

## DESENVOLVIMENTO DE APLICAÇÃO WEB PARA APOIAR O CONTROLE DE EVASÃO

Richard Ribeiro Silva<sup>1</sup>, Gilson Saturnino dos Santos<sup>1</sup>

<sup>1</sup>Instituto Federal do Mato Grosso do Sul– Coxim-MS

richard.cxm@hotmail.com.br, gilson.santos@ifms.edu.br

### Resumo

Pode-se definir evasão como sendo a saída permanente de algum curso, evitando assim a conclusão do mesmo, ou seja, todo aquele que inicia um curso e não termina está automaticamente evadido. O Aprendizado de Máquina entra como uma possível solução, podendo auxiliar a identificar quais os alunos que irão evadir, fazendo assim com que a instituição tenha mais chance atuar nos estudantes ainda matriculados. Neste trabalho foi desenvolvido um *Web Service Rest* com modelos de classificação J48, SMO, *Multilayer Perceptron* e Naive-Bayes, que permitirá ao Instituto Federal de Mato Grosso do Sul (IFMS) identificar antecipadamente estudantes de cursos superiores que irão evadir podendo atuar para evitar a desistência.

**Palavras-chave:** Evasão, Aprendizado de Máquina, Web Service.

### Introdução

A evasão vem se tornando um grande problema para as instituições de ensino, que por sua vez vem diminuindo a quantidade de estudantes nas salas de aula. Em todas as situações a evasão é algo negativo tanto para instituição quanto para o aluno (CARDOSO; LUDOVICO, 2017).

O Aprendizado de Máquina (AM) é uma subárea da Inteligência Artificial, que utiliza sistemas capazes de aprender a partir de fatos que já ocorreram e assim criar um modelo que consiga classificar novos exemplos (AMARAL, 2016).

A identificação de quem possui tendência a evadir do curso por meio do AM pode proporcionar maiores chances para as instituições de ensino realizarem ações específicas com esses alunos, evitando assim a evasão do mesmo.

*Web Service* é uma tecnologia de integração que é utilizada para criar software ou componente de software que tem a finalidade de interagir, tanto enviando quanto recebendo informações vinda de outros sistemas, independentemente da linguagem a qual foi desenvolvida (GOMES, 2014). *Representational State Transfer* (REST) é uma arquitetura de *Web Service* que utiliza o protocolo *Hyper Text Transfer Protocol* (HTTP) para enviar e receber dados. *Web Service Restful* utiliza o método *Get* do protocolo HTTP para

consultar, *Post* para inserir, *Put* para atualizar e *Delete* para excluir um recurso (LECHETA, 2015).

O objetivo deste trabalho foi utilizar modelos de classificação (SANTOS, G.H. E SANTOS G.S., 2018 no prelo) e a biblioteca Weka (AMARAL, 2016) para criar um *Web Service Restful*. Este *Web Service* recebe e retorna dados de questionários socioeconômicos no formato *Java Script Object Notation* (Json) (LECHETA, 2015; DEITEL P. E DELTEL H., 2011). Os dados recebidos são convertidos para *Instances* da biblioteca Weka e classificados (AMARAL, 2016). Esse *Web Service* permitirá que o IFMS identifique antecipadamente estudantes de cursos superiores que irão evadir, podendo atuar para evitar a desistência.

### Metodologia

No início do trabalho foram realizadas revisões bibliográficas sobre evasão escolar e Aprendizado de Máquina. Em seguida foi realizado uma pesquisa sobre *Web Service Rest* e a utilização da biblioteca Weka.

Posteriormente foi desenvolvido um *Web Service Rest*, utilizando a Linguagem Java, o servidor Web Tomcat, o *framework* Jersey (LECHETA, 2015; DEITEL P. E DELTEL H., 2011) e o ambiente de desenvolvimento Netbeans. Os testes foram realizado por meio da ferramenta SoapUI (LECHETA, 2015). Para carregar e utilizar os modelos de classificação foi utilizado a biblioteca do Weka (AMARAL, 2016; WITTEN E FRANK, 2011).

### Resultados e Discussão

O *Web Service* desenvolvido possui um serviço que recebe uma lista de estudantes com suas respectivas respostas do questionário socioeconômico do IFMS. Este serviço utiliza a biblioteca Weka para classificar os estudantes em alunos que podem evadir e alunos que não irão evadir. Após a classificação, o *Web Service* retorna a lista de estudantes com suas respectivas classificações.

Conforme o diagrama de classes representado pela Figura 1, a classe Java *Aluno* possui 50 atributos privados. Esses atributos representam as informações do questionário

socioeconômico e o rótulo de classe (WITTEN E FRANK, 2011). Também há 50 métodos *get* (retorno de dados) e 50 métodos *set* (definição de dados), tendo no total 100 métodos. A classe Java *ListaAluno* é utilizado para empacotar uma lista de objetos do tipo *Aluno* (DEITEL P. E DEITEL H., 2011).

