

Análise do Modelo BKT na Avaliação da Curva de Aprendizagem de Alunos de Algoritmos

Antonio Carlos Raposo¹, Dj Jefferson Maranhão¹, Carlos de Salles Soares Neto¹

¹Departamento de Informática – Universidade Federal do Maranhão (UFMA)
São Luis – MA – Brazil

{antoniocarlosraposo93,djefferson.maranhao}@gmail.com, csalles@deinf.ufma.br

Abstract. *The assessment of student learning algorithms is complex, either from the point of view of data collection and in the analysis of the result obtained with such collection. This article discusses the use of one of the most popular predictive model in the literature for the assessment of the student learning curve, which is the Bayesian Knowledge Tracing (BKT), in the context of an introductory discipline of algorithms. From data collected in the virtual learning environment created for the teaching of algorithms, the use of the BKT is analyzed in a qualitative manner, in an individual inspection process of several volunteers of an experiment. The results show that BKT is promising for the design of intelligent tutors that respond to individual student performance and recommend appropriate content during learning.*

Resumo. *A avaliação de aprendizagem de alunos de algoritmos é complexa, tanto do ponto-de-vista da coleta de dados quanto na análise do resultado obtido com tal coleta. Este artigo discute o emprego de um dos modelos preditivos mais populares na literatura para a avaliação da curva de aprendizagem de alunos, que é o Bayesian Knowledge Tracing (BKT), no contexto de uma disciplina introdutória de algoritmos. A partir de dados coletados em ambiente virtual de aprendizagem criado para o ensino de algoritmos, o emprego do BKT é analisado sob uma ótica qualitativa, em um processo de inspeção individual de diversos voluntários de um experimento. Os resultados mostram que o BKT é promissor para a concepção de tutores inteligentes que reajam ao desempenho individual dos alunos e recomendem conteúdo adequado durante a aprendizagem.*

1. Introdução

O aprendizado de algoritmos é um tema complexo, com inúmeras iniciativas didáticas que tentam aperfeiçoar a experiência de aprendizes e professores ¹ [Ribeiro et al. 2018]. Aprender algoritmos requer o conhecimento de fundamentos como abstração, decomposição e identificação de padrões, os quais nem sempre são bem tratados em uma disciplina introdutória de algoritmos em nível superior.

Adicionalmente, o processo de aprender algoritmos é modular, com etapas bem definidas, que são pré-requisitos umas das outras. Apenas para ilustrar, inicialmente se aprende o conceito de variáveis e atribuições como base para depois se aprender comandos

¹<https://code.org/>

condicionais (se). Por sua vez, comandos condicionais são necessários para se aprender o conceito de laços de repetição.

O progresso de um aluno em cada uma dessas etapas é fundamental para o progresso adequado nas etapas seguintes. Isso torna ainda mais complexa a avaliação da curva de aprendizagem de um aluno, pois a falta de aptidão em um conhecimento prévio pode ser motivo da aprendizagem inadequada de um tópico futuro.

A coleta de dados é outro fator que torna custoso o trabalho de acompanhamento de uma turma de aprendizes de algoritmos. A correção manual de uma lista de exercícios, por exemplo, pode se tornar proibitiva, pois requer muito tempo do professor. Por outro lado, esses exercícios possibilitam a avaliação da curva de aprendizagem dos alunos de uma forma mais prática.

É por essa razão que se vê uma ampla diversidade de ferramentas que apoiam o professor nessa tarefa, como o URI Online [Bez et al. 2014] e o UVA Online Judge [Revilla et al. 2008]. Tais ferramentas, no entanto, limitam-se a informar se o aluno conseguiu, ou não, realizar as tarefas pedidas, sem apresentar informações qualitativas acerca do código criado por tais alunos (inclusive sem suporte a outros requisitos funcionais, como mecanismos de detecção de plágio eficientes).

