HKUST SPD - INSTITUTIONAL REPOSITORY

Title	HRCR: Hidden Markov-based Reinforcement to Reduce Churn in Question Answering Forums
Authors	Hadi Mogavi, Reza; Gujar, Sujit; Ma, Xiaojuan; Hui, Pan
Source	Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), v. 11670 LNAI, 2019, p. 364-376
Version	Accepted Version
DOI	10.1007/978-3-030-29908-8_29
Publisher	Springer
Copyright	© Springer Nature Switzerland AG 2019
License	The final publication is available at Springer via https://doi.org/10.1007/978-3-030-29908-8_29

This version is available at HKUST SPD - Institutional Repository (<https://repository.ust.hk>)

If it is the author's pre-published version, changes introduced as a result of publishing processes such as copy-editing and formatting may not be reflected in this document. For a definitive version of this work, please refer to the published version.

HRCR: Hidden Markov-based Reinforcement to Reduce Churn in Question Answering Forums

Reza Hadi Mogavi¹, Sujit Gujar², Xiaojuan Ma¹, and Pan Hui^{1,3}

¹ Hong Kong University of Science and Technology, Hong Kong
`{rhadimogavi,mxj,panhui}@cse.ust.hk`

² International Institute of Information Technology, Hyderabad, India
`sujit.gujar@iiit.ac.in`

³ University of Helsinki, Finland

Abstract. The high rate of *churning users* who abandon the Community Question Answering forums (CQAs) may be one of the crucial issues that hinder their development. More personalized question recommendation to users might help to manage this problem better. In this paper, we propose a new algorithm (we name HRCR) that recommends questions to users such *to reduce their churning probability*. We present our algorithm in a two-fold structure: First, we use Hidden Markov Models (HMMs) to uncover the users' engagement states inside a CQA. Second, we apply a Reinforcement Learning Model (RL) to recommend users the questions that match better with their engagement mood and thus help them get into a better engagement state (the one with the least churning probability). Experiments on a large-scale offline dataset from Stack Overflow show a meaningful reduction in the churning probability of the users who comply with HRCR's question recommendations.

Keywords: Question Answering forum · Churn · Flow theory · Computational user engagement · Reinforcement learning

1 Introduction

Community Question Answering forums (CQAs) like Stack Overflow⁴ facilitate knowledge sharing online [23]. CQAs are dependent on continuous user participation to preserve their sustainability [10, 16]. Especially, retention of *contributing users* who provide answers to questions has priority and is challenging [13, 15]. Related literature suggests *personalizing tasks* before crowdsourcing can grow users' willingness to make more and better quality contributions [25, 21].

In this paper, we introduce a new question recommendation algorithm (we name HRCR) that aims to reduce churning (see [16, 11]) of the contributing users through recommending more personalized questions. While literature is rich in question recommendation algorithms (see [9, 4, 18]), HRCR is distinctive in following aspects: First, HRCR considers churn tendency of contributing users

⁴ <https://stackoverflow.com/>

Fig. 1: High-level illustration of the HRCR. HRCR assigns questions to users. If the assigned questions lead users to a higher engagement state (i.e., Flow Zone), HRCR receives a positive reinforcement (feedback); otherwise, the agent receives a negative reinforcement. Through iterations, HRCR finds the best policy for assigning questions to users.

before recommending questions. Second, HRCR is inspired by the psychological theory of *Flow* (see [8]) to increase user engagement.

HRCR, in essence, is a Markov Decision Process (MDP) based recommendation system. Although the idea of using MDP to build recommender systems is not new (see [6, 20]), we are first to use them for building a more personalized question distributing algorithm. Figure 1 shows a high-level idea of how HRCR is working. After HRCR assigns a question to a user, it receives a positive reinforcement (feedback) if the users’ engagement state is high (i.e., Flow Zone); otherwise, it receives a negative reinforcement. Our goal is to fill the gap of the existing question recommendation algorithms in term of considering churn.

