Desenvolvimento de uma aplicação para a predição de desempenho de candidatos do Enem

Guilherme Dal Más[1], **Roger Sá da Silva**[1]

[1]Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Sul (IFRS)
Campus Veranópolis - RS - Brasil

dalmasguilherme58@gmail.com

Resumo. *Este trabalho propõe o desenvolvimento de uma aplicação Java para a descoberta de padrões no perfil de candidatos do Enem do ano de 2021. De maneira mais específica, pretende-se desenvolver um servlet com um modelo classificador para predizer o desempenho dos estudantes nas provas do Enem baseado nos dados do perfil demográfico, utilizando a média das notas da prova como métrica para esta classificação com objetivo de trazer a realidade das questões socioeconômicas dos alunos que participaram do Enem.*

Abstract. *This work proposes the development of a Java application for the discovery of patterns in the profile of candidates for the Enem of the year 2021. More specifically, it is intended to develop a servlet with a classifier model to predict the performance of students in the Enem tests based on data from the demographic profile, using the average of the test scores as a metric for this classification in order to bring the reality of the socioeconomic issues of the students who participated in the Enem.*

1. Introdução

A pesquisa em descoberta do conhecimento em base de dados tem crescido e atraído esforços, baseada na disseminação da tecnologia de bancos de dados e na premissa de que as grandes coleções de dados hoje existentes podem ser fontes de conhecimento útil, o qual está implicitamente representado e por ser extraído [Gonçalves e Freitas, 2002].

A mineração de dados é uma área de estudos multidisciplinar cujo objetivo é a extração de informações implícitas em bases de dados. As avaliações padronizadas em larga escala (standards) vêm ganhando força no Brasil e no mundo, possibilitadas por inovações tecnológicas que garantem a eficácia desses instrumentos como indicadores da qualidade do processo ensino aprendizagem em países e estados. Isso serve de referência para a cultura escolar e para políticas de incentivos e punições, ocupando um papel central nos atuais sistemas de educação [Travitzki, 2013].

Um dos temas mais pesquisados para a descoberta de padrões e informações implícitas em dados são os fatores que determinam a classificação dos registros em certas classes ou categorias, inclusive na área educacional e de processos seletivos. A partir disso, entende-se ser relevante o desenvolvimento de uma ferramenta computacional capaz de analisar quais características demográficas dos participantes do Exame Nacional do Ensino Médio (Enem) tem potencial de levar a um melhor ou pior desempenho, dada a

importância do resultado das provas para acesso ao ensino superior em universidades brasileiras.

Diante desse contexto, este trabalho propõe o desenvolvimento de uma aplicação Java para a descoberta de padrões no perfil de candidatos do Enem do ano de 2021. De maneira mais específica, pretende-se desenvolver um servlet com um modelo classificador para predizer o desempenho dos estudantes nas provas do Enem baseado nos dados do perfil demográfico, utilizando a média da soma das notas da prova como métrica para esta classificação.

O presente trabalho está dividido em seções. Na seção 2, aborda-se a fundamentação teórica sobre a mineração de dados, o classificador utilizado e um breve resumo do Enem. A seção 3 aborda a metodologia utilizada no trabalho, explicando detalhes do desenvolvimento da aplicação, ferramentas utilizadas e como foi realizada a preparação da base de dados. A seção 4 aborda os resultados obtidos com este trabalho, e por fim, a seção 5, as considerações finais.

2. Fundamentação Teórica

Conforme o avanço da tecnologia, em que a maioria das operações e atividades das instituições privadas e públicas são registradas computacionalmente, se acumulam em grandes bases de dados. A técnica de mineração de dados é uma das formas na qual é possível extrair conhecimento a partir de grandes volumes de dados, sendo capaz de descobrir padrões, predizer e correlacionar dados, que serão utilizados para auxiliar as instituições nas tomadas de decisões [Galvão et al., 2009].

2.1. Mineração de dados

A mineração de dados é uma área de estudos multidisciplinar que consiste na extração de informações implícitas em bases de dados através de soluções computacionais não triviais [Han e Kamber, 2001].

Mineração de dados, segundo Han e Kamber (2001), é um campo multidisciplinar que inclui as seguintes áreas: banco de dados, inteligência artificial, aprendizado de máquina, redes neurais, estatísticas, reconhecimento de padrões, sistemas baseados em conhecimento, aquisição de conhecimento, recuperação de informação, computação de alto desempenho e visualização de dados.

As técnicas de mineração de dados, de acordo com Rezende (2003), descrevem um paradigma de extração de conhecimento e vários algoritmos podem seguir esse paradigma, ou seja, para uma técnica pode-se ter vários algoritmos.

Conforme Galvão et al. (2009), a descoberta de conhecimento em base de dados é o processo de extração de informação a partir de dados registrados em uma base de dados, sendo um conhecimento implícito, que será útil e entendível. O processo de KDD utiliza vários conceitos, como base de dados, métodos estatísticos, ferramentas de visualização e técnicas de inteligência artificial.

Segundo Carvalho (2001), muitas vezes os termos “Mineração de dados” e “Descoberta de Conhecimento em Banco de Dados” são confundidos como sinônimos. Porém, o KDD refere-se ao processo completo de descoberta de conhecimento, composto

de várias etapas, conforme ilustrado na figura 1. Já a mineração de dados é uma de suas etapas voltada a aplicar algoritmos específicos e a produzir padrões sobre uma base de dados [FAYYAD et. al.,1996].

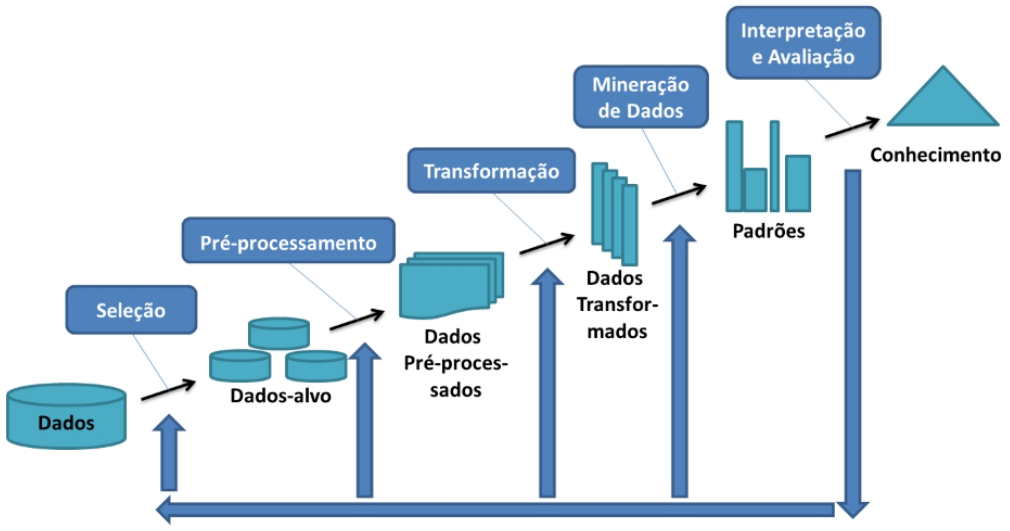


Figura 1. Visão geral das etapas do processo de Descoberta de Conhecimento em Banco de Dados - adaptada de [FAYYAD et. al. 1996]

O primeiro passo no processo de KDD é entender o domínio da aplicação, identificar o problema e definir os objetivos a serem atingidos. O processo inicia com os dados brutos e finaliza com a extração de conhecimento [MARTINHAGO, 2005].