Em suma, o aprendizado de algoritmos requer fundamentos que são complexos de serem assimilados, pois exigem bastante prática. Além disso, a coleta e a análise dos dados não é trivial, posto que se trata de um conhecimento modular. Pela soma desses fatores, criar uma única métrica capaz de estabelecer em que ponto da curva de aprendizagem está um aluno ou uma turma é algo complexo.

Neste artigo, apresenta-se o processo de análise de dados coletados no Ambiente Virtual de Aprendizagem (AVA) COSMO criado especificamente para o ensino de algoritmos [RABÊLO JÚNIOR et al. 2018]. Na presente proposta, empregou-se o Bayesian Knowledge Tracing (BKT) para avaliação da curva de aprendizagem de aprendizes. Há uma discussão qualitativa discorrendo sobre a análise dos resultados obtidos com um experimento que leva em consideração a coleta de dados de 46 voluntários de uma disciplina introdutória de algoritmos. A discussão apresenta uma avaliação qualitativa detalhada de alguns desses voluntários do experimento.

O objetivo deste trabalho é analisar qualitativamente o emprego do BKT como modelo para a predição da curva de aprendizado de alunos em uma disciplina introdutória de algoritmos. Ao avaliar a probabilidade de transição para o estado de conhecimento de um determinado assunto, o que é definido pelo BKT, os voluntários foram observados individualmente e algumas dessas análises são apresentadas. O BKT se demonstra promissor como uma forma rica e útil para a construção de sistemas tutores inteligentes e recomendadores de conteúdo educacional que leva em conta a curva de aprendizado atual dos alunos.

O artigo está organizado como se segue. A seção 2 apresenta fundamentação sobre o *Bayesian Knowledge Tracing*. A seção 3 apresenta trabalhos relacionados a esta pesquisa. A seção 4 explica como o experimento foi executado e a seção 5 apresenta uma discussão sobre alguns dos resultados obtidos nesta pesquisa.

2. Bayesian Knowledge Tracing

A *Bayesian Knowledge Tracing* (BKT) é uma das técnicas de modelagem mais populares no campo dos Sistemas de Tutores Inteligentes [Yudelson 2016]. Modelos BKT são derivados dos *Hidden Markov Models* (HMM) e constituídos por variáveis latentes e observadas [Käser et al. 2017]. As variáveis latentes representam o conhecimento acerca de um determinado conteúdo, assumindo que o aprendiz domina o conteúdo ou não. Por sua vez, as variáveis observadas representam as respostas atribuídas pelos aprendizes às questões de um determinado assunto, as quais podem ser classificadas em corretas e incorretas.

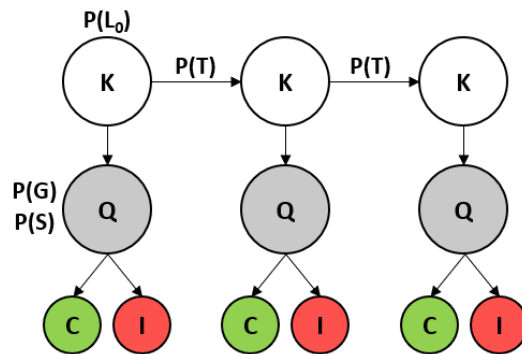


Figura 1. BKT.

O modelo básico de *Knowledge Tracing* é mostrado na Figura 1. Segundo [Yudelson et al. 2013], esse modelo é definido pelos quatro parâmetros seguintes: a probabilidade inicial de se possuir um determinado conhecimento K a priori, definida $p(L_0)$; a probabilidade de que a habilidade será aprendida em cada oportunidade de praticar o conhecimento, definida $p(T)$; a probabilidade do estudante cometer um erro quando já detém um determinado conhecimento, definida $p(S)$; e a probabilidade do estudante acertar uma questão ao aplicar uma habilidade ainda não conhecida, definida $p(G)$.

A variável Q corresponde à resposta atribuída por um aprendiz a uma determinada questão. Essas variáveis possuem valores binários, isto é, podem ser corretas ou incorretas. Com base nesses parâmetros, pode-se inferir quanto ao conhecimento de um estudante na n -ésima oportunidade de aplicar um conhecimento, que é definido por $P(L_n)$.