More precisely, we first use Hidden Markov Models to uncover the *stochastic behavioral pattern* behind user participation. We are particularly inspired by the psychological theory of Flow to interpret the hidden states of our HMM. We then use the resolved hidden states of our HMM to reinforce a standard Markov Decision Process model. We make several simplifying assumptions to avoid the otherwise prohibitive number of parameters in our HRCR formulation. Through iterations on the MDP, HRCR finds the best recommendations for users. Experiments on a real-world dataset from a well-known CQA forum shows that HRCR can meaningfully help in churn reduction of the contributing users while preserving its simplicity.

2 Users’ Action Choices

Users often have their own set of principles for deciding what questions they answer. Although knowledgeability in the field comes as an important factor, there are often more parameters involved. Table 1 provides a reference to such parameters. We use Stack Overflow, to characterize the CQA users [18, 16]. However,

Table 1: Reference of possible action choices

Factors	Action Choices
Questioner’s Reputation	Low, moderate, high
Question’s Recency	Archaic, recent
Question’s Score	Low, high
State of Having Answers	With or without a prior answer
State of Acceptance	With or without an accepted contribution
Familiarity with Topic	Familiar or unfamiliar with the asked topic
Bounty Prize	With or without a bounty prize

the features should be similar on the CQAs like Yahoo! Answers and Quora. In the rest, we elaborate on our list of factors.

Questioner’s Reputation (QR). A contributing user can be selective on whose question she answers. For example, a contributing user who has got a lower reputation score might not have the sufficient skill or knowledge to answer to a question asked by a highly reputed user. We use, first and third quartiles of the user reputation values to define low, moderate and highly reputed users. Any user with a reputation between 26 and 1,580 is assumed to be a moderately reputed user. Reputation scores below 26 and above 1,580 are labeled as low and high reputation values respectively.

Question’s Recency (QRC). CQAs like Stack Overflow are designed in a way to give a higher priority to the questions submitted most recently. Our dataset shows that the average response time for a typical question in the CQA of Stack Overflow is 6.23 days. In this paper, we name a question as recent if its time gap from being asked to the time that a user wants to contribute to it does not exceed seven days; otherwise, it is labeled as an archaic question.

Question Score (QS). It is routine for CQAs to ask users to express their opinions about the quality of a post. The CQA of Stack Overflow provides a voting system (up-votes or down-votes) to its users, to collect their opinions. A question’s score is a measure which is found by the deduction of a question’s number of down-votes from its number of up-votes. This measure helps a contributor to skip poor quality questions. We label a question low-scoring if its score is ≤ 1 ; otherwise, it is labeled a high-scoring question.

State of Having Answers (SA). Some of the CQA users waste a great deal of time to contribute to the questions which are already answered. Prior answers to a question are more likely to be appreciated (through receiving up-votes or being selected as the accepted answers) by the questioner [2]. A CQA user should select from contributing to a question with or without another answer.

State of Acceptance (SAC). It is an uneasy decision for many CQA users to decide whether to contribute to an already answered question or not. Some of the CQAs provide their questioners with a flag option to inform the others if they do not need a further contribution to their questions. In the CQA of

Stack Overflow, if the questioner is satisfied by one of the answers received, she can flag that answer as an accepted answer to inform the other contributors. By considering this flag option, a user can decide to make a further contribution to that question or not.

Familiarity with Topic (FT). In many of the CQAs, users are demanded to choose related tags (or keywords) to summarize their post briefly. All of the tags that a user has created or is related to are stored in the user’s profile. For a contributing user, a question is labeled as familiar if there is at least one common tag between the user’s profile and that question; otherwise, we label that question as an unfamiliar question.

Bounty Prize (BP). A bounty is a special prize which is made up of the reputation scores. In a CQA like Stack Overflow, a questioner can offer a share of its reputation to the best answer contributor of a question. The questions which have bounties are typically more challenging than ordinary questions.

3 HRCR Architecture

HRCR is a question recommendation algorithm with the focus on the churn reduction of the contributing users. HRCR uses a two-fold structure to decide which questions to recommend users first. HRCR first trains a Hidden Markov Model (HMM) to elicit the engagement state of the users. The resolved engagement states of the HMM comply generally with the notion of the Flow theory in psychology. According to the Flow theory, users are highly engaged with their experiences if the challenges they face are in-line with their level of skills [8].