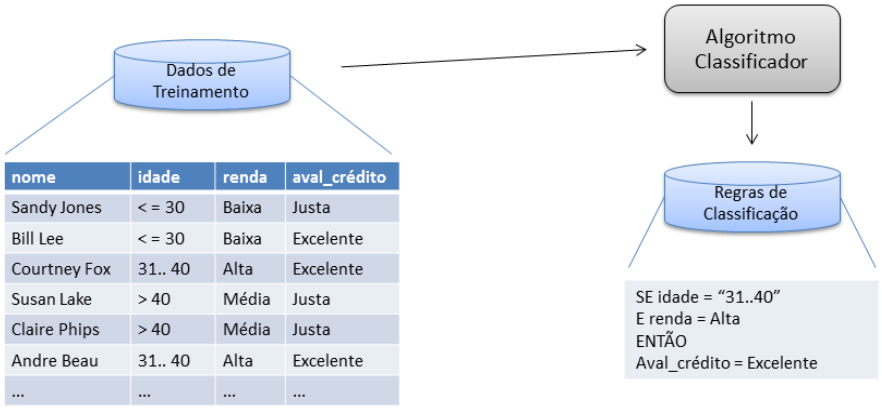
Na fase de seleção de dados ocorre a identificação do subconjunto das bases de dados existentes que deve ser efetivamente considerado durante o processo de KDD [GOLDSCHMIDT et al., 2015].

Na etapa de pré-processamento ocorre a limpeza dos dados, ou seja, as informações consideradas desnecessárias são removidas. Adotam-se estratégias para manusear dados faltantes ou inconsistentes, em conjunto com a etapa de transformação. Se os erros não forem descobertos neste estágio, poderão contribuir para a obtenção de resultados de baixa qualidade [MARTINHAGO, 2005].

Na mineração de dados, aplicam-se algoritmos para extrair padrões dos dados ou gerar regras que descrevam o comportamento da base de dados. Para isto, utiliza-se uma ou mais técnicas para se extrair o tipo de informação desejada. Durante esse procedimento, pode ser necessário acessar dados adicionais e/ou executar outras transformações nos dados originalmente selecionados [MARTINHAGO, 2005].

O modelo preditivo é definido como uma tentativa de descobrir o que acontecerá em um momento futuro. Em Mineração de Dados, essa tentativa é feita a partir de um modelo construído usando a base de dados como referência, conforme mostrado na figura 2. A base usada na construção do modelo é chamada de treinamento (base de treinamento) e, para o modelo preditivo, é necessário que ela tenha o atributo especial com a classe ou valor que se deseja prever. O processo de construção do modelo recebe o

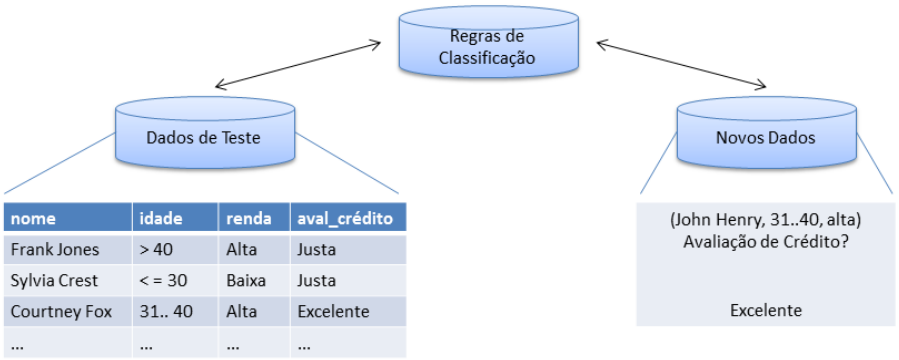
nome de aprendizado e, por ter o atributo especial como referência, designamos como sendo supervisionado [HAN et al., 2006; TAN et al., 2009; FACELI et al., 2011].



nome	idade	renda	aval_crédito
Sandy Jones	< = 30	Baixa	Justa
Bill Lee	< = 30	Baixa	Excelente
Courtney Fox	31.. 40	Alta	Excelente
Susan Lake	> 40	Média	Justa
Claire Phips	> 40	Média	Justa
Andre Beau	31.. 40	Alta	Excelente
...	...	...	...

Figura 2. Ilustração das atividades referentes à etapa de treinamento ou aprendizado em uma tarefa de classificação - adaptada de [Han et al., 2006]

Na etapa final de interpretação e avaliação, são realizadas análises dos resultados. As análises podem ser quantitativa ou qualitativa, e dependem da tarefa que foi escolhida. O modelo preditivo é construído por um conjunto de dados rotulados. Estes dados podem ser separados em dois conjuntos, sendo geralmente chamado de treinamento e outro de teste. O conjunto de treinamento usa-se para gerar o modelo e o conjunto de teste, por outro lado, para fazer análises quantitativas, conforme indica a figura 3. Ou seja, apresenta-se um exemplar do conjunto de teste ao modelo, verifica-se o resultado estimado e compara-se o resultado desejado para quantificar desempenho [Silva & Silva, 2014].



nome	idade	renda	aval_crédito
Frank Jones	> 40	Alta	Justa
Sylvia Crest	< = 30	Baixa	Justa
Courtney Fox	31.. 40	Alta	Excelente
...	...	...	...