De acordo com [David et al. 2016], as hipóteses básicas desse modelo são: 1) o aprendiz é avaliado de forma dicotômica em cada oportunidade de aprendizado; e 2) o aprendiz pode estar no estado conhecido ou desconhecido para cada componente de conhecimento, e, uma vez aprendida, a habilidade jamais é esquecida.

Usando-se os parâmetros definidos acima, prever quando a resposta de um aprendiz está correta na oportunidade n pode ser calculada por meio da Lei das Probabilidades Totais, da seguinte forma:

$$P(\text{correto}_n) = P(L_n) \times (1 - P(S)) + (1 - P(L_n)) \times P(G) \quad (1)$$

De forma similar, a probabilidade de que a resposta seja incorreta ao tempo n pode ser calculada pela fórmula a seguir:

$$P(\text{incorreto}_n) = P(L_n) \times P(S) + (1 - P(L_n)) \times (1 - P(G)) \quad (2)$$

Segundo [David et al. 2016], há dois estágios distintos ao se utilizar o modelo de conhecimento. O primeiro estágio consiste no ajuste dos parâmetros do modelo a partir dos dados, denominado de *fitting*. O segundo estágio consiste em usar o modelo para inferir o conhecimento do estudante ao longo do tempo com base em suas respostas. Dada a observação de uma resposta de estudante no tempo n (correta ou incorreta), a probabilidade $p(L_n)$ de que o estudante conheça a habilidade pode ser calculada pelo Teorema de Bayes. Assim, quando uma resposta correta é observada, a probabilidade pode ser obtida da seguinte forma:

$$P(L_n | \text{correto}_n) = \frac{P(L_n) \times (1 - P(S))}{P(\text{correto}_n)} \quad (3)$$

em que $P(\text{correto}_n)$ foi definida na Equação 1. Quando uma resposta incorreta é observada, a probabilidade é obtida da seguinte maneira:

$$P(L_n | \text{incorreto}_n) = \frac{P(L_n) \times P(S)}{P(\text{incorreto}_n)} \quad (4)$$

em que $P(\text{incorreto}_n)$ foi definida na Equação 2.

Por último, mostra-se como o conhecimento do estudante acerca do conteúdo pode ser atualizado dadas suas interações com o sistema. O cálculo é a soma de duas probabilidades: a probabilidade posterior de que o estudante já conheça a habilidade, e a probabilidade de que o estudante não detenha a habilidade, mas foi capaz de aprendê-la.

$$P(L_n) = P(L_{n-1} | \text{obs}_{n-1}) + (1 - P(L_{n-1} | \text{obs}_{n-1})) \times P(T) \quad (5)$$

2.1. Abordagens para ajuste dos parâmetros do modelo

O modelo tradicional do BKT é vulnerável ao problema estatístico da "identificabilidade", isto é, uma mesma estimativa de desempenho pode ser resultado de várias combinações distintas de parâmetros do modelo [Beck and Chang 2007]. Dessa forma, múltiplas combinações de $P(L_0)$, $P(T)$, $P(G)$ e $P(S)$ podem resultar em uma mesma taxa de erro para $P(\text{correto}_n)$.

Para resolver esse problema, [Beck and Chang 2007] propôs avaliar as diferentes soluções para encontrar as mais plausíveis, restringindo os parâmetros do modelo por meio de uma probabilidade prévia obtida de todos os componentes de conhecimento. Contudo, segundo [d Baker et al. 2008], essa solução pode levar a um novo problema, a "degeneração do modelo", que é a obtenção de parâmetros que refletem um comportamento paradoxal.

Isso acontece quando $P(S)$ e $P(G)$ são maiores que 0.5. Quando $P(S) > 0.5$, o estudante que sabe um conteúdo tem maior chance de errar do que de acertar a resposta. Por outro lado, quando $P(G) > 0.5$, o estudante que não sabe um conteúdo tem maior chance de acertar do que de errar a resposta.

