

Um modelo bayesiano para o reconhecimento da motivação no ensino e aprendizagem de programação

Evandro Franzen¹, Magda Bercht²

Universidade do Vale do Taquari – Univates¹, Universidade Federal do Rio Grande do Sul – UFRGS²

e-mail: evandrofra@gmail.com, bercht@ufrgs.br

Resumo. Este artigo apresenta um modelo para o reconhecimento da motivação de estudantes na realização de atividades de programação em disciplinas introdutórias. Ele foi desenvolvido baseado nos fatores de esforço e confiança, que podem caracterizar a motivação. Foram coletados dados usados na construção de duas Redes Bayesianas que permitem identificar o esforço e a confiança demonstrados em cada atividade. Experimentos foram aplicados em estudantes universitários de uma Instituição de Ensino Superior, em cursos da área de computação. Os testes realizados demonstram que é possível reconhecer o nível de esforço e confiança e as probabilidades associadas aos estes níveis. Os resultados permitiram concluir que as Redes Bayesianas se constituíram em uma boa alternativa para o reconhecimento automático da motivação. Este modelo está sendo incorporado a um software de apoio ao desenvolvimento de atividades em disciplinas de programação, que utiliza metodologia ativa como base da ação pedagógica.

Palavras-chave: Ensino e aprendizagem de programação, motivação, problematização, mineração de dados educacionais, Redes Bayesianas.

Introdução

As disciplinas de programação são componentes curriculares fundamentais e presentes na grande maioria dos cursos de graduação na área de computação. Entretanto, o ensino de programação ainda é um desafio, pois grande parte dos estudantes demonstra dificuldades no aprendizado, reprova ou evade durante o período letivo. Tradicionalmente a reprovação em disciplinas introdutórias de programação tem sido significativa, como mostram os estudos de Watson e Li (2014) e Ramos et al. (2015). Pesquisas mostram que grande parte dos alunos admite que tem dificuldade de compreender o enunciado ou de acompanhar o ritmo das aulas, mas não faz perguntas ou questionamentos para resolução de dúvidas (GIRAFFA e MORA, 2013).

Estudos realizados por Twenge e Campbell (2011) indicam que gerações mais novas apresentam elevados níveis de narcisismo, ansiedade e depressão, além de menor capacidade para assumir a responsabilidade pelo sucesso ou fracasso. Ao se depararem com as primeiras dificuldades, estes estudantes podem se desmotivar e a considerar que o curso é inadequado ou que o tempo necessário para se tornar um profissional será grande demais. Em outros casos, falta disposição e persistência para se manter na disciplina ou no curso. Dum Dumaya, Banawan e Rodrigo (2018) estabelecem uma relação direta entre persistência, esforço e motivação, destacando que a persistência do estudante é um dos aspectos fundamentais em tarefas que envolvem a resolução de problemas.

É nas disciplinas introdutórias que o problema acontece com mais frequência, em muitos casos o professor precisa atuar para manter o interesse, a motivação e para deixar claro que as atividades exigem esforço e dedicação. Desta forma, a identificação dos aspectos relacionados a motivação dos estudantes nestes componentes curriculares pode contribuir para que professores executem ações que visam estimular a dedicação e a autonomia do estudante.

O modelo apresentado neste trabalho se baseia nos fundamentos da motivação definidos por Du Boulay e Del Soldato (2016) que argumentam que a motivação está relacionada ao esforço, à independência e confiança demonstradas pelo aluno. A motivação é definida de forma pragmática, em uma combinação do esforço empreendido para o aprendizado e da confiança que o aluno demonstra em

sua habilidade de aprender ou resolver um problema em um contexto específico com um mínimo de auxílio, o que caracteriza também um comportamento mais independente.

Entre as diversas técnicas computacionais disponíveis para o reconhecimento da motivação e outros estados afetivos, estão as técnicas baseadas nos fundamentos da mineração de dados, especialmente as Redes Bayesianas (RB). A possibilidade de classificação perante a incerteza e diante de contextos onde não é possível afirmar que uma entrada pertence a uma classe com total certeza fazem das Redes Bayesianas uma boa alternativa em aplicações que visam reconhecer emoções (LONGHI, BEHAR E BERCHT, 2010).

Considerando o exposto, este trabalho apresenta o desenvolvimento de um modelo que utiliza RB para reconhecer o esforço e confiança demonstrado por estudantes nas atividades desenvolvidas em disciplinas introdutórias de programação. A seção 2 apresenta os fundamentos da motivação e das Redes Bayesianas. As etapas e a metodologia adotada é apresentada na seção 3 e na sequência são apresentados os resultados que consistem na descrição do modelo e nos testes realizados.

Fundamentação

Motivação

O estado emocional ou estado afetivo diz respeito ao conjunto de características que determinam a emoção em um indivíduo, em determinado momento, um estado da mente e do corpo que caracteriza a emoção em determinado instante. Motivação, engajamento, estado de ânimo e frustração estão entre os estados afetivos apresentados com frequência. A Computação Afetiva é a área da computação que investiga como capacitar os computadores a identificar, compreender e representar estados afetivos (PICARD, 1997). Conferir a um sistema recursos para identificar que um estudante se encontra desmotivado, por exemplo, pode permitir a indicação de ações ou tarefas alternativas, durante a resolução de um problema. Embora considerados muitas vezes sinônimos, motivação, interesse ou engajamento não são caracterizados pelo mesmo estado físico e mental, e a identificação de cada um envolve diferentes estratégias de observação e coleta de dados. Considerando uma perspectiva pedagógica, é possível utilizar sistemas computacionais que capturem informações sobre o comportamento do aluno, tais como o número de erros cometidos em uma atividade, o tempo de dedicado à tarefa, ou ainda o número de tentativas para resolver o problema (LONGHI, BEHAR e BERCHT, 2010; DU BOULAY e DEL SOLDATO, 2016).