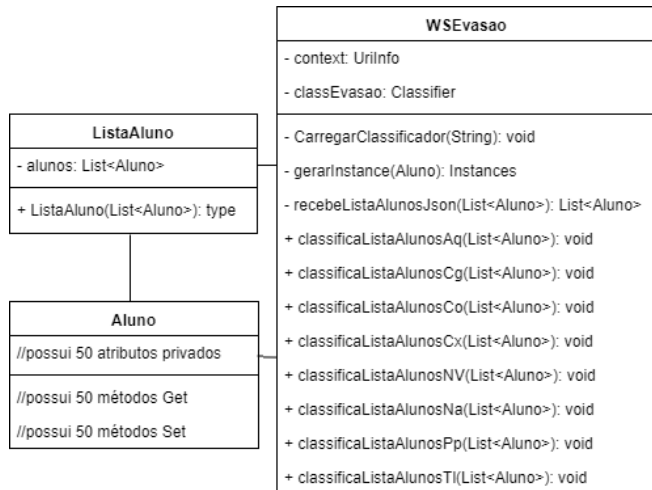


Figura 1. Diagrama de Classes implementado na linguagem Java. Fonte: Autoria Própria.

A classe Java *WSEvasao* é responsável por receber as requisições do *Web Service* e realizar as classificações. Esta classe Java possui o método privado chamado *CarregarClassificador*, utilizado para carregar o modelo de classificação específico para cada campus do IFMS. Esse método recebe por parâmetro a sigla da cidade que se deseja realizar a classificação de estudantes.

O método privado *gerarInstancia* é responsável por converter objetos do tipo *Aluno* em objetos do tipo *Instances*, classe Java da biblioteca Weka. Os métodos públicos da classe Java *WSEvasao* serão utilizados pelas aplicações cliente por meio de endereços específicos para cada campus. Ou seja, para classificar os alunos de um determinado campus, a aplicação cliente irá utilizar o endereço para o campus do usuário que está requisitando a informação. Cada um destes métodos públicos carrega o respectivo modelo de classificação e envia os dados para o método *recebeListaAlunosJson*.

O método *recebeListaAlunosJson* utiliza o método *gerarInstancia* e classifica os estudantes em evadido ou não evadido. As classe do Weka utilizadas foram *Classifier*, *DenseInstance*, *FastVector*, *Attribute*, *Instance* e *Instances*.

Na Figura 2 é apresentado uma parte do conteúdo do Json retornado pelo *Web Service* desenvolvido. O campo situação recebeu a classificação. No caso da Figura 2, o estudante foi classificado como *Não Evadido*.

```

"pViaRadio": "Sim",
"pWifi": "Sim",
"problemaDeVisao": "Sim, usa óculos e/ou lentes de contato"
"qtdeFilhos": 1,
"raca": "BRANCA",
"responsavel1": "Pai",
"responsavel2": "Mãe",
"sexo": "Masculino",
"situacao": "Nao_Evadido",
"situacaoMoradia": "Própria",
"usaDeMedicacao": "Não"
  
```

Figura 2. Exemplo de retorno de dados com a classificação realizada. Fonte: Autoria Própria.

### Considerações Finais

O IFMS poder criar ou alterar aplicações para que enviem dados socioeconômicos dos estudantes de cursos superiores para o *Web Service* desenvolvido, identificando quais irão evadir. Ao identificar esses estudantes o IFMS poderá desenvolver ações específicas para os mesmos, evitando sua desistência. Como trabalhos futuros pode-se criar novos modelos de classificação, com melhores acurácias, aumentando assim a confiança nas classificações.

### Agradecimentos

Programa de Iniciação Científica e Tecnológica do IFMS.

### Referências

- AMARAL, Fernando. **Aprenda Mineração de Dados: Teoria e Prática**. Rio de Janeiro: Alta Books, 2016. 220 p.
- CARDOSO, Daniela Freire; LUDOVICO, Nelson. **Estudo Longitudinal Sobre As Pesquisas De Evasão No Ensino Superior**: Diretório IBICT. Refaz: Revista Fatec Zona Sul, São Paulo, v. 3, n. 4, p.01-18, jun. 2017.
- DEITEL, Paul; DEITEL, Harvey. **Java como programar**. 8. ed. São Paulo: Pearson, 2011. 1144 p.
- GOMES, Daniel Adorno. **Web Services Soap em Java**. 2. ed. São Paulo: Nova, 2014. 198 p.
- LECHETA, Ricardo R. **Web Services RESTful: Aprenda a criar Web Services RESTful em Java na nuvem do Google**. São Paulo: Novatec, 2015. 432 p.
- Santos, Gabrielle Helpis; Santos, Gilson Saturnino. **Aprendizado de Máquina com Dados Socioeconômicos para Apoiar o Controle de Evasão**. Seminário de Iniciação Científica do IFMS, 2018. No prelo.
- WITTEN, Ian H.; FRANK, Eibe; MARK, A. Hall. 2011. **Data mining: Practical machine learning tools and techniques**, 2011.