Next, we use a Reinforcement Learning Model (RL) to recommend users the action choices that increase the probability of getting into the Flow Zone with respect to the trained probabilities of the HMM. This way, our HMM and RL models complement each other: HMM by providing the engagement states of the users, and RL by using the HMM predictions for each contribution to update the value of the agent’s reward function. We show that the action choices that put the users in the Flow Zone can meaningfully decline the churning probability of the users.

4 Implementation Details

Data Statistics. We use the public access data dump of Stack Overflow in Archive⁵ to build our target dataset. Our target data is enclosed within a range from January 1st, 2014 to January 1st, 2017. The target data includes 12,390 contributing users. We have randomly excluded 3,810 users for testing purposes and the remaining data (8,580 users) are applied for training purposes. Table 2 summarizes the important features of our dataset.

Data Preprocessing. In order to study churning behavior of the contributing users, we need a ground truth dataset of churned and non-churned users. Towards this end, we define churned users similar to [11]. We assume that a user

⁵ <https://archive.org/details/stackexchange>

Table 2: Data statistics of the CQA of Stack Overflow

Measured Statistics	Value
Average contribution of a user	6.93
Variance of user contributions	2.13
Average reputation of a user	481.81
Variance of user reputations	17.52
Total number of users	12,390
Number of churned users	4,821

Fig. 2: A user is labeled as churned if her average participation rate within the inspection period is less than 20% of her prior participation rate in the observation period.

has churned if her average rate of participation (of all measurable activities) in an inspection period (which is the immediate 180 days following the observation period) drops to less than 20% of her average participation rate in a prior time that we call the observation period (which is the immediate 180 days after a user's first contribution). Figure 2 is an illustration of the observation and inspection time periods.

Furthermore, we have carried out the following data preprocessing procedures to eliminate the unwanted consequences of noise: 1) The majority of the users of Stack Overflow are ephemeral users and thus churn the CQA before making enough contributions [16]. We thus remove the users who make no contributions and less than 5 posts in sum, from our study. 2) Users with a lower reputation measure (below 10) are sifted out of this study to resolve the problem of imbalanced class labels.

Methodologies. The major application of *Hidden Markov Models* (HMMs) is to render the sequence of observations that have an underlying stochastic process [17]. The HMM is formally defined with a tuple λ as follows:

$$\lambda = (S_{\text{HMM}}, V_{\text{HMM}}, E, F, \pi) \quad (1)$$

$S_{\text{HMM}} = \{s_1, s_2, \dots, s_N\}$ is a set of N individual states which are hidden. $V_{\text{HMM}} = \{v_1, v_2, \dots, v_M\}$ is a set comprised of M possible observations which are measured directly for a state. E is an $N \times N$ matrix to show the transition probabilities

of the states. Entries of matrix E are derived from the following equation.

$$E = \{e_{ij}\}; \quad e_{ij} = P(q_{t+1} = s_j | q_t = s_i) \quad (2)$$

Equation 2 shows the transition probability between states i and j . The variables of q_t and q_{t+1} show the hidden states before and after a new contribution is made respectively.

F is the emission probability of an observation k being generated from a hidden state j . Observation likelihood, F , is formally defined as follows:

$$F = \{f_j(k)\}; \quad f_j(k) = P(v_k | s_j) \quad (3)$$

Finally, π is to show the probability of the initial states. In this research, we stick to the assumption that all of the hidden states have the same probability at the beginning. The HMM is learnt if only all of the parameters of λ are resolved.