O modelo descrito por Du Boulay e Del Soldato (2016) sustenta que a motivação pode ser definida por três fatores: Esforço, Confiança e Independência (ECI). Segundo os autores, o esforço é um indicativo de motivação intrínseca e a sua utilização na maioria dos casos é mais indicada do que outros aspectos, como o desempenho. Na visão dos autores, o esforço pode ser medido a partir da dedicação e perseverança do estudante, e de uma análise de como ele enfrenta dificuldades e obstáculos. A confiança e a independência podem ser obtidas pela observação da quantidade de auxílio demandada durante a atividade. Alunos desmotivados na maioria das situações não se esforçam ou não acreditam que podem ter sucesso na resolução das tarefas (DU BOULAY e DEL SOLDATO, 2016). Um aspecto importante no contexto deste trabalho é que a abordagem citada permite associar os fatores a dados coletados e a partir de uma análise qualitativa, é possível medir, categorizar ou definir níveis de motivação.

Redes Bayesianas

Segundo Tan, Steinbach e Kumar (2009) uma Rede Bayesiana (RB) é uma representação gráfica de relacionamentos probabilísticos entre variáveis. Consiste em um grafo acíclico direcionado (DAG) que mostra as dependências entre os atributos. Uma RB possui também uma tabela das probabilidades associadas às ligações entre os nós. A figura 1 mostra uma rede que visa prever o desempenho de

estudantes (JÚNIOR, WILGES, NASSAR, 2014). É possível visualizar uma variável que é a classe (desempenho) e os demais atributos, dependentes, que indicam nível, assunto e tipo de atividade.

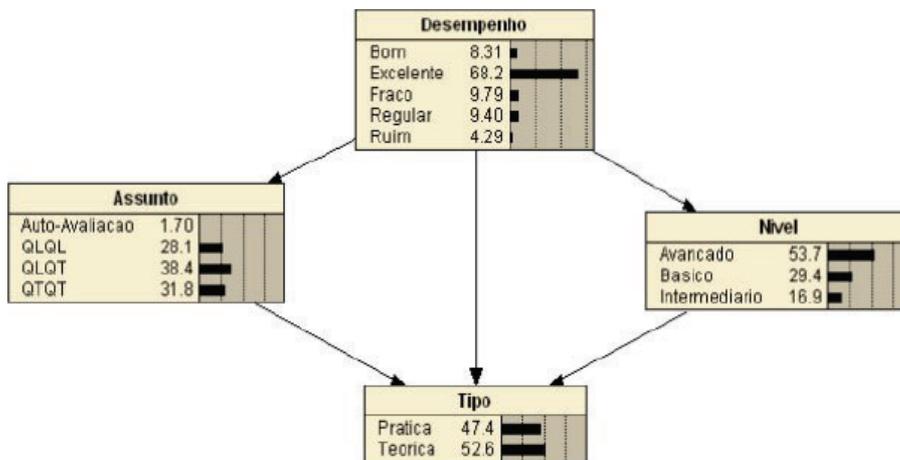


Figura 1. Exemplo de Rede Bayesiana. Fonte: Júnior, Wilges, Nassar (2014).

As RB são uma das estratégias probabilísticas mais usadas para representação do conhecimento e de informações permeadas pela incerteza. As arestas que ligam os nós representam uma precedência, ou influência causal entre os nodos (variáveis do problema). As precedências dos nodos pais em relação aos filhos determina uma distribuição de uma probabilidade (P), gerando uma quantificação da influência dos pais em cada um dos filhos (LONGHI, BEHAR E BERCHT, 2010).

Redes Bayesianas podem ser elaboradas de duas maneiras distintas. A primeira é a construção e a determinação das probabilidades, de acordo com o conhecimento de um especialista. Outra forma é a aprendizagem usando um conjunto de dados sobre os quais são aplicados algoritmos de *Machine Learning*. No processo de treinamento, pode ocorrer tanto o aprendizado da estrutura da rede, quanto dos parâmetros ou probabilidades associadas às entradas. A aprendizagem da estrutura da rede identifica as dependências entre os atributos e a direção da causalidade. Se a estrutura da rede for pré- definida, a principal atividade no treinamento é obter a tabela de probabilidades, usadas na classificação dos casos de entrada (JÚNIOR, WILGES, NASSAR, 2014).

Pesquisas relacionando Redes Bayesianas e o reconhecimento de afetos ou estados afetivos na área da educação têm sido cada vez mais frequentes. Um dos trabalhos mais relevantes foi desenvolvido por Longhi, Behar e Bercht (2010). Os principais objetivos foram a identificação dos estados de ânimo dos estudantes, dotando um AVEA (Ambiente Virtual de Ensino e Aprendizagem) de recursos para identificação dos mesmos. Fatores associados ao comportamento e à personalidade foram utilizados pela rede para mapear o estado afetivo exibido pelo estudante.

A aplicação de Redes Bayesianas no aprendizado de programação pode ser observada também no trabalho desenvolvido por Vier, Glutz e Jaques (2015). Foi proposto um modelo probabilístico do aluno para a representação do conhecimento do estudante em um STI (Sistema de Tutor Inteligente). O principal objetivo dos autores foi aprimorar o feedback dado aos estudantes durante a realização das atividades.