Figura 3. Ilustração das atividades referentes à etapa de teste em uma tarefa de classificação - adaptada de [Han et al., 2006]

2.2. Classificador baseado em árvore de decisão

Possui este nome porque a sua estrutura se assemelha a uma árvore. A sua estrutura consiste em dividir os dados em subgrupos, com base nos valores das variáveis. O resultado é uma hierarquia de declarações do tipo "Se... então..." que são utilizadas, principalmente, quando o

objetivo da mineração de dados é a classificação de dados ou a predição de saídas. [MARTINHAGO, 2005]

A análise dos resultados obtidos é o método de classificação por Árvore de Decisão. Funciona como um fluxograma em forma de árvore, onde cada nó (não-folha) indica um teste feito sobre um valor. As ligações entre os nós representam os valores possíveis do testado nó superior, e as folhas indicam a classe a qual o registro pertence. Após a árvore de decisão montada, para classificarmos um novo registro, basta seguir o fluxo na árvore (mediante os testes nos nós não-folhas) começando no nó raiz até chegar a uma folha, conforme a figura 4 [SILVA, 2014].

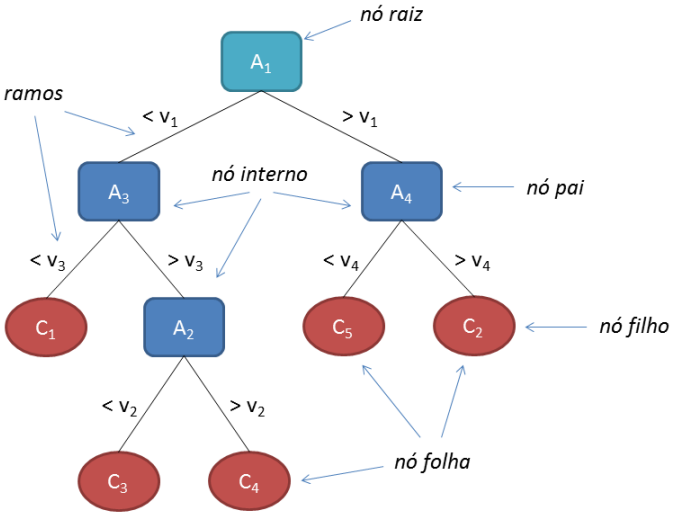


Figura 4. Árvore de Decisão hipotética que ilustra alguns conceitos em relação ao modelo - adaptada de [Han et al., 2006]

Pela estrutura que formam, as árvores de decisões podem ser convertidas em Regras de Classificação. O sucesso das árvores de decisão, deve-se ao fato de ser uma técnica extremamente simples, não necessita de parâmetros de configuração e geralmente tem um bom grau de assertividade. O algoritmo J48 é uma implementação do algoritmo de árvore de decisão C4.5 proposto por Quinlan (1993) para o Weka. O algoritmo J48 constrói a árvore de cima para baixo (top-down) com o objetivo de escolher sempre o melhor atributo para cada nó de decisão da árvore. É um processo recursivo que após ter escolhido um atributo para um nó, começando pela raiz, aplica o mesmo algoritmo aos descendentes desse nó, até que os critérios de parada sejam verificados

Dentre os algoritmos destacados por Dias (2001) para a construção de uma árvore de decisão, que são: CART (BERRY e LINOFF, 1997), CHAID (BERRY e LINOFF, 1997), ID3 (QUINLAN, 1983), C4.5 (QUINLAN, 1993), SLIQ (METHA et al, 1996) E SPRINT (SHAFER et al, 1996).

2.3. Trabalhos relacionados

O trabalho de Martinhago (2005) investigou a base de dados dos vestibulandos da UFPR em dezembro de 2003. Foi utilizado o software WEKA em conjunto com a técnica de mineração de dados árvore de decisão em conjunto com algoritmos de classificação J48 e

J48.PART com o objetivo de encontrar o perfil do vestibulando da Universidade Federal do Paraná. Na sua base de dados foram utilizadas informações coletadas do questionário socioeducacional preenchido pelos candidatos no momento da inscrição. Os dados do questionário eram compostos pelos dados gerais do candidato contendo o registro das notas obtidas pelo candidato nas provas e na redação, a opção pelo ENEM, a nota do ENEM, a média das notas do candidato e o resultado do vestibular para o aluno.

Já no trabalho de Silva (2014) utilizou-se dados dos questionários socioeconômicos preenchidos por alunos participantes do ENEM de 2010, localizados nas capitais da região Sudeste do Brasil. Foi utilizada a técnica de mineração de dados baseada em regras de associação, utilizando diferentes parametrizações do algoritmo A Priori. Os autores identificaram que fatores como a renda familiar baixa, escolaridade dos pais de nível primário e a quantidade alta de pessoas que moram com o estudante diminuem o desempenho dos alunos.

Por fim, Simon et. al (2017) apresentam um artigo que relaciona fatores socioeconômicos das escolas do ensino médio de todo o território nacional como indicador de desempenho médio em ciências da natureza e suas tecnologias. Para isto, foi utilizada a base de dados fornecida pelo INEP (Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira, responsável pela elaboração das provas) informando os valores médios de proficiência dos alunos agrupados por escola. Foi construído um modelo preditivo utilizado o algoritmo J48 em conjunto com o software Weka.

2.4. Enem - Exame Nacional do Ensino Médio

O Enem trata-se de uma prova aplicada individualmente em diversas instituições situadas em todo o território nacional, na qual participam milhões de estudantes, obtendo um resultado que varia numa nota de 0 a 1000. Além da prova, há um questionário socioeconômico com 25 questões.

É uma prova única aplicada anualmente, composta por 180 perguntas objetivas de múltipla escolha dividida em 4 áreas de conhecimento: ciências humanas, linguagens e códigos, matemática e ciências da natureza. Possui cinco alternativas de resposta para cada questão (representadas pelas letras A, B, C, D e E) e uma redação.

Os dados fornecidos pelo instituto responsável por aplicar a prova apresentam os valores agrupados por aluno. São disponibilizados seis conjuntos de dados para cada aluno, são eles: dados do participante, dados do local de aplicação da prova, dados da prova objetiva, dados da redação e dados do questionário socioeconômico.

A partir desses conjuntos de dados disponibilizados, referentes ao ano de 2021, é que este trabalho foi desenvolvido.

3. Métodos

Nesta seção, aborda-se os métodos utilizados no trabalho, explicando detalhes do desenvolvimento da aplicação, as ferramentas utilizadas e como foi realizada a preparação da base de dados.

3.1. Ferramentas e tecnologias

Para o desenvolvimento do modelo preditivo de dados referente à descoberta de conhecimento em bases de dados, foi utilizada a API disponibilizada pelo software Waikato Environment for Knowledge Analysis (WEKA), versão 3.8.6. O Weka é um projeto de software livre com o objetivo de fornecer algoritmos de aprendizado de máquinas e diversas ferramentas de processamentos de dados.