Reinforcement Learning Models (RLs) are applied where an agent attempts to learn the best action policy by interacting with and receiving feedback from its stochastic environment [19]. First, the agent observes its state and then executes an action (or series of actions) which lead to another subsequent state. After getting to the next state, the agent assesses the value of its action according to a *reward function*. The agent considers the punishments and gains it has incurred over time to enhance its action policies iteratively [19]. The RL is formally defined with a tuple v as follows:

$$v = (S_{RL}, V_{RL}, R) \quad (4)$$

S_{RL} refers to the state space of the model and V_{RL} shows the set of all possible actions an RL agent can perform [19]. The feedback function, R , is used to guide an agent toward its goal. The goal of RL agents is to maximize their reward gains within a certain time period. In this research, we use the engagement states inspired by the Flow theory to define R and to recommend users the actions that put them into the Flow Zone.

5 Extraction of User Engagement State

Setup. Inspired by the flow theory measures, we use the user's skill and challenge levels to find out the user's engagement state. We modify a measure by [24], called Z-score to determine user skills in the CQA of Stack Overflow.

$$Z_{mod} = \frac{\alpha - \beta}{\sqrt{\alpha + \beta}} \quad (5)$$

Equation 5 shows the modified Z-score. The variables of α and β represent the total number of accepted answers, and the questions asked, respectively. The feature of competitiveness by [16], provided us with a basis to develop the measure of the perceived challenge of users through Equation 6.

$$Challenge = \frac{\gamma}{\prod_{n=1}^{\gamma} Rank(c_n)} \quad (6)$$

Fig. 3: Marginal skill and challenge distribution of the churned and non-churned

The variables of γ and c_n refer to the total number of answers and the n th contribution of a user, respectively. $Rank(\cdot)$ is a function which returns the total number of contributions received for a *question* divided by the user's position within the contributions (sorted by the answer score). Although more parameters should be included to get a precise measurement of user skill and challenge, we stick to the introduced parameters to avoid complexity. Besides, our skill and challenge plots of churned and non-churned users comply with the psychological implications of the Flow theory. Figures 3a and 3b show the skill and challenge distributions of the churned and non-churned users respectively. As is shown, users with higher levels of skill and challenge are less likely to churn. There also exists a Pearson correlation of 0.163 with the significance of $P = 0.01$, between the user skill and challenge measures. This implies that the users' perceived level of challenge increases as users' level of skill increases.

We further conduct a Brown-Forsythe test [3] to find if there is a significant difference between skill and challenge levels of the users with different churning attitudes. The results show that the variations of user skill ($P = 0.05$) and challenge ($P = 0.05$) are statistically significant for the churned and non-churned users as we have hypothesized.

Inference of States. We train an HMM λ to infer the engagement state of the users. We feed the input of HMM with the skill and challenge measures of users. As true number of hidden states is typically determined by the Bayesian Information Criterion (BIC) and Akaike Information Criterion (AIC) (see [5, 1, 12]), we exhaustively search for the best AIC and BIC measures among two to four hidden states. As Figure 4a shows, the best number of hidden states returned is four which complies with the simple quadrant framework of the Flow theory. Thereby, we name the hidden states of HMM to be $S_{HMM} = \{\text{Apathy, Anxiety, Boredom, Flow Zone}\}$. We use the Forward-Backward Algorithm by [17] to resolve the entries of the matrices E and F . After running 100 iterations of the algorithm with three different random starts, the Expectation Maximization value converges in the 23rd iteration as Figure 4b implies. We show the transition probabilities between the hidden states in Figure 4c. As

Fig. 4: (a) BIC and AIC measures (b) Convergence of HMM (c) Transition probabilities (d) Churning probabilities

is shown, the probabilities of the self-loops in all of the states are below half; thus, the users are more likely to change their states over time. However, among all of the hidden states, the state of Flow Zone has the largest self-loop value. This means that users are more likely to stay in the Flow Zone once they are in it. We notice that the users who are in the Apathy state are not as likely as the users in the Anxiety and Boredom states to enter the Flow Zone. Furthermore, as Figure 4d depicts, the users who are in the Flow Zone are also less likely to churn the CQA. This makes the Flow Zone a perfect destination to steer users' behaviors to. We exploit this knowledge to devise our Reward function of RL.

Contrary to the Flow Zone, the Apathy state has the largest probability of churning. The Apathy state users are mostly new-comers to the CQA. The users who are in this state are mainly susceptible to entering to the Anxiety state.