O estudo apresentado neste artigo assemelha-se ao trabalho desenvolvido por Longhi, Behar e Bercht (2010) por utilizar os fundamentos da motivação definidos por Du Boulay e Del Soldato (2016) e por ter como principal objetivo o reconhecimento de um estado afetivo. Em relação ao estudo de Vier, Glutz e Jaques (2015), a principal semelhança diz respeito ao foco em atividades de programação e na tentativa de elaborar um modelo que contribua para acompanhar o comportamento do estudante. Como principais diferenças salienta-se a utilização de um método de Aprendizagem Ativa, baseado na problematização e o foco nas etapas anteriores à escrita do código fonte, com o intuito de avaliar o esforço a confiança associados à compreensão e especificação do problema.

Metodologia

O modelo descrito está inserido em uma pesquisa de caráter experimental quem iniciou com um estudo piloto exploratório, no qual se avaliou a viabilidade do uso da problematização e da coleta de dados para identificação da motivação. Na etapa seguinte foi realizado um experimento que permitiu a coleta de dados e a definição dos atributos utilizados nas redes. O processo de elaboração do modelo se baseia nas etapas da descoberta de conhecimento em bancos de dados (TAN, STEINBACH, KUMAR, 2009). A figura 2 mostra as etapas do processo.

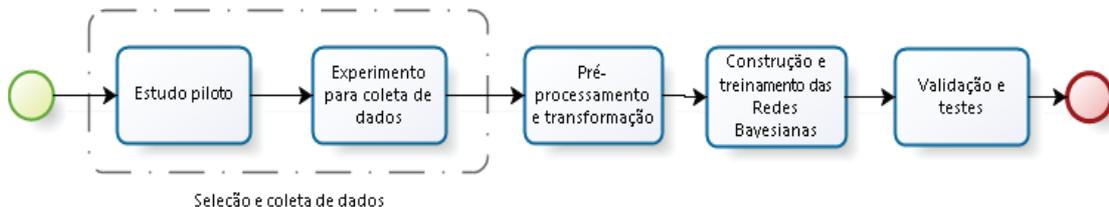


Figura 2. Etapas do processo de desenvolvimento do modelo afetivo. Fonte: Do autor.

Seleção e coleta de dados

A coleta dos dados foi realizada durante as atividades desenvolvidas em três turmas da disciplina de Algoritmos e Programação que compõe a matriz de diferentes cursos de graduação na Universidade do Vale do Taquari, localizada em Lajeado-RS. A disciplina se constitui no primeiro componente curricular de programação cursado pelos estudantes. Para realização dos exercícios foi adotado um método baseado na problematização com o Arco de Maguerez (BERBEL, 2012).

Em cada atividade foi apresentada uma descrição do contexto do problema (figura 3). Na definição dos pontos chave o estudante foi estimulado a compreender e pensar nos recursos e elementos que poderia utilizar para construir a solução. A hipótese deveria apresentar uma sequência de passos, blocos, indicando uma solução em alto nível, sem a utilização de instruções de programação. Nesta etapa o aluno tinha a disposição, dicas e a listagem do próprio pseudocódigo, como recurso de ajuda. O acesso a este conteúdo era opcional, a decisão de buscar auxílio foi delegada ao próprio aluno.

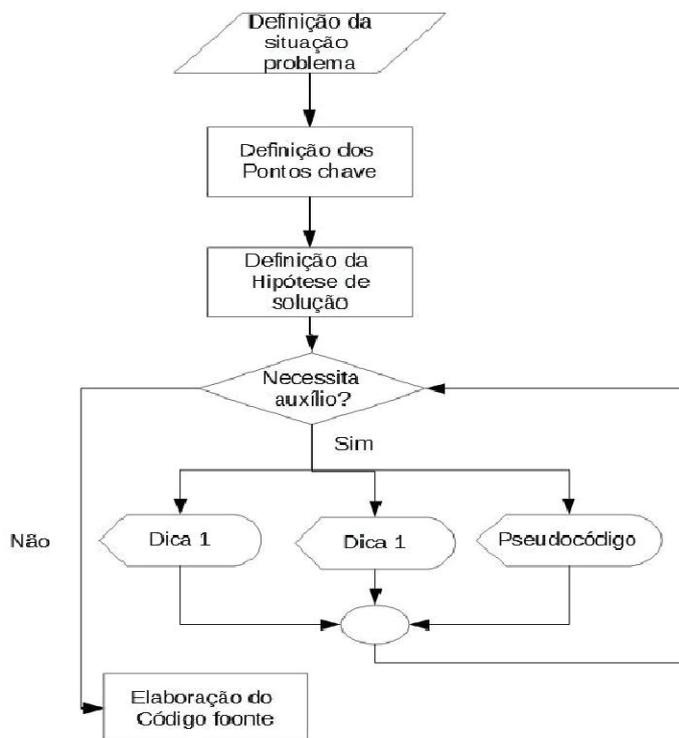


Figura 3. Fluxo do método de resolução das atividades. Fonte: Do autor.

O experimento para coleta de dados dividido duas fases, cada um com uma turma distinta, em semestres consecutivos, no ano de 2017. A tabela 1 mostra o número de participantes, o número de atividades propostas pelo professor e o total de atividades entregues. A definição dos dados que seriam coletados teve por base os modelos propostos por Du Boulay e Del Soldato (2016). Em cada atividade foram coletados os tempos do problema, da hipótese e do código, além dos conteúdos das respostas dos estudantes. As respostas foram avaliadas para determinar níveis de detalhe compreensão. Os demais dados dizem respeito a trajetória do estudante, de acessou dicas, se abandonou e retomou a atividade.