Há diversas maneiras de se fazer o ajuste dos parâmetros para tentar resolver os problemas de degeneração e identificabilidade. A mais simples consiste na atribuição de valores fixos aos parâmetros $P(L_0)$, $P(T)$, $P(G)$ e $P(S)$ para cada componente de conhecimento, conforme [Corbett and Anderson 1994]. Nesse caso, $P(S)$ e $P(G)$ devem receber valores menores que 0.3, como forma de evitar a degeneração do modelo.

Outras abordagens para ajuste dos parâmetros são: *Expectation-Maximization* [Beck and Chang 2007], *Conjugate Gradient Search* [Corbett and Anderson 1994], *Discretized Brute-Force Search* [Yudelson et al. 2013], *Contextual Guess-and-Slip* [Beck and Chang 2007] e *Individual Difference Weights* [Corbett and Anderson 1994].

3. Trabalhos Relacionados

3.1. *Individualized Bayesian Knowledge Tracing Model*

Em [Yudelson et al. 2013] é apresentada uma abordagem para a criação de modelos individualizados utilizando o método BKT, diferenciando os estudantes com relação a sua probabilidade inicial de aprender um certa habilidade. Essa implementação é capaz de melhorar a acurácia da predição de sucesso de um aluno em relação a uma atividade.

3.2. *More Accurate Student Modeling through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing*

Em [d Baker et al. 2008] é apresentado um método para se instanciar os parâmetros do BKT utilizando aprendizado de máquina para calcular os valores iniciais de *guess* e *slip*. Esse método reduz o problema de degeneração e, além disso, não intensifica o problema de identificabilidade, situação descrita anteriormente, em que dois alunos com dados parecidos poderiam ter modelos significativamente diferentes.

3.3. *Modeling Student Learning Outcome in Studying Program Language Course*

Em [Wang et al. 2017] é apresentada uma variação do modelo BKT utilizando a Teoria da Resposta ao Item (TRI) para calcular o estado inicial do conhecimento de estudantes de uma turma do curso de programação. [Wang et al. 2017] apresenta algumas das dificuldades que existem para se modelar o conhecimento de estudantes de programação.

O modelo BKT-IRT usa a Teoria da Resposta ao Item para calcular o estado inicial do conhecimento de uma determinada habilidade $P(L_0)$ baseado no nível de dificuldade das atividades e no nível de proficiência do estudante. Esse modelo não precisa de dados para o treinamento de cada habilidade, pois consegue estimar o conhecimento do estudante baseado em outras habilidades, na dificuldade de cada habilidade e nas diferenças entre os alunos.

4. Experimento

Para a realização do experimento, foram coletadas informações de 46 alunos utilizando o Ambiente Virtual de Aprendizagem COSMO [RABÊLO JÚNIOR et al. 2018]. Tal ambiente conta com 24 questões básicas de programação, as quais foram desenvolvidas pelo professor da disciplina de Algoritmos I de um curso superior em Ciência da Computação e por sua equipe de monitoria. Embora realizado em sala de aula, o experimento era voluntário, sem nenhum caráter obrigatório, nem oferecia qualquer benefício na avaliação da disciplina. Adicionalmente, as informações pessoais dos alunos foram anonimizadas.

Os participantes foram alunos do primeiro semestre do curso de Ciência da Computação, inscritos na disciplina de Algoritmos I. É nessa disciplina que, normalmente, o aluno tem o primeiro contato com os conceitos básicos de programação, sendo

de extrema importância. O experimento foi realizado durante a terceira e a quarta semanas de aula da disciplina. Nesse período, os alunos responderam atividades voltadas aos conteúdos de comandos de atribuição/variáveis e comandos de controle/condicionais.

O experimento foi conduzido em três etapas. Primeiro, durante aula realizada no laboratório de ensino, o professor apresentou as atividades disponíveis no Ambiente Virtual. Após isso, os alunos tiveram acesso a plataforma para responder as atividades por aproximadamente duas semanas. Por último, os alunos fizeram a primeira prova da disciplina.