We show the emission distributions F , for the skill and challenge measures, in Figure 5. As is shown, the majority of the users have skills higher than zero, and the perceived challenges are relatively higher in the Flow Zone and the Anxiety state in comparison with the Apathy and Boredom states.

Fig. 5: Continuous emission distribution of skill and challenge measures per each engagement state

In the rest of this research, we use our built HMM to predict the engagement state of the users after each contribution.

6 Recommending the Action Choices

Setup. From all possible combinations between the action choices of Table 1, we can get 144 types of authentic actions for a user to choose from that form our V_{RL} set. We use *1-of-k coding* to model the states of S_{RL} [7]. Each state of S_{RL} is a 144-dimensional column vector that has only one of its entries equal to 1 and the remaining entries are zeros. If we think of a user as an agent, the long-term goal would be to choose among the action choices in such a way to maximize its expected reward value over time. Equation 7 shows the discounted reward function that we use to train the RL.

$$Reward = \sum_{l=1}^{\infty} \delta P_{HMM}(FlowZone|.) \quad (7)$$

P_{HMM} is the probability of getting into the Flow Zone (after making l th contribution) given the user's current satisfaction status. δ shows the discount factor which is a number between zero and 1 that shows the difference of importance between future and present rewards [19]. In this study, δ is set to 0.4. The inspiration behind our reward function is that as users choose action choices

which increase the probability of getting into the Flow Zone, their probability of churning should drop.

Best Action Policy. We assume that an agent κ chooses from the authentic actions V_{RL} in contribution t so as to get a higher payoff. We recommend this choice of action with an RL as follows. We start with the Bellman’s optimality equation [19], and initiate the Q-value of the agent κ ($Q_{t=0}^\kappa(v_{RL}, s_{RL})$) at $t = 0$ to a positive constant Q_0 for $\forall v_{RL} \in V_{RL}$ and $\forall s_{RL} \in S_{RL}$. We update the value of $Q_{t+1}^\kappa(v_{RL}, s_{RL})$ at the end of each contribution through Equation 8 if s_{RL} is a new action choice; or else, we stick with $Q_{t+1}^\kappa(v_{RL}, s_{RL}) = Q_t^\kappa(v_{RL}, s_{RL})$.

$$Q_{t+1}^\kappa(v_{RL}, s_{RL}) = (1 - \theta)Q_t^\kappa(v_{RL}, s_{RL}) + \theta R_t \quad (8)$$

From Equation 8, R_t is the Reward function which is attained through Equation 7 at the end of each contribution. θ indicates the learning rate which is a number between zero and 1. In this research, we set $\theta = 0.2$. In the beginning of each contribution, the agent κ selects an action through $\text{argmax}_{s \in S_{RL}} Q_t^\kappa(v_{RL}, s_{RL})$ and running a majority vote over the possible actions. And if the results are tied, one action is randomly selected. Throughout the learning procedure, the agent uses the ε - *greedy* approach within each time step to explore the new types of actions. In this study, the entries of the RL model (v) are learnt with $\varepsilon = 0.01$. Since our model applies a tabular version of Q-learning Algorithm (rather than the version which applies an approximation function), the solution should deterministically converge to an optimal policy [22]. We test ten random seeds, each with 5000 rounds of iterations to resolve the model parameters. It takes an average of 2230 iterations for each seed to converge.

7 Verification of Churn Reduction

The empirical evaluation of the personalized recommending systems appears to be challenging for many researchers who study the popular CQAs that do not provide direct control options. Hence, we use an off-line churn prediction model as a baseline to see if following the HRCR’s recommended questions will help to reduce the churning probability of the contributing users.

Baseline Churn Predictor. We build a probabilistic classification model (i.e., a Logistic Regression (LR)) to predict the churning probability of each user over time as she contributes to answering the questions. We use the churn prediction features of [16] as the evidential features for predicting churn. As Figure 6a summarizes, our baseline LR retrieves the Accuracy of 84% and the F1-score of 78% when it is applied to the test data. We get the Area Under Curve (AUC) value of 96%.