Etapa	Número de participantes	Atividades propostas pelo professor	Número de atividades entregues
Estudo piloto	14	8	84
Coleta de dados: fase 1	28	11	278
Coleta de dados: fase 2	26	7	147

Tabela 1. Número de participantes e atividades. Fonte: Do autor.

Resultados

Nesta seção serão apresentadas as atividades desenvolvidas na etapa de pré-processamento e os resultados da construção e treinamento das redes propostas. Com o objetivo de demonstrar o reconhecimento do esforço e da confiança serão apresentados os resultados da validação das RB sobre os dados de atividades realizadas em outra turma de estudantes. Cabe salientar que as entradas para o processo de treinamento das redes são os registros de cada exercício resolvido, ou seja, os tempos e as

respostas de cada estudante, para cada atividade. Tal abordagem corrobora com a abordagem de Silva et al (2016) que apresenta os fundamentos da mineração de dados orientada por atividades, contrapondo o modelo holístico que propõe a análise dos dados em nível de disciplina ou curso.

Pré-processamento e transformação

O pré-processamento consistiu na integração dos dados coletados nas três etapas e na transformação destes dados visando o treinamento das redes. Foi possível observar que alguns estudantes não finalizaram ou nem iniciaram determinados exercícios, o que acarretou no descarte destes registros. O número de entradas que foram consideradas para o treinamento foi de 509 (84+278+147), o que corresponde ao número total de entregas descritas na tabela 1.

Nome	Descrição	Regra de transformação
Classe_tp_prob_ef	Classe associada ao tempo para definição dos pontos chave do problema	Se o tempo está no 1º. Quartil, é considerado baixo , se está no 3º. Quartil, é alto . Os demais casos são considerados como médio .
Classe_tp_hip_ef	Classe associada ao tempo para definição da hipótese.	Utiliza a mesma regra da variável classe_tp_prob_ef.
Classe_cod_f	Classe associada ao tempo para escrita do código	Utiliza a mesma regra da variável classe_tp_prob_ef.
Classe_prop_ph	Proporção do tempo gasto na definição dos pontos chave e hipótese em relação ao código.	Se a soma dos tempos do problema e hipótese é superior a 80% do tempo do código ou inferior a 20%, é considerado desproporcional , senão é proporcional .
Execucao_codigo	Indica se o estudante executou o código ou não.	A partir dos dados que indicam o número de compilações e tentativas de execução, sem erros, é obtida a informação S(sim) ou N(não).
Visualizou_pseudo	Indica se o estudante acessou o conteúdo do pseudocódigo.	Dados coletados permitem saber se o estudante acessou o pseudocódigo, S(sim) ou N(não).
Classe_nivel_detalle_prob	Avaliação do professor sobre o detalhamento dos pontos chave do problema.	O professor avalia a resposta relativa aos pontos chave e define como alto , médio ou baixo o nível de detalhamento.
Classe_nivel_detalle_hip	Avaliação do professor sobre o detalhamento da hipótese.	O professor avalia a resposta relativa a hipótese e define como alto , médio ou baixo o detalhamento.
Classe_nivel_Compreensao	Avaliação do professor sobre a compreensão demonstrada.	O professor avalia se o estudante demonstrou alto , médio ou baixo nível de compreensão.
Classe_tp_prob_cf	Classe usada para indicar a confiança. Relacionada ao tempo para definição dos pontos chave.	A média dos tempos de cada atividade é calculada. Se o tempo do estudante for superior ao dobro da média, é considerado acima do normal, caso contrário a classe é normal .
Classe_tp_hip_cf	Indica confiança. Relacionado ao tempo da hipótese.	Utiliza a mesma regra de transformação da classe, classe_tp_prob_cf.

Avaliacao_estudante	Avaliação do estudante sobre o grau de dificuldade da atividade.	São apresentadas três opções para o estudante, fácil , médio ou difícil e este seleciona a alternativa de acordo com a sua avaliação.
Visualizou_dica1 e Visualizou_dica2	Indicam se o estudante acessou os conteúdos das dicas.	Dados coletados permitem saber se o estudante acessou a dica 1 e a dica 2, S(sim) ou N(não) .
Classe_retomadas	Número de vezes que o aluno iniciou, abandonou e reiniciou a atividade.	Se a contagem de reinícios for maior do que 2, foram muitas retomadas, se for 1 ou 2, poucas e 0, nenhuma .

Tabela 2. Variáveis utilizadas nas redes bayesianas. Fonte: Do autor.

A seguir foi realizada uma análise dos atributos, especialmente dos tempos para definição dos pontos chave, da hipótese e da escrita do código. Devido a necessidade de converter estas variáveis que estavam representadas por dados numéricos, contínuos para categorias, foi necessário definir um conjunto de regras de conversão. Os dados numéricos deram origem a diversas variáveis, mostradas na tabela 2.

Variáveis derivadas dos tempos são usadas tanto para identificar esforço, quanto confiança. Com relação ao esforço, tempos altos podem indicar maior dedicação, portanto são mais positivos. Na identificação da confiança, tempos acima do normal podem estar associados a certa confusão e menor confiança. Os níveis de detalhe decorrentes da avaliação do professor indicam que o estudante tentou especificar mais e se esforçou para demonstrar a sua compreensão do problema, o que de certa forma está relacionado ao número de tentativas utilizado no modelo de Du Boulay e Del Soldato (2016).