O WEKA é aceito largamente tanto na academia e em empresas, com uma comunidade de usuários ativa, contando com mais de um milhão e quatrocentos mil downloads desde abril do ano de 2000. Sua implementação é realizada em linguagem Java, o que também contribui para sua manutenção e modificação (SIMON et al., 2017).

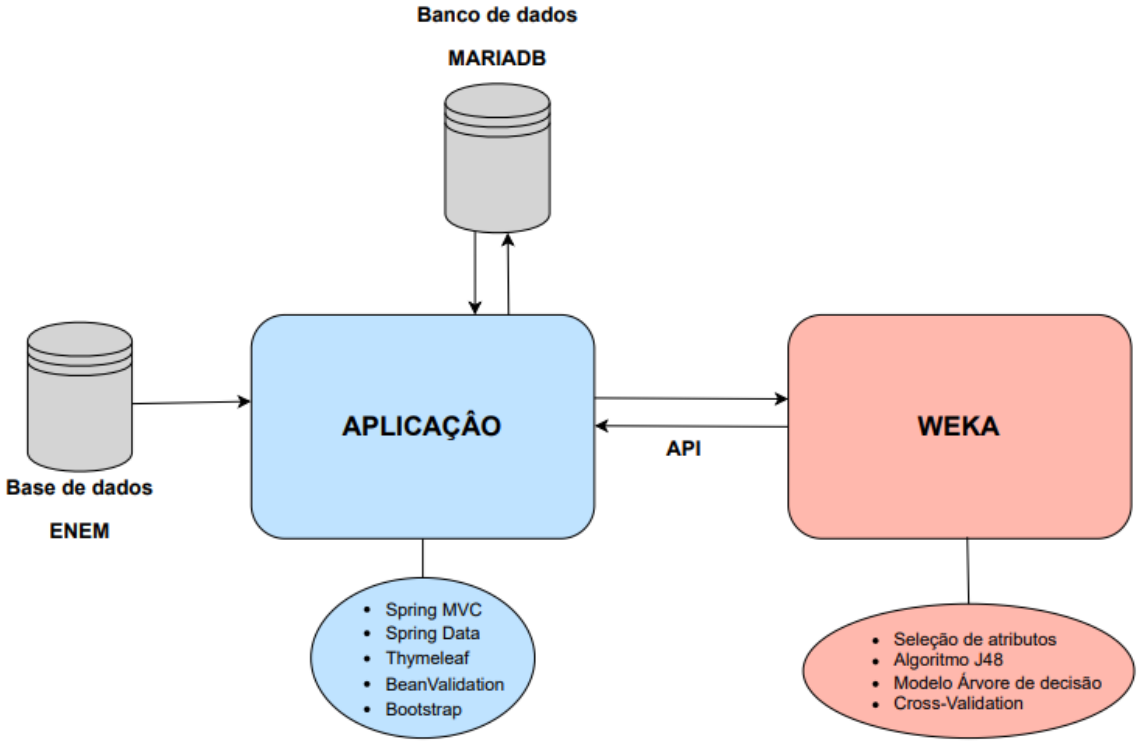


Figura 5. Ilustração da estrutura da aplicação, suas tecnologias e fluxos de dados [autor]

Utilizando-se a tecnologia que o ambiente Java fornece para desenvolvedores web, foi criado um servlet para interação dos dados da aplicação com a API do software Weka, a partir do padrão *Model*, *View* e *Controller* com o *framework* Spring MVC. Para a persistência e armazenamento de dados, são utilizados o framework Spring Data juntamente do banco de dados MariaDB, conforme ilustrado na figura 5.

Para o desenvolvimento de *views* mais otimizadas, foi utilizado o Thymeleaf nos formulários junto com o BeanValidation para validação dos dados. Para a elaboração das interfaces para o usuário, foi utilizado o framework Bootstrap, que dispõe de uma vasta biblioteca de estilos predefinidos na linguagem CSS. Ainda, as ferramentas que auxiliaram no desenvolvimento do trabalho foram o software IntelliJ e a plataforma GitHub para hospedagem de código-fonte com controle de versionamento.

3.2 Modelo Preditivo

Segundo Silva et al. (2014), na etapa de mineração de dados é possível identificar as informações mais importantes da base de dados. A partir dos padrões descobertos, é possível gerar conhecimento para um processo de tomada de decisão. É através de técnicas implementadas por meio de algoritmos que, com a entrada de um conjunto de fatos ocorridos no mundo real, é possível identificar um padrão de comportamento.

Variados estudos são encontrados na literatura acadêmica citando a aplicação de árvores de decisão para o fornecimento de informações em apoio gestores de instituições de ensino na avaliação do desempenho dos seus alunos e a clusterização de dados na identificação de fatores relacionados à satisfação de alunos em instituições de ensino, também em apoio à tomada de decisão de seus gestores [SILVA et al., 2014].

Segundo Agusto et al. (2017), a árvore de decisão é uma das abordagens mais poderosas na mineração de dados. Além disso, estas ferramentas são muito efetivas na mineração de texto, extração de informação, aprendizado de máquina e reconhecimento de padrões. Bhargava et al. (2013) utilizou o algoritmo J48 dentre os algoritmos de árvore de decisão em seu trabalho, além do que, o algoritmo J48 é mencionado como muito popular em aplicações educacionais por Peña-Ayala (2014).

Nesse sentido, para realizar a mineração de dados nesse trabalho, foi utilizado o algoritmo J48 implementado no software Weka. Conforme Agusto et al. (2017), este algoritmo elabora um modelo de árvore de decisão, baseado em um conjunto de dados de treinamento. Um diferencial da árvore de decisão é que trata-se de um algoritmo que aprende um conjunto de regras, capaz de gerar um modelo de dados passível de compreensão e interpretação pelos usuários finais ou pesquisadores envolvidos. Em resumo, o algoritmo J48 é utilizado para aprender funções de classificação que indicam o valor de uma variável dependente através de dados de variáveis independentes.

A partir dos conjuntos de dados do Exame Nacional do Ensino Médio (Enem) do ano de 2021, a criação do modelo preditivo busca compreender quais características socioeconômicas dos candidatos têm maior potencial de levar a uma nota mais alta ou mais baixa no desempenho das provas. Inicialmente, a base de dados selecionada possui 76 atributos referentes a dados do participante, do local da prova, da prova objetiva, da redação e do questionário socioeconômico, com um total de quase 3.400.000 registros armazenados.

Para fins de realização da tarefa de classificação, foi necessário reduzir o tamanho da base de dados de maneira a permitir a execução dos algoritmos do software Weka satisfatoriamente, visto que todos os dados são carregados previamente em memória. Para tanto, selecionou-se aleatoriamente em torno de 5% dos registros existentes para fins de treinamento e teste do algoritmo de classificação. Este valor foi definido após uma série de testes empíricos que demonstraram a viabilidade da construção do modelo de classificação trabalhando com uma base de dados com no máximo 200 mil registros.