Comparison of the Churning Probabilities. Next we compare the average churn probability of the users based on the average similarity of their actions to the actions that are recommended by our HRCR algorithm over time. From Table 1, we know that there are 7 varying factors that the users can decide on before contributing to a question. At each contribution time, if the user totally complies with the HRCR’s recommended action factors, the user gains

Fig. 6: (a) Performance of the base churn predictor (b) Users’ churning probability for different ranges of similarity with the HRCR’s recommended actions

the similarity value of 7. If there are no common factors between the user’s action factors and the ones that the HRCR recommends, the user gets the similarity value of 0; thus, it is apparent that the similarity value should range between 0 and 7 based on the number of common activity factors between the HRCR’s action recommendations and the user performance. Figure 6b shows the average similarity of actions along with the average churning probability of each user over time. We observe a decreasing trend of the average churning probability of the users over time as they comply with the HRCR’s action recommendations.

Hypothesis Testing. We also run a statistical Brown-Forsythe test to see the effect of action similarities on the churning behavior of the users. The results show that the users whose actions share more than a 50% similarity with the HRCR’s recommended action choices over time, have a significantly different churning probability ($P = 0.05$) from those who share less similarities. Thus the hypothesis that the users who share more similarity with our proposed model are less likely to churn is achieved.

8 Conclusion and Future Work

Takeaway Message. Flow theory implies that the users who get into the Flow Zone might be less likely to churn [8]. Our results comply with the Flow theory. HRCR also fills the gap of question recommendations to the contributing users by considering their tendency for churning. We observe that the churning probability of the users in the engagement state of the Flow Zone is respectively 22.2%, 48.4%, and 6.7% less than the churning probability in the engagement states of Anxiety, Apathy, and Boredom. We also observe that there exists approximately 42% difference between the average churning probability of those who do not follow the HRCR’s recommendations at all and those who happen to contribute to the CQA fully in accordance to what the HRCR recommends.

Limitations and Future Work. In this research we investigated the users who have made at least more than or equal to 5 contributions and a reputation score above 10. Since these users are contributing more often to the CQA forums, reducing their churning probability was of higher interest for us in comparison with the ephemeral users. Besides, the ephemeral users make the dataset intensively imbalanced. Hence, our work is limited by the users we have studied.

Although there exists various versions of the flow theory [14], we stick with the quadrant framework of the flow theory for two main reasons: First, it matches with the best number of hidden states we retrieve for our HMM. In addition, if we increase the number of engagement states, the emission distributions of skills and challenges would become very similar to one another, and the efficiency of the predictions will drop. Our second reason is simplicity of the model and ease of understanding. Nevertheless, exploring the other variations of the flow theory remains as future work.

In this paper, we used the concept of Flow theory to recommend users better questions. We introduced an innovative two-fold algorithm (we called HRCR) using a *Hidden Markov Model* and a *Reinforcement Learning Model* to personalize the question recommendations on the CQAs. Experiment results on the popular CQA of Stack Overflow proves HRCR helpful in reducing churn.

9 Acknowledgement

This research has been supported in part by project 16214817 from the Research Grants Council of Hong Kong and the 5GEAR project from the Academy of Finland. We would like to also thank our reviewers and Mr. Young D. Kwon for their valuable comments.

References

1. Akaike, H.: A new look at the statistical model identification. *IEEE transactions on automatic control* **19**(6), 716–723 (1974)
2. Bosu, A., Corley, C.S., Heaton, D., Chatterji, D., Carver, J.C., Kraft, N.A.: Building reputation in stackoverflow: an empirical investigation. In: 2013 10th Working Conference on Mining Software Repositories (MSR). pp. 89–92. IEEE (2013)
3. Brown, M.B., Forsythe, A.B.: Robust tests for the equality of variances. *Journal of the American Statistical Association* **69**(346), 364–367 (1974)
4. Chang, S., Pal, A.: Routing questions for collaborative answering in community question answering. In: 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM 2013). pp. 494–501. IEEE (2013)
5. Chen, S.S., Gopinath, R.A.: Model selection in acoustic modeling. In: Sixth European Conference on Speech Communication and Technology (1999)
6. Choi, S., Ha, H., Hwang, U., Kim, C., Ha, J.W., Yoon, S.: Reinforcement learning based recommender system using biclustering technique. *arXiv preprint arXiv:1801.05532* (2018)
7. Christopher, M.B.: *Pattern Recognition and Machine Learning*, vol. 1. Springer-Verlag New York (2016)