As informações coletadas – registros de tentativas dos voluntários, suas respostas e horários de uso – foram utilizadas para alimentar o algoritmo BKT e, com isso, criar os modelos de conhecimento dos voluntários. Alguns desses modelos serão analisados individualmente na seção seguinte.

5. Resultados

Esta seção apresenta os resultados principais do experimento, por meio dos dados obtidos e da análise individual dos modelos de conhecimento considerados mais relevantes.

5.1. Relação entre o número de questões respondidas e a nota na avaliação

Em média, os alunos que participaram do experimento obtiveram 7.17 na avaliação, enquanto os alunos que escolheram não participar atingiram uma média de 5.8, de um total de 10. Além disso, notou-se que os alunos que praticaram mais, isto é, responderam um número maior de questões, se saíram melhor na avaliação da disciplina, o que pode ser visto na Figura 2.

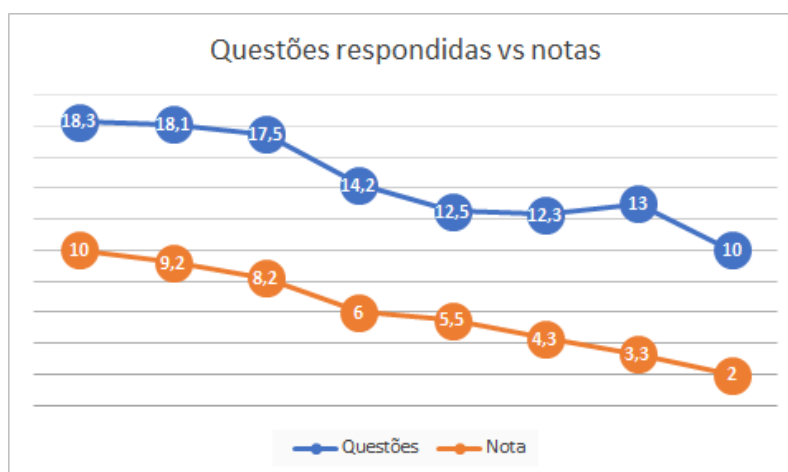


Figura 2. Questões Respondidas vs Notas

Foram inspecionados e avaliados individualmente os dados de todos os 46 voluntários. Há uma clara coerência entre os modelos gerados pelo BKT e as notas obtidas durante a primeira avaliação da disciplina de Algoritmos. Devido a limitação de páginas do artigo, foram escolhidos três voluntários para ilustrar essa coerência, situações de contorno, riscos e limitações que o emprego do BKT pode gerar. Tais informações são detalhadas nas subseções seguintes.

5.2. Análise do modelo de conhecimento do Voluntário 1

O Voluntário 1 atingiu um total de 24 questões corretas, com um total de 139 tentativas. A Figura 3 apresenta a sua curva de aprendizado, elaborada com base nas probabilidades de transição $P(T)$ calculadas pelo BKT para a componente de conhecimento (atribuições e variáveis). A probabilidade de conhecimento inicial $P(L_0)$ para o presente voluntário foi estimada em 0.00. Dessa forma, o modelo apresentou o comportamento esperado, pois a curva mostra que o voluntário atinge o ápice muito rapidamente, e há pequenas reduções quando responde às questões incorretamente.

Para a outra componente de conhecimento (condicionais), o modelo teve um comportamento diferente, porém dentro do esperado. Como se pode observar na Figura 4, o Voluntário 1, em seu início, respondeu várias questões de forma precisa, fazendo com que sua probabilidade de transição $P(T)$ se aproximasse do máximo. Entretanto, em determinado momento, o Voluntário 1 chegou a uma questão e não conseguiu resolvê-la, mesmo realizando diversas tentativas (especificamente 68 tentativas erradas apenas em uma questão, que ele nunca veio a acertar). Isso fez com que sua probabilidade de transição tivesse uma redução, estabilizando-se em 0.4.