Em relação ao desempenho nas provas, foi considerado como medida de qualidade das notas, de forma a classificar os grupos de estudantes com notas mais altas ou mais baixas, a média aritmética das somas notas dos alunos nas provas das 4 áreas de conhecimento e de redação, conforme realizado em outros estudos (ADEODATO, 2016; TRAVITZKI, 2013). Após o conhecimento da nota média dos alunos, criou-se um

atributo na base de dados para armazenar o valor "*acima_média*" quando a soma das notas for igual ou acima da média geral, e quando a soma das notas for abaixo da nota média, o valor "*abaixo_média*".

Na etapa de pré-processamento, para a escolha de quais atributos devem ser utilizados para realizar a predição na base de dados, foi utilizado o algoritmo de seleção de atributos *WrapperSubsetEval*, disponibilizado no software Weka. O algoritmo de seleção consiste em uma técnica capaz de utilizar o algoritmo de aprendizado desejado e avaliar o desempenho desse algoritmo no conjunto de dados, com diferentes subconjuntos de atributos selecionados. Foram executados vários testes com o algoritmo J48 no ambiente Weka, utilizando diferentes subconjuntos de atributos selecionados pelo próprio algoritmo de seleção. Ao fim, o algoritmo resultou em 34 atributos selecionados da base de dados do Enem 2021 para serem utilizados na etapa de mineração de dados.

3.3 Desenvolvimento da aplicação

O desenvolvimento da aplicação pautou-se por um problema de negócio intimamente relacionado ao objetivo do trabalho e por uma série de requisitos levantados a partir disso.

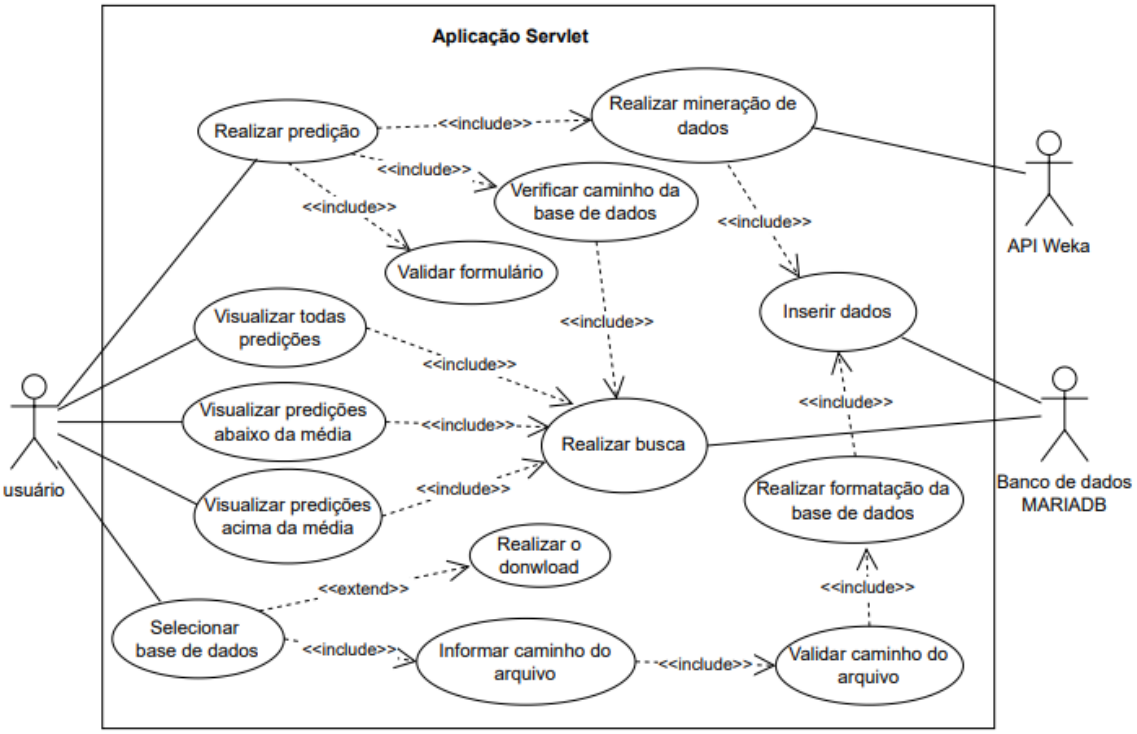


Figura 6. Diagrama de casos de uso do usuário com a aplicação [autor]

O problema de negócio foi assim definido: "Através dos dados socioeconômicos dos candidatos do Enem, necessita-se de uma descoberta de conhecimento em bases de dados (KDD) capaz de classificar a nota do aluno com base em dados reais obtidos no ano de 2021. Devido à complexidade do entendimento de mineração de dados, será desenvolvido um software capaz de simplificar e automatizar o processo para que usuários sem conhecimento sobre o assunto possam utilizar a tecnologia de classificação

da nota.”. A partir disso, alguns requisitos foram levantados para subsidiar a modelagem das interações do usuário com a aplicação, conforme mostrado no diagrama de casos de uso da figura 6.

A partir dos casos de uso dispostos no diagrama da figura 6, foi construída uma interface para o usuário visualizar as predições, realizar predições e atualizar a base de dados.

Conforme ilustrado no diagrama de casos de uso, foram desenvolvidas três seções para a visualização das predições. O conteúdo das seções apresentado para o usuário foi desenvolvido a partir da busca na base de dados MariaDB.

Ao realizar a predição, foi desenvolvida a validação do formulário caso algum campo fique em branco. Após passar pela validação, foi desenvolvida uma consulta na base de dados que retorna o caminho do arquivo da base de dados. Finalizada a validação e encontrado o caminho da base de dados, será realizada a mineração de dados iniciando a criação do modelo da árvore de decisão e a classificação através da API do Weka. Por fim, os dados utilizados na mineração de dados e o resultado da classificação são inseridos no banco de dados, conforme indicado no diagrama de casos de uso.

Conforme a estrutura do diagrama da figura 6, é possível selecionar a base de dados que será consumida pela API do Weka. Na seção “Selecionar base de dados” foi inserido um link que direciona para a página oficial do INEP onde estão as bases de dados mais recentes do Enem. Para selecionar a base de dados, foi desenvolvida uma busca pelo arquivo através do caminho informado e as mesmas formatações realizadas na base de dados do Enem 2021 serão aplicadas em uma nova base de dados a ser selecionada.

