Hasselt, and M. A. Wiering, "Using continuous action spaces to solve discrete problems," *Proc. International Joint Conference on Neural Networks (IJCNN 09)*, pp. 1149-1156. IEEE, 2009.

[16] N. Hansen, S. D. Müller, and P. Koumoutsakos, "Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES)," *Evolutionary computation*, vol. 11, no. 1 pp. 1-18, 2003.

[17] W. Feller, *An Introduction to Probability Theory and its Applications*, vol. 1, 3rd Edition, Wiley & Sons, 1968.

[18] S. Ross, *A First Course in Probability*, 9th Edition, Pearson, 2014.

[19] R. Gupta, M. Khomami Abadi, J. A. Cárdenes Cabré, F. Morreale, T. H. Falk, and N. Sebe, "A quality adaptive multimodal affect recognition system for user-centric multimedia indexing". In: Proceedings of the 2016 ACM on international conference on multimedia retrieval. ACM, p. 317-320, 2016.

[20] Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A.,"Deep reinforcement learning: A brief survey," *IEEE Signal Processing Magazine*, 34 (6), 26-38, 2017.

[21] X. Yao, J. Du, N. Zhou, and C. Chen, "Microblog Search Based on Deep Reinforcement Learning," *In Proceedings of 2018 Chinese Intelligent Systems Conference* (pp. 23-32). Springer, Singapore, 2019.

[22] Y. C. Wu, T. H. Lin, Y. D. Chen, H. Y Lee, and L. S. Lee, "Interactive spoken content retrieval by deep reinforcement learning". arXiv preprint arXiv: 1609.05234, 2016.

[23] S. Lan, R. Panda, Q. Zhu, and A. K. Roy-Chowdhury, "FFNet: Video fast-forwarding via reinforcement learning", *In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition,* pp. 6771-6780, 2018.

[24] R. Hong, Y. Yang, M. Wang, and X. S. Hua, "Learning visual semantic relationships for efficient visual retrieval", *IEEE Transactions on Big Data*, 1 (4), 152-161, 2015.

[25] R. Yan, A. Hauptmann, and R. Jin, "Multimedia search with pseudo-relevance feedback. In International Conference on Image and Video Retrieval" (pp. 238-247). *Springer*, Berlin, Heidelberg, 2003.

[26] J. Deng, C. H. C. Leung: Dynamic Time Warping for Music Retrieval Using Time Series Modeling of Musical Emotions. *IEEE Transactions on Affective Computing*, Vol. 6, No. 2, pp. 137-151 (2015).

[27] H. L. Zhang, C. H. C. Leung, G. K. Raikundalia: Topological analysis of AOCD-based agent networks and experimental results. *Journal of Computer and System Sciences*, pp. 255–278, (2008).

[28] Azzam, I., Leung, C. H. C., Horwood, J.: Implicit concept-based image indexing and retrieval. In *Proceedings of the IEEE International Conference on Multi-media Modeling*, pp. 354-359, Brisbane, Australia (2004).

[29] H. Zhang, C. H. C. Leung and G. K. Raikundalia: Classification of intelligent agent network topologies and a new topological description language for agent networks. In *Proceedings of the 4th International Conference on Intelligent Information Processing*, Adelaide, Australia, pp. 21-31 (2006).

[30] N. L. J. Kuang, C. H. C. Leung, and V. Sung: Stochastic Reinforcement Learning. In *Proc. IEEE International Conference on Artificial Intelligence and Knowledge Engineering*, pp. 244-248, California, USA (2018).

[31] N. L. J. Kuang, and C. H. C. Leung: Performance Dynamics and Termination Errors in Reinforcement Learning – A Unifying Perspective. In *Proc. IEEE International Conference on Artificial Intelligence and Knowledge Engineering*, pp. 129-133, California, USA (2018).

[32] N. L. J. Kuang, C. H. C. Leung, Analysis of Evolutionary Behavior in Self-Learning Media Search Engines, *in Proceedings of the IEEE International Conference on Big Data*, Los Angeles, USA, (2019).

[33] N. L. J. Kuang, C. H. C. Leung, Performance Effectiveness of Multimedia Information Search Using the Epsilon-Greedy Algorithm, in *Proceedings of the 2019 IEEE International Conference on Machine Learning and Applications*, pp. 929-936, Florida, USA (2019).

[34] E. Thorpe, *Portfolio Choice and the Kelly Criterion*, Academic Press, 1975.