8. Csikszentmihalyi, M.: Beyond boredom and anxiety. san francisco: Josseybass. Well-being: The foundations of hedonic psychology pp. 134–154 (1975)
9. Dror, G., Koren, Y., Maarek, Y., Szpektor, I.: I want to answer; who has a question?: Yahoo! answers recommender system. In: Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 1109–1117. ACM (2011)
10. Dror, G., Pelleg, D., Rokhlenko, O., Szpektor, I.: Churn prediction in new users of yahoo! answers. In: Proceedings of the 21st International Conference on World Wide Web. pp. 829–834. ACM (2012)
11. Karnstedt, M., Hennessy, T., Chan, J., Hayes, C.: Churn in social networks: A discussion boards case study. In: 2010 IEEE Second International Conference on Social Computing. pp. 233–240. IEEE (2010)
12. Li, C., Biswas, G.: A bayesian approach to temporal data clustering using hidden markov models. In: ICML. pp. 543–550 (2000)
13. Movshovitz-Attias, D., Movshovitz-Attias, Y., Steenkiste, P., Faloutsos, C.: Analysis of the reputation system and user contributions on a question answering website: Stackoverflow. In: Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. pp. 886–893. ACM (2013)
14. Nakamura, J., Csikszentmihalyi, M.: Flow theory and research. Handbook of positive psychology pp. 195–206 (2009)
15. Oliveira, N., Muller, M., Andrade, N., Reinecke, K.: The exchange in stackexchange: Divergences between stack overflow and its culturally diverse participants. Proceedings of the ACM on Human-Computer Interaction **2**(CSCW), 130 (2018)
16. Pudipeddi, J.S., Akoglu, L., Tong, H.: User churn in focused question answering sites: characterizations and prediction. In: Proceedings of the 23rd International Conference on World Wide Web. pp. 469–474. ACM (2014)
17. Rabiner, L.R., Juang, B.H.: An introduction to hidden markov models. *ieee assp magazine* **3**(1), 4–16 (1986)
18. San Pedro, J., Karatzoglou, A.: Question recommendation for collaborative question answering systems with rankslsda. In: Proceedings of the 8th ACM Conference on Recommender systems. pp. 193–200. ACM (2014)
19. Sutton, R.S., Barto, A.G.: Reinforcement learning: An introduction, vol. 1. MIT press Cambridge (1998)
20. Tang, X., Chen, Y., Li, X., Liu, J., Ying, Z.: A reinforcement learning approach to personalized learning recommendation systems. *British Journal of Mathematical and Statistical Psychology* **72**(1), 108–135 (2019)
21. Trimponias, G., Ma, X., Yang, Q.: Rating worker skills and task strains in collaborative crowd computing: A competitive perspective. In: The World Wide Web Conference. pp. 1853–1863. ACM (2019)
22. Watkins, C.J., Dayan, P.: Q-learning. *Machine learning* **8**(3-4), 279–292 (1992)
23. Yuan, S., Zhang, Y., Tang, J., Hall, W., Cabotà, J.B.: Expert finding in community question answering: a review. *Artificial Intelligence Review* pp. 1–32 (2019)
24. Zhang, J., Ackerman, M.S., Adamic, L.: Expertise networks in online communities: structure and algorithms. In: Proceedings of the 16th international conference on World Wide Web. pp. 221–230. ACM (2007)
25. Zheng, L., Chen, L.: Mutual benefit aware task assignment in a bipartite labor market. In: 2016 IEEE 32nd International Conference on Data Engineering (ICDE). pp. 73–84. IEEE (2016)