Isso mostra que os modelos gerados pelo BKT se comportaram de maneira adequada. Quando o Voluntário 1 acerta a questão, a probabilidade de transição sobe de maneira exponencial. Porém, quando erra, a descida é gradual. É interessante perceber que a probabilidade de transição estabiliza em 0.4, pois representa que uma parcela do conhecimento adquirido anteriormente não é perdido, mesmo após uma série de respostas incorretas. É provável que, quando o voluntário voltar a acertar questões desse componente de conhecimento, sua probabilidade de transição volte a subir novamente.

Por fim, quando a curva de aprendizado tiver uma queda substancial como a relatada, o tutor inteligente deve agir para melhorar a experiência de aprendizado e o desempenho do aluno naquele conteúdo, oferecendo dicas para a resolução da atividade ou sugestões de outras atividades.



Figura 3. Vol 1 - Variáveis

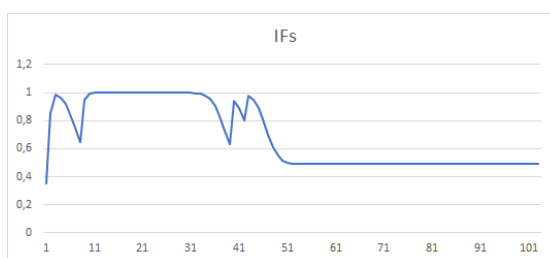


Figura 4. Vol 1 - Condicionais

5.3. Análise do modelo de conhecimento do Voluntário 2

O Voluntário 2 respondeu a um total de 16 questões em 28 tentativas, obtendo um dos melhores desempenhos em termos de questões por tentativa. A Figura 5 apresenta o gráfico de sua curva de aprendizado utilizando as probabilidades de transição para as questões de variáveis. Devido ao seu comportamento incomum, a sua curva de aprendizado se aproxima do limite muito rapidamente, considerando que: 1) o aluno possuía algum conhecimento anterior $P(L_0) = 0.0$; e 2) o aluno assimilou o conhecimento necessário.

Para o conteúdo de condicionais, o Voluntário 2 realizou um total de 9 (nove) tentativas, das quais acertou 6 (seis) e errou 3 (três). A curva de aprendizado estimada pelo modelo corresponde com os dados de uso do sistema, contudo, não refletem o comportamento do aluno em sala de aula, como se explicará adiante. Assim, seu nível de aprendizado teórico variou de 0.35, ao responder a primeira questão corretamente, a 1.00, a partir da quinta questão correta, como mostra a Figura 6.

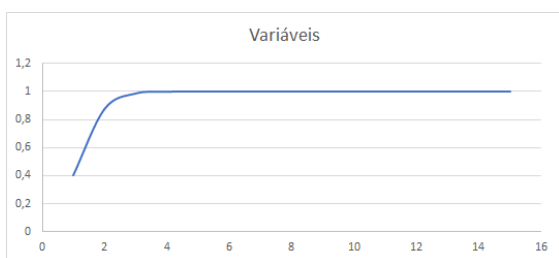


Figura 5. Vol 2 - Variáveis

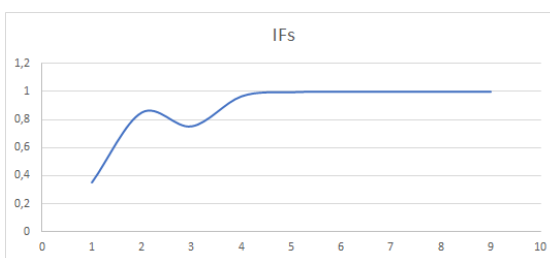


Figura 6. Vol 2 - Condicionais

Finalmente, deve-se alertar para o fato de que o Voluntário 2 apresentou uma das menores notas na avaliação da disciplina, tendo obtido 3.5, quando era esperado em torno de 7.0, pelo gráfico mostrado na Figura 2. Assim, com base nas estatísticas retiradas do modelo e no relatório de notas, pode-se concluir que o Voluntário 2 provavelmente teve acesso às respostas das questões (todas as suas submissões ocorreram apenas em horários das aulas de laboratório, onde estava próximo de outros alunos) e por isso não desenvolveu seu conhecimento como deveria. Informações como essas são importantes para que o professor possa atuar junto aos alunos, corrigindo os rumos do aprendizado.