Os acessos às dicas e pseudocódigo, bem como o número de retomadas são utilizados como indicativo de menor confiança ou independência. No caso do pseudocódigo, o acesso a este recurso também pode demonstrar que o aluno buscou uma representação da solução antes mesmo de tentar elaborar a solução final, sinal de menor esforço.

Rede Bayesiana para reconhecimento do esforço por atividade

Após definir o conjunto de variáveis que compõe cada modelo, foi necessário classificar as entradas, uma vez que o método de RB utiliza classificação supervisionada e deve existir um atributo que seja a classe. Foi realizada uma classificação a priori, baseada em uma pontuação associada a cada valor, para cada variável.

No caso das variáveis que representam os tempos, níveis de detalhe e compreensão, para cada entrada foi atribuído uma pontuação positiva (+1), para o melhor caso (alto) ou negativa (-1) para o pior caso (baixo). Casos de valores médios foram considerados neutros e receberam valor 0. Na visualização do pseudocódigo, o acesso tem pontuação negativa e na execução de código, a não execução recebe -1, o mesmo que ocorre com a desproporcionalidade da variável Classe_prop_ph. Após a atribuição da pontuação, calcula-se a soma para cada entrada e se a mesma for negativa, o nível de esforço (variável Nivel_esforco) para a atividade é classificado como baixo. Se o total for maior ou igual a 0, a classe é alto.

A estrutura da rede foi elaborada manualmente, sendo o principal objetivo do processo de treinamento, a obtenção de um modelo probabilístico que permitisse classificar corretamente as entradas. O principal resultado do treinamento foi a definição das probabilidades associadas a cada ligação, por exemplo, a probabilidade de que o valor da variável Classe_tp_prob_ef seja baixo, dado que o nível de esforço é alto ou baixo. Para avaliar se o algoritmo é capaz de classificar corretamente as entradas, a ferramenta Weka (<https://www.cs.waikato.ac.nz/ml/weka>) foi usada para aplicar o método BayesNet ao conjunto de dados de treinamento. A figura 4 mostra que a acuracidade foi superior a 92% e que as medidas precision, recall e ROC também apresentam dados positivos, o que atesta a robustez e confiabilidade do modelo obtido.

```

Correctly Classified Instances      469          92.1415 %
Incorrectly Classified Instances    40           7.8585 %
Kappa statistic                    0.8371
Mean absolute error                0.1134
Root mean squared error            0.2244
Relative absolute error            24.1217 %
Root relative squared error        46.2884 %
Total Number of Instances         509

=== Detailed Accuracy By Class ===

          TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
          0,896   0,036   0,976     0,896   0,934     0,842   0,988    0,993    BAIXO
          0,964   0,104   0,849     0,964   0,902     0,842   0,988    0,983    ALTO
Weighted Avg.   0,921   0,062   0,928     0,921   0,922     0,842   0,988    0,989

=== Confusion Matrix ===

  a  b  <-- classified as
284 33 |  a = BAIXO
 7 185 |  b = ALTO
    
```

Figura 4. Resultados da aplicação do método *BayesNet* para o esforço. Fonte: Do autor.

Após a avaliação da classificação, deu-se a construção da rede mostrada na figura 5. O software Netica (<https://www.norsys.com/netica.html>) foi empregado na construção e treinamento das redes. Todas as variáveis foram modeladas como dependentes da variável que corresponde ao nível de esforço, uma vez que o principal objetivo do modelo é obter um modelo capaz de determinar o esforço em cada tarefa. Os percentuais mostrados ao lado das categorias em cada atributo correspondem a distribuição dos dados utilizados no treinamento.

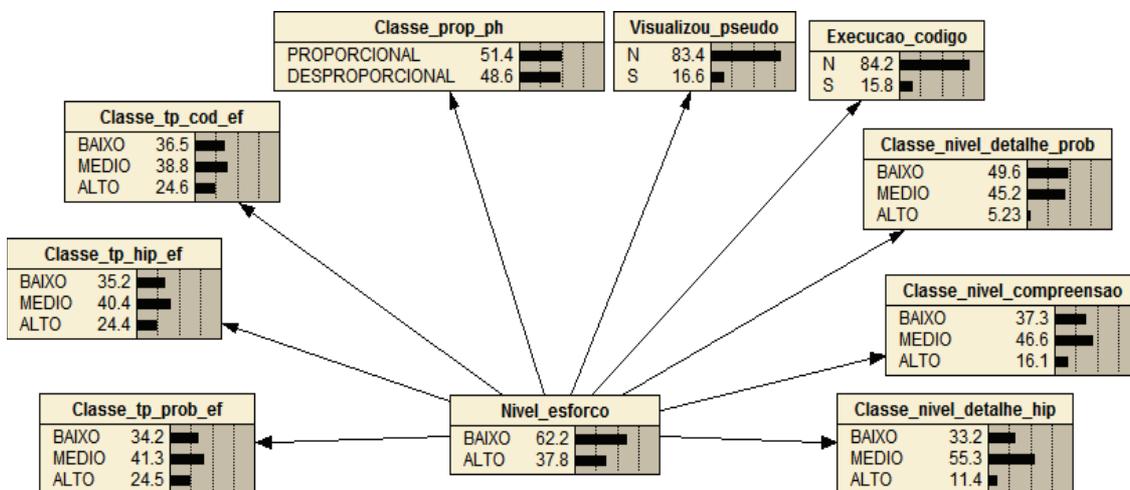


Figura 5. Rede Bayesiana para o reconhecimento do esforço por atividade. Fonte: Do autor.