4. Apresentação e discussão dos resultados

Nesta seção, são apresentados os resultados obtidos a partir da aplicação das técnicas de mineração de dados através da ferramenta Weka, bem como do desenvolvimento de uma aplicação Java para a descoberta de padrões no perfil de candidatos do Enem do ano de 2021. Estes resultados foram obtidos após a aplicação da técnica de mineração de dados baseada em Árvores de Decisão, através do algoritmo J48, que realiza a tarefa de classificação no software Weka.

4.1 Modelo de classificação

Depois de construído o modelo de classificação, ele pode ser testado por um conjunto de dados de teste com o intuito de classificar registros ainda não vistos, gerando assim métricas de avaliação. Sabe-se que é muito útil medir o desempenho do modelo de dados frente a uma base de testes, pois essa medida gera uma boa estimativa do erro de generalização do modelo, ou seja, como ele se portará ao classificar dados inéditos de todas as classes existentes [HAN et al., 2006].

Para tanto, optou-se pelo uso do método de avaliação da validação cruzada, técnica que consiste em particionar o conjunto de dados em *n* partições. Uma dessas partições é retirada (representa o conjunto de dados de teste), enquanto as demais partições são utilizadas como conjuntos de treinamento do modelo. O processo de validação é repetido até ser realizado *n* vezes, com cada uma das partições sendo utilizada somente uma vez como conjunto de teste. Ao final, os resultados são combinados e é

calculada a média de todos, para se obter um valor único das métricas de desempenho do classificador. Usualmente, e assim foi feito no presente trabalho, opta-se pela partição em 10 subconjuntos, pois essa se mostrou a mais eficiente em testes empíricos [HAN et al., 2006].

Com base nesse método de avaliação, o modelo classificador baseado em árvore de decisão foi capaz de identificar corretamente 69,65% das instâncias, dentre as 155.698 inseridas. Os registros classificados corretamente são apresentados na tabela 1, a qual apresenta os valores de acurácia para cada valor da classe de interesse. A tabela possui um valor máximo 1, indicando 100%. O campo *TP Rate* indica a porcentagem das instâncias classificadas corretamente. O campo *precision* é utilizado em situações nas quais os falsos positivos (FP) são mais prejudiciais do que os falsos negativos. No contexto do trabalho foi utilizado pois, ao classificar se um aluno vai ter nota acima da média, é necessário que esteja correto, mesmo que acabe classificando alguns alunos que podem tirar notas acima da média como abaixo da média.

Tabela 1. Acurácia do modelo de dados detalhada por classe [autor]

	TP Rate	*Precision*	*Class*
	0,765	0,700	*abaixo_média*
	0,617	0,691	*acima_média*
Média	0,697	0,696	

Agora, partindo-se para a interpretação do modelo de classificação gerado, é possível observar os cinco principais atributos iniciados pelo modelo classificador. Os atributos utilizados de primeiro momento no modelo de árvore representam os mais significativos para a definição da classificação das instâncias. São eles, respectivamente:

- **Questão 24 do questionário socioeconômico**: indica se o candidato tem computador na residência;
- **Tipo de língua estrangeira**: indica a opção de língua estrangeira no momento de realização da prova;
- **Questão 6 do questionário socioeconômico**: indica a renda mensal da família do candidato;
- **Questão 7 do questionário socioeconômico**: indica se trabalha empregado(a) doméstico(a) na residência do candidato;
- **Raça**: indica a raça do estudante, conforme a sua percepção.

```
Q024 = A
|   TP_LINGUA = 0
|   |   Q006 = B
|   |   |   Q007 = A
|   |   |   |   TP_COR_RACA = 0
```

Figura 7. Visão de parte do modelo de árvore de decisão, destacando os cinco principais atributos utilizados na classificação [autor]

Além disso, observando a árvore de decisão, é possível ver a relação dos atributos indicados anteriormente, conforme mostra a figura 8. O modelo separa os candidatos que não têm computador em casa, separa o tipo de escola do aluno, caso for particular, e na parte da raça, dentre as 6 opções, o classificador separa raça branca ou não declarada das demais. Ao separar estes valores para cada um destes atributos, o modelo está criando regras de classificação que indicam uma possível classificação, dentre os 2 valores possíveis de classe, para os registros que possuem estes valores e, outra classificação para os registros que têm os demais valores.

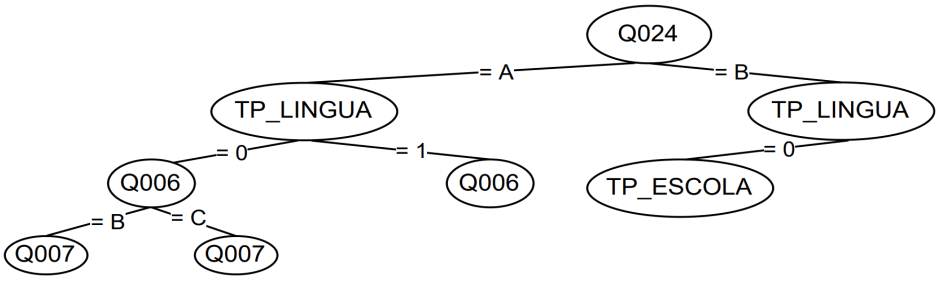


Figura 8. Visão de parte do modelo de classificação no formato de árvore de decisão [autor]

4.2 Aplicação de predição de desempenho

Em relação à aplicação desenvolvida, na tela inicial são apresentadas 3 abas para visualizar as classificações realizadas, conforme a figura 9. A interface possui a aba "Todas", para ver todas as classificações, a aba "Acima da média", na qual são apresentadas as predições que foram acima da média e, por fim, na aba "Abaixo da média", são mostradas as predições abaixo da média. Além disso, as apresentações das notas estão caracterizadas por verde e laranja. As classificações são apresentadas na ordem das mais recentes para as mais antigas.