5.4. Análise do modelo de conhecimento do Voluntário 3

O Voluntário 3 respondeu um total de 23 questões em 121 tentativas, obtendo um rendimento médio em termos de questões por tentativa. Para o tópico de comandos de atribuição/variáveis, esse voluntário realizou 35 (trinta e cinco) tentativas, obtendo acertos em 12 (doze) questões. A Figura 7 apresenta a curva de aprendizado do estudante para esse conceito, mostrando que o voluntário teve dificuldades em duas questões desse assunto. De fato, ao analisar os registros de uso da plataforma, verifica-se que em duas questões o usuário precisou de aproximadamente 8 (oito) tentativas para chegar ao acerto.



Figura 7. Vol 1 - Variáveis

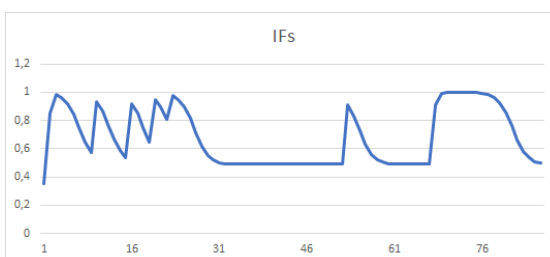


Figura 8. Vol 1 - IF

Para o conteúdo de condicionais, o Voluntário 3 realizou um total de 86 (oitenta e seis) tentativas, tendo conseguido acertar 11 (onze) questões. Ao todo, o estudante realizou 75 (setenta e cinco) tentativas incorretas e não conseguiu acertar apenas 2 (duas)

de 13 (treze) questões cadastradas para esse conceito. A sua curva de aprendizado, mostrada na Figura 8, representa bem o comportamento desse estudante, com as oscilações típicas de quem está tendo contato com um assunto pela primeira vez. Assim, a curva mostra o declínio do nível de domínio daquele tópico quando ocorrem falhas sucessivas e a retomada quando os acertos voltam a ocorrer.

Por fim, verifica-se que, embora a probabilidade de transição para o conteúdo de condicionais tenha terminado em aproximadamente 0.5, o voluntário apresentou uma das melhores notas na primeira prova da disciplina, tirando 9,5 de um total de 10. Isso mostra a importância das tentativas e dos erros durante a etapa de consolidação de um conhecimento.

6. Conclusão

Este artigo descreve uma análise do emprego do BKT como modelo para a predição da curva de aprendizado (ou o modelo de conhecimento) de alunos em uma disciplina introdutória de algoritmos. Um experimento foi realizado com 46 voluntários, sobre os quais se fez uma detalhada coleta de dados em um ambiente virtual de aprendizado focado na resolução de diversos problemas computacionais. Tais problemas versam sobre dois assuntos introdutórios: variáveis e atribuições; e comandos condicionais (se). A análise do emprego do BKT demonstra que o modelo é promissor. Percebe-se coerência ao se comparar a predição identificada pelo BKT e depois a nota na avaliação formativa da disciplina e opinião do professor da mesma sobre seus alunos.

Foram destacados neste artigo os dados de três voluntários, com a finalidade de ilustrar tanto a riqueza de conclusões quanto as limitações e riscos no emprego do BKT. O Voluntário 1 teve dificuldades em concluir uma das atividades, com um comportamento persistente de 68 tentativas erradas para o mesmo problema, sem nunca conseguir sucesso. Ainda assim, o BKT não degradou excessivamente a predição para aquele aluno, porque leva em consideração seu histórico. Nesse caso o BKT seria capaz de reestabelecer rapidamente para uma chance alta de aptidão caso o aluno resolvesse novos problemas corretamente. O Voluntário 2 tirou uma nota baixa na avaliação formativa da disciplina, porém mostra um desempenho muito eficiente no ambiente virtual de aprendizado. O histórico do Voluntário 2 mostra que o mesmo só usou o ambiente durante as aulas de laboratório e é provável que esse resultado não-conforme seja motivado por ele ter feito os problemas a partir da interação com alunos ao seu lado, copiando as soluções. Aqui o BKT demonstra-se limitado em ser um bom modelo preditivo com essa intervenção externa. Já o Voluntário 3 é um exemplo em que o BKT identifica falta de aptidão prévia na disciplina e que mesmo em um cenário de tentativa e erro, o modelo de conhecimento ainda é adequado.