Rede Bayesiana para reconhecimento da confiança por atividade

O desenvolvimento do modelo para identificação da confiança seguiu os mesmos passos descritos para o reconhecimento do esforço. A aplicação do método na ferramenta Weka apresentou resultados igualmente robustos para a classificação do nível de confiança (Figura 6), com uma acuracidade superior

a 94%.

Correctly Classified Instances	479	94.1061 %
Incorrectly Classified Instances	30	5.8939 %
Kappa statistic	0.8768	
Mean absolute error	0.1257	
Root mean squared error	0.218	
Relative absolute error	25.9654 %	
Root relative squared error	44.3223 %	
Total Number of Instances	509	

=== Detailed Accuracy By Class ===

	TP Rate	FP Rate	Precision	Recall	F-Measure	MCC	ROC Area	PRC Area	Class
	0,890	0,023	0,964	0,890	0,925	0,879	0,989	0,985	BAIXO
	0,977	0,110	0,927	0,977	0,951	0,879	0,989	0,992	ALTO
Weighted Avg.	0,941	0,074	0,942	0,941	0,941	0,879	0,989	0,989	

=== Confusion Matrix ===

```

a  b  <-- classified as
186 23 | a = BAIXO
 7 293 | b = ALTO
    
```

Figura 6. Resultados da aplicação do método BayesNet para a confiança. Fonte: Do autor.

A figura 7 mostra a estrutura da rede para a identificação do nível de confiança e a distribuição dos valores para cada atributo após o treinamento. A arquitetura do modelo para o reconhecimento da confiança segue os padrões da rede para reconhecimento do esforço, onde todos atributos dependem da variável que determina as categorias para a confiança do estudante.

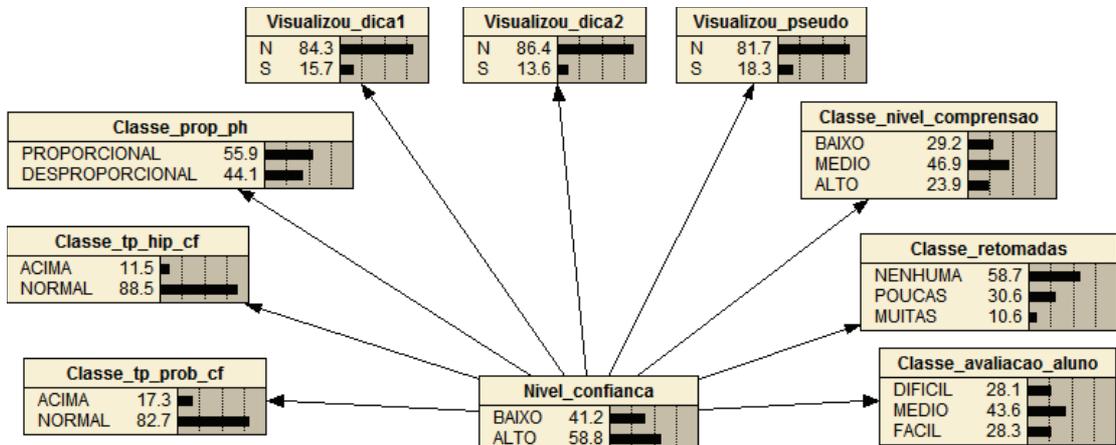


Figura 7. Rede Bayesiana para o reconhecimento da confiança por atividade. Fonte: Do autor.

Validação do modelo afetivo

Os testes e a validação do modelo afetivo foram realizados no segundo semestre de 2018. As atividades foram desenvolvidas em uma turma da disciplina de Algoritmos e Programação, composta por 25 estudantes, oriundos dos cursos de Engenharia de Software, Engenharia de Computação e Sistemas de Informação. Os resultados apresentados se baseiam em um conjunto de 6 exercícios propostos pelo professor e resolvidos pelos estudantes. Os percentuais mostrados na tabela 3 correspondem a probabilidade da confiança e esforço serem categorizados como alto (verde), o que caracteriza os casos

com percentuais inferiores a 50 como baixo (vermelho).