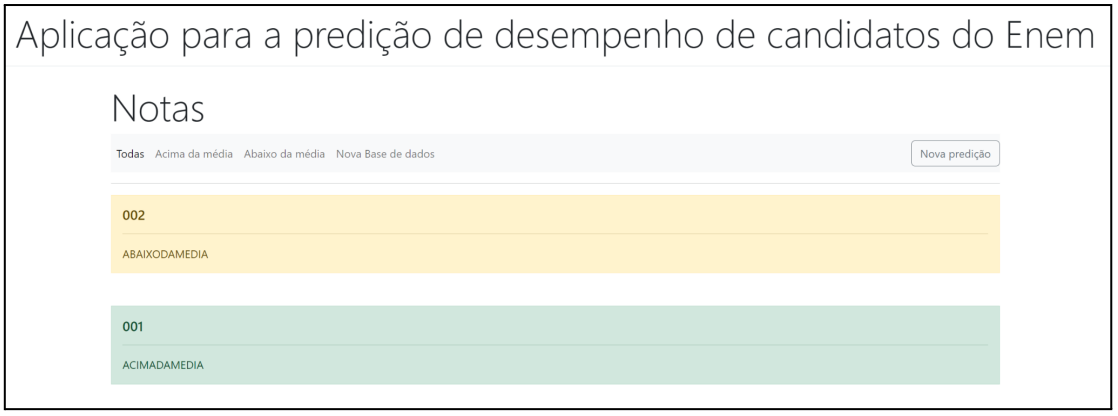


Figura 9. Interface principal da aplicação com as classificações divididas em abas [autor]

Para as classificações realizadas serem mostradas, a aplicação possui conexão com o banco de dados MariaDB em conjunto com o framework *Spring Data*. Ao executar a aplicação, o *framework* cria os campos no banco de dados de acordo com os objetos

criados nas classes da aplicação. Na figura 10 é possível verificar a criação das tabelas no banco de dados com o mesmo nome das classes declaradas na aplicação.

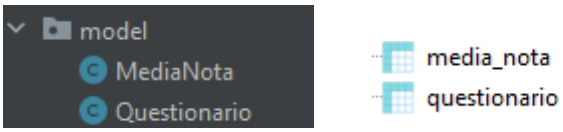


Figura 10. Classes utilizadas em conjunto com o framework Spring Data (A) e as respectivas tabelas criadas no banco de dados (B) [autor]

Assim que as predições são realizadas, o *framework Spring Data* realiza a inserção dos dados da classificação no banco de dados, garantindo a persistência das informações. Após, quando a aba "Todas as notas" é acessada, é realizada uma seleção de dados ao banco, que retorna a identificação e a classificação realizada. Na aba "Acima da média" a seleção de dados objetiva mostrar apenas as notas classificadas como *acimadamedia* no banco de dados. Já na aba "Abaixo da media" possui um select que busca apenas as notas classificadas como *abaixodamedia*. Mas, antes das classificações serem apresentadas para o usuário, com o auxílio do *framework thymeleaf*, foi desenvolvida uma condição na interface HTML para que todas as notas *acimadamedia* fiquem com o fundo verde e todas as notas *abaixodamedia* fiquem com o fundo em laranja, conforme mostra o código-fonte da figura 11.

```
<div class="mb-5" th:each="dado:${questionario}">

    <th:block th:switch="${dado.status.name()}">

        <div th:case="'ABAIXODAMEDIA'" class="card-header alert alert-warning" role="alert">
            <h5 th:text="${dado.identificacao}" class="alert-heading"></h5>
            <hr>
            <p class="mb-0" th:text="${dado.status}"></p>
        </div>
        <div th:case="'ACIMADAMEDIA'" class="card-header alert alert-success">
            <h5 th:text="${dado.identificacao}" class="alert-heading"></h5>
            <hr>
            <p class="mb-0" th:text="${dado.status}"></p>
        </div>
    </th:block>
</div>
```

Figura 11. Trecho de código-fonte que diferencia os resultados da classificação com cores na interface das classificações já realizadas [autor]

Ainda, foi desenvolvido um mecanismo para realizar a classificação e adicionar uma base de dados na qual o *framework Spring Data* realiza a busca do caminho do arquivo necessário no banco de dados. Ao realizar uma classificação, a aplicação verifica se possui o caminho do arquivo na tabela *media_nota* do banco de dados. Caso não tenha, por padrão é realizada uma busca pelo arquivo na pasta local da aplicação. Além disso, antes de iniciar a classificação, implementou-se uma verificação para a aplicação buscar no banco de dados o valor da média das notas dos alunos.

Então, na inclusão da base de dados, os dados do caminho do arquivo que o usuário informou, o caminho do arquivo formatado pela aplicação (*.csv*) e o caminho da base de dados transformada (*.arff*) são salvos no banco de dados. Além disso, a aplicação realiza o cálculo da média e também salva no banco para futuras classificações, conforme mostra a figura 12.

caminhoarff	caminhocsv	caminho_enem	nota_media
D:/TCC/teste100.arff	D:/TCC/microdados/MICRODADOS_ENEM_2021.csv	D:/TCC/microdados/MICRODADOS_ENEM_2021.csv	2.680
D:/TCC/microdados/MICRODADOSENEMFORMATA...	D:/TCC/microdados/MICRODADOSENEMFORMATA...	D:/TCC/microdados/MICRODADOS_ENEM_2021.csv	2.709

Figura 12. Exemplo de como a aplicação armazena os caminhos dos arquivos no banco de dados [autor]

A funcionalidade de adicionar uma nova base de dados foi criada para o usuário ter a possibilidade de atualizar a base de dados para uma mais recente. Assim que o usuário informar o caminho no computador para a base de dados atualizada, a aplicação realizará a formatação da base de dados. Tendo em vista as premissas do desenvolvimento de interfaces, foi inserido um exemplo de como deve ser realizado o processo nos campos responsáveis pela inserção do caminho da nova base de dados. A página da nova base de dados também possui o link para verificar se o INEP disponibilizou uma base de dados mais atual do Enem. Ao adicionar uma nova base de dados, a página redireciona para a página inicial.

Na interface na qual ocorre a classificação, foi criado um formulário socioeconômico com as opções e rótulos preservados conforme o formulário do Enem de 2021, conforme a figura 13.

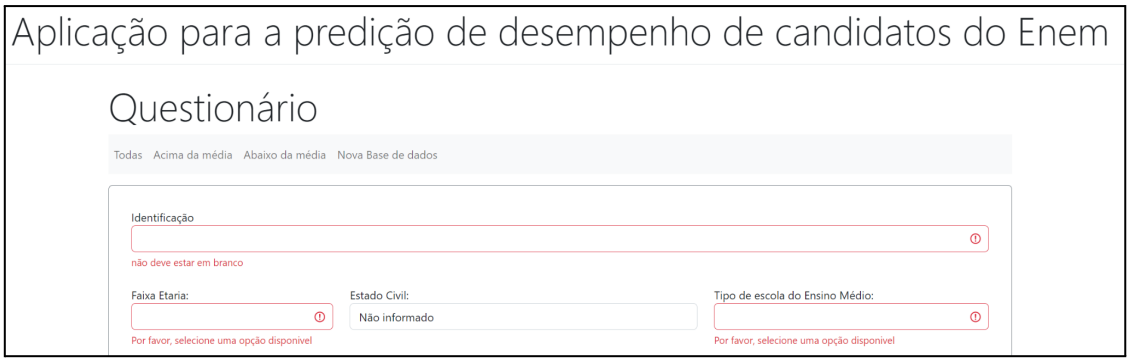


Figura 13. Interface com formulário para realização da predição [autor]

Para o desenvolvimento da interface do questionário foi utilizado o *thymeleaf* em conjunto com o *framework Bean Validation,* para realizar a validação dos campos, visto que todos os dados são necessários para realizar a predição. Além disso, cada campo não preenchido fica contornado com a cor vermelha e possui uma mensagem informando que é preciso preenchê-lo, conforme as premissas do design de interfaces. Para a especificação dos estilos visuais das interfaces da aplicação, inclusive do formulário, foi utilizada a biblioteca *Bootstrap*.