Os resultados obtidos neste trabalho permitem subsidiar a criação de tutores inteligentes e recomendadores de conteúdo que se baseiem no modelo de conhecimento gerado pelo BKT para a tomada de decisão sobre o progresso, ou não, de aprendizes.

Durante o trabalho se percebeu como os dados coletados no ambiente virtual de aprendizado poderiam ser melhores. Um trabalho futuro interessante reside na melhoria da coleta de dados, bem como um aumento na amostra do experimento. Como exemplo, percebeu-se que é necessário que os erros coletados sejam tratados de maneira diferente. Um erro de compilação, por exemplo, deve ser tratado diferente de um erro semântico, já

que os alunos usam o próprio AVA para testar a sintaxe de seus códigos.

Além disso, é necessária uma investigação de outras técnicas de modelagem de conhecimento como *Performance Factor Analysis*, a fim de se fazer uma análise comparativa de tais modelos e verificar a viabilidade de outras técnicas para a modelagem do conhecimento de alunos de algoritmos.

Referências

- Beck, J. E. and Chang, K.-m. (2007). Identifiability: A fundamental problem of student modeling. In *International Conference on User Modeling*, pages 137–146. Springer.
- Bez, J. L., Tonin, N. A., and Rodegheri, P. R. (2014). Uri online judge academic: A tool for algorithms and programming classes. In *2014 9th International Conference on Computer Science Education*, pages 149–152.
- Corbett, A. T. and Anderson, J. R. (1994). Knowledge tracing: Modeling the acquisition of procedural knowledge. *User modeling and user-adapted interaction*, 4(4):253–278.
- d Baker, R. S., Corbett, A. T., and Aleven, V. (2008). More accurate student modeling through contextual estimation of slip and guess probabilities in bayesian knowledge tracing. In *International conference on intelligent tutoring systems*, pages 406–415. Springer.
- David, Y. B., Segal, A., and Gal, Y. K. (2016). Sequencing educational content in classrooms using bayesian knowledge tracing. In *Proceedings of the sixth international conference on Learning Analytics & Knowledge*, pages 354–363. ACM.
- Käser, T., Klingler, S., Schwing, A. G., and Gross, M. (2017). Dynamic bayesian networks for student modeling. *IEEE Transactions on Learning Technologies*, 10(4):450–462.
- RABÊLO JÚNIOR, D. J. L. et al. (2018). Cosmo: Um ambiente virtual de aprendizado com foco na introdução de algoritmos.
- Revilla, M. A., Manzoor, S., and Liu, R. (2008). Competitive learning in informatics: The uva online judge experience. *Olympiads in Informatics*, 2(10):131–148.
- Ribeiro, R. B., Fernandes, D., de Carvalho, L. S. G., and Oliveira, E. (2018). Gamificação de um sistema de juiz online para motivar alunos em disciplina de programação introdutória. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 29, page 805.
- Wang, S., Han, Y., Wu, W., and Hu, Z. (2017). Modeling student learning outcomes in studying programming language course. In *2017 Seventh International Conference on Information Science and Technology (ICIST)*, pages 263–270.
- Yudelson, M. V. (2016). Individualizing bayesian knowledge tracing. are skill parameters more important than student parameters?. *International Educational Data Mining Society*.
- Yudelson, M. V., Koedinger, K. R., and Gordon, G. J. (2013). Individualized bayesian knowledge tracing models. In *International conference on artificial intelligence in education*, pages 171–180. Springer.