Aluno	Atividade 1		Atividade 2		Atividade 3		Atividade 4		Atividade 5		Atividade 6	
	Conf.	Esf.										
Aluno1	78,59	13,18	56,48	24,54	16,90	2,70	88,29	78,80	88,29	98,03	2,96	95,40
Aluno2	88,41	99,97	92,71	29,97	82,66	58,92	61,27	31,69	0,04	92,13	49,98	100,00
Aluno3	99,37	99,77	92,03	99,99	99,07	100,00	96,84	100,00	99,76	100,00	97,55	99,20
Aluno4	0,00	97,09			0,05	5,54			10,88	1,19		
Aluno5							11,99	61,27	96,18	8,85	1,72	81,69
Aluno6	99,37	99,56	99,37	93,72	99,01	47,80	97,55	94,78				
Aluno7	97,55	99,58	99,63	100,00	99,37	93,72	71,13	100,00	97,55	55,97	87,26	83,63
Aluno8	98,13	42,33	90,04	0,06	97,08	4,02	69,43	0,10	58,97	0,00	76,39	4,01
Aluno9	76,39	25,07	87,26	86,78	92,95	73,60	1,46	93,72	56,59	97,48		
Aluno10	99,07	99,94	76,39	99,69	95,26	100,00	98,53	100,00	98,53	100,00	99,07	100,00
Aluno11	97,14	99,80	91,99	99,84	0,35	42,71	8,46	75,50	0,01	42,71	4,32	0,06
Aluno12	99,37	84,49	66,94	98,23	76,39	97,20	9,46	0,76			87,26	96,35
Aluno13			97,55	99,73	0,01	0,03			18,33	60,29		
Aluno14	0,49	15,18	26,90	18,23	0,05	0,04	1,41	0,04	19,25	0,03	0,22	22,27
Aluno15	87,90	99,31	87,90	93,39	91,99	99,44	0,45	10,00	99,37	95,58		
Aluno16	0,157	14,60	74,10	47,96	27,36	2,59					0,19	5,85
Aluno17	97,55	95,58							45,09	86,68		
Aluno18	99,76	100,00	99,07	100,00	92,71	100,00	97,55	99,91	99,07	100,00	97,55	99,73
Aluno19	99,76	99,98	84,37	100,00	76,39	91,05	94,52	77,18	99,01	99,77	35,78	96,57
Aluno20	98,53	99,93	97,55	87,01	58,97	0,06	16,50	96,63	13,73	94,40	35,53	85,23
Aluno21	16,50	81,59	98,13	8,17	99,01	99,29	76,19	96,22				
Aluno22	92,95	8,17	76,39	96,63	81,25	83,07	99,01	85,57	99,01	62,07	85,13	2,23
Aluno23	99,01	23,62	92,95	0,05	15,60	0,13	69,43	0,00				
Aluno24	99,01	94,09	99,07	100,00			99,01	99,77	99,37	94,40	85,54	99,97
Aluno25	96,18	97,73	89,61	99,99	16,90	2,29	51,76	2,29	89,30	2,29	45,09	85,22

Tabela 3. Probabilidades de confiança e esforço na validação do modelo. Fonte: Do autor.

O valor de 78,59 mostrado na primeira atividade para o aluno 1, por exemplo, mostra que a RB classificou a confiança como alta com 78% de certeza. Em sentido oposto, há uma probabilidade próxima de 22% da confiança ser baixa. É possível verificar que algumas tarefas não foram finalizadas o que acarretou em uma impossibilidade de determinar os níveis de motivação.

Percebe-se que em grande parte dos casos, os valores se aproximam de 100 ou ficam próximos a 0

(zero), o que mostra que o modelo apresenta alto grau de certeza na determinação dos níveis de confiança e esforço. Entretanto, há outras situações onde existe uma tendência maior para uma das categorias, mas não é possível determinar claramente a classe dominante, como no caso da confiança associada a atividade 6 e ao estudante 2, cujo percentual foi 49,98. Neste exemplo há 50% de chance de ser alto e praticamente o mesmo percentual de ser baixo, o que caracteriza um baixo grau de certeza na classificação do nível de confiança para a atividade. Este é um dos aspectos importantes de uma abordagem probabilística, a determinação de uma classe com uma probabilidade associada.

A figura 8 ilustra os valores dos atributos e as respostas das RB para o caso da atividade 4 desenvolvida pelo estudante de código 11. É possível constatar a influência de alguns atributos nos níveis de confiança e esforço. O fato de ter muitas retomadas, aliado ao acesso ao pseudocódigo é decisivo para que o nível de confiança seja baixo, mesmo que nas demais variáveis, como nos tempos, os valores sejam os desejáveis. No caso do esforço, alguns atributos apresentam valores negativos, como os tempos do código e da hipótese, porém a ocorrência de valores médios nos níveis de detalhe e compreensão contribui para aumentar o nível de esforço, que foi definido como alto com uma probabilidade de 75,5%.