Assim que o formulário é enviado, os dados são salvos no banco de dados e a classificação é realizada e, logo após, ocorre o direcionamento para a página inicial onde estão as classificações mais recentes.

5. Considerações finais

A partir do desenvolvimento da aplicação Java foi possível criar um modelo classificador capaz de facilitar a utilização da mineração de dados para usuários finais que não possuem conhecimento técnico sobre o assunto, visto que a aplicação abstrai a necessidade de conhecimento do processo de mineração de dados.

O modelo foi capaz de predizer o desempenho dos estudantes das provas do Enem baseado no perfil demográfico, utilizando a média da soma das notas como métrica para a classificação, com 70% de acerto na predição. A partir do desenvolvimento e disponibilização da aplicação, entende-se que um usuário leigo poderá utilizar a ferramenta de classificação para realizar estudos em relação ao desempenho dos alunos e suas características socioeconômicas, além de possibilitar a manutenção do histórico ao atualizar a base de dados.

Verificado que o desenvolvimento desta aplicação foi importante para trazer a realidade das questões socioeconômicas dos alunos que participaram do Enem, onde é possível utilizar os dados do perfil socioeconômico do aluno para verificar se o aluno pode vir a ter um bom desempenho quando comparado ao padrão dos candidatos do Enem. Além disso, pode ser possível verificar a nota do Enem do aluno, e servir como uma métrica para avaliar o desempenho da instituição com base no perfil socioeconômico do aluno.

A continuação desse trabalho prevê a melhoria das técnicas de mineração de dados, aperfeiçoamento do algoritmo de classificação, e disponibilização da aplicação online para que qualquer usuário consiga acessar pelo navegador, sem a necessidade de baixar o software. Além de um estudo mais aprofundado dos recursos oferecidos pela ferramenta Weka.

Referências

ADEODATO, P J. L. (2016). Solução de mineração de dados para avaliação do ensino médio brasileiro qualidade com base em dados do enem e censo. Dia 13. Conferência internacional sobre sistemas de informação & gestão de tecnologia

AGUSTO, S. CAZELLA, C. S. (2017). Mineração de Dados Educacionais nos Resultados do ENEM de 2015. Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação.

BANNISTER, Kristian. (2018) “Understanding Sentiment Analysis: What It Is & Why It’s Used.”, https://www.brandwatch.com/blog/understanding-sentiment-analysis/, July.

BHARGAVA, N. et al. (2013) Decision Tree Analysis on J48 Algorithm for Data Mining. International Journal of Advanced Research in Computer Science and Software Engineering, v. 3, n. 6, p. 1114–1119.

CARVALHO, L. A. V. de. Data mining – a mineração de dados no marketing, medicina, economia, engenharia e Administração. São Paulo: Érica, 2001.

DIAS, M. M. Um modelo de formalização do processo de desenvolvimento de sistemas de descoberta de conhecimento em banco de dados. 2001.Tese de Doutorado do Programa de Pós-Graduação em Engenharia de Produção UFSC. Florianópolis, Santa Catarina.

FACELI, K., LORENA, A. C., GAMA, J., & CARVALHO, A. C. P. L. F. (2011). Inteligência Artificial: Uma Abordagem de Aprendizagem de Máquina. LCT.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery: an overview. In: Advances in knowledge discovery and data mining, AAAI Press / The MIT Press, MIT, Cambridge, Massachusets, and London, England, 1996, p.1-34.

GALVÃO, NOEMI and MARIN, HEIMAR de Fátima. (2009) "Técnica de mineração de dados: uma revisão da literatura.", https://www.scielo.br/j/ape/a/Lzj9vW6Fp4QVdXyNKhmtfvv/?lang=pt, July.

GOLDSCHMIDT et al. Data Mining: Conceitos, técnicas, algoritmos, orientações e aplicações. 2° ed. Rio de Janeiro: Elsevier, 2015.

GONÇALVES, L. P. F. and FREITAS H. (2002) "Ferramentas de mineração de dados: Resultado efetivo?", http://www.ufrgs.br/gianti/files/artigos/2002/2002_112_CLADEA.pdf, July.

HAN, J.; KAMBER, M. Data mining – concepts and techniques. United States: Morgan Kaufmann Publishers, 2001.

HAN, J., KAMBER, M., & PEI, J. (2006). Data mining: concepts and techniques. Morgan kaufmann .

MARTINHAGO, S. (2005). Descoberta de Conhecimento sobre o Processo Seletivo da UFPR. Dissertação. Programa de Pós-Graduação em Métodos Numéricos em Engenharia. Universidade Federal do Paraná.

PEÑA-AYALA, A. (2014) Educational data mining: A survey and a data mining-based analysis of recent works. Expert Systems with Applications, v. 41, n. 4, p. 1432–1462.762Anais dos Workshops do VI Congresso Brasileiro de Informática na Educação (WCBIE 2017). VI Congresso Brasileiro de Informática na Educação (CBIE 2017).

REZENDE, S. O. Sistemas inteligentes: fundamentos e aplicações. 1ª ed. Barueri: Editora Manole, 2003.

SILVA, L. A.; SILVA, L. (2014). Fundamentos de Mineração de Dados Educacionais. Anais dos Workshops do Congresso Brasileiro de Informática na Educação, v. 3, n. 1, p. 568 -581.

SILVA, M. de S. (2014). O Pré-Processamento em Mineração de Dados como método de suporte à modelagem algorítmica.

TAN, P.-N., STEINBACH, M., & KUMAR, V. (2009). Introdução ao data mining: mineração de dados. Rio de Janeiro: Ciência Moderna.

TRAVITZKI, R. "ENEM: limites e possibilidades do Exame Nacional do Ensino Médio enquanto indicador de qualidade escolar". Tese, USP, São Paulo, 2013 (em português).

WEKA, https://www.scielo.br/j/ape/a/Lzj9vW6Fp4QVdXyNKhmtfvv/?lang=pt, August.