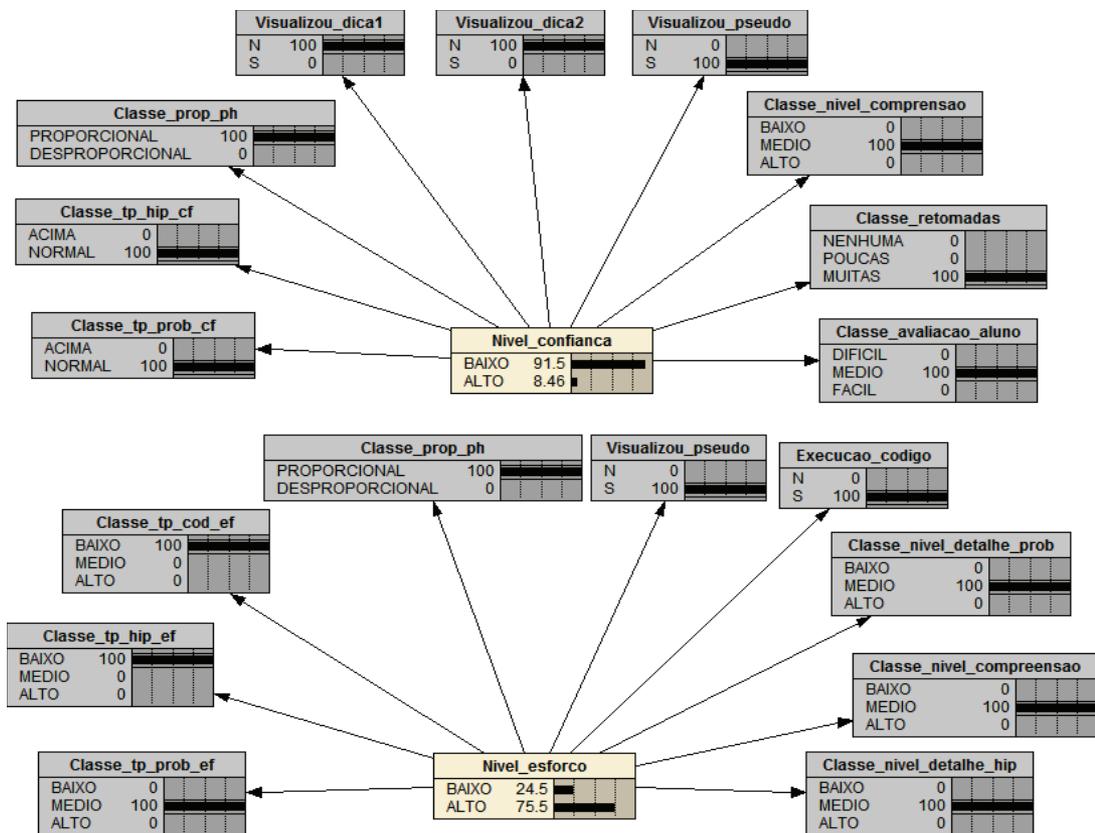


Figura 8. RB correspondente a atividade 4 do estudante 11. Fonte: Do autor.

Conclusão

O principal objetivo deste trabalho foi a apresentação de um modelo baseado em Redes Bayesianas para o reconhecimento da motivação dos estudantes na realização de atividades de programação. Os resultados demonstram que o modelo apresenta confiabilidade, boa acuracidade e permite identificar os níveis de esforço e confiança em cada atividade. Observando os dados obtidos na validação, percebe-se

que em muitas situações o esforço foi considerado baixo, embora a confiança do estudante fosse alta, enquanto em outros casos maiores índices de esforço também estão relacionados à maior confiança, sendo necessário avaliar os dois fatores em conjunto para a tomada de ações que estimulem o esforço e aumentem a confiança ou independência do aluno.

Uma das características fundamentais do modelo é a possibilidade de determinar a categoria com um grau de certeza, o que contribui para que sejam priorizados nas ações aqueles estudantes que exibem probabilidades de confiança e esforço mais baixas. Ao longo das atividades é possível verificar se o estudante apresenta baixos níveis de forma sistemática em todas as tarefas ou se este comportamento é pontual, em uma ou outra atividade.

Trabalhos futuros incluem a incorporação do modelo como um módulo para reconhecimento automático da motivação em um software se encontra em desenvolvimento. O software já permite a coleta e transformação dos dados, sendo necessário incorporar as Redes Bayesianas para o reconhecimento do esforço e da confiança. O referido módulo pode contribuir para que os docentes percebam de forma mais rápida os contextos e estudantes que demandam uma maior atenção, possibilitando a inserção de ações que melhorem o aprendizado.

Referências

BERBEL, Neusi Aparecida Navas. As metodologias ativas e a promoção da autonomia de estudantes. *Semina: Ciências Sociais e Humanas*, v.32, n.1, p.25-40, 2012.

DU BOULAY, Benedict; DEL SOLDATO, Teresa. Implementation of Motivational Tactics in Tutoring Systems: 20 years on. *International Journal of Artificial Intelligence in Education*, v. 26, n. 1, p. 170-182, 2016.

DUMDUMAYA, Cristina E.; BANAWAN, Michelle P.; RODRIGO, Ma Mercedes T. Identifying Students' Persistence Profiles in Problem Solving Task. In: *Adjunct Publication of the 26th Conference on User Modeling, Adaptation and Personalization*. ACM, 2018. p. 281-286.

GIRAFFA, Lucia Maria Martins; MÓRA, Michael da Costa. Evasão na disciplina de algoritmo e programação: um estudo a partir dos fatores intervenientes na perspectiva do aluno. In: *Conferência Latino americana sobre o abandono na educação superior*. 2013.

JUNIOR, Ademar Crotti; WILGES, Beatriz; NASSAR, Silvia Modesto. Modelagem bayesiana da aprendizagem de estudantes em um AVA. In: *Brazilian Symposium on Computers in Education (SBIE)*. 2014. p. 772.

LONGHI, Magali T.; BEHAR, Patricia A.; BERCHT, Magda. Máquina de inferência baseada na teoria bayesiana para identificar os estados de ânimo do aluno em um ambiente virtual de aprendizagem. *RENOTE*, v. 8, n. 3, 2010.

PICARD, Rosalind W.; PICARD, Rosalind. *Affective computing*. Cambridge: MIT press, 1997.

RAMOS, Vinicius et al. A Comparação da Realidade Mundial do Ensino de Programação para Iniciantes com a Realidade Nacional: Revisão sistemática da literatura em eventos brasileiros. In: *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*. 2015. p. 318.

SILVA, João Carlos Sedraz et al. Mineração de Dados Educacionais Orientada por Atividades de Aprendizagem. *RENOTE*, v. 14, n. 1, 2016.

TAN, Pang-Ning; STEINBACH, Michael; KUMAR, Vipin. Introdução ao Datamining: mineração de dados. Rio de Janeiro: Ciência Moderna, 2009.

TWENGE, Jean; CAMPBELL, Stacy. Generational differences in psychological traits and their impact on the workplace. *IEEE Engineering Management Review*, v. 39, n. 2, p. 72-84, 2011.

VIER, Juliano; GLUZ, João; JAQUES, P. Empregando redes bayesianas para modelar automaticamente o conhecimento dos aprendizes em lógica de programação. *Revista Brasileira de Informática na Educação*, v. 23, n. 2, 2015.

WATSON, Christopher; LI, Frederick WB. Failure rates in introductory programming revisited. In: *Proceedings of the 2014 conference on Innovation & technology in computer science education*. ACM, 2014. p. 39-44.