

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

MARCOS LAYDNER

**AUTOMAÇÃO DA AVALIAÇÃO DE APRENDIZAGEM DE MACHINE  
LEARNING PARA CLASSIFICAÇÃO DE IMAGENS  
NO ENSINO FUNDAMENTAL**

FLORIANÓPOLIS

2022

MARCOS SILVA LAYDNER

**AUTOMAÇÃO DA AVALIAÇÃO DE APRENDIZAGEM DE MACHINE  
LEARNING PARA CLASSIFICAÇÃO DE IMAGENS  
NO ENSINO FUNDAMENTAL**

Trabalho de Conclusão do Curso de Graduação em Sistemas de Informação, do Departamento de Informática e Estatística, do Centro Tecnológico da Universidade Federal de Santa Catarina, requisito parcial à obtenção do título de Bacharel em Sistemas de Informação.

Orientadora: Prof.<sup>a</sup> Dr.<sup>a</sup> rer. nat. Christiane Gresse von Wangenheim, PMP

FLORIANÓPOLIS

2022

## RESUMO

Com a crescente evolução e popularização do uso de tecnologias de *Machine Learning* no dia a dia, é imperativo que a população geral tenha um entendimento básico sobre como essas tecnologias funcionam, de forma que sua utilização seja melhor aproveitada. Para isso, é importante que o ensino desses conceitos seja feito cedo já na Educação Básica. Com isso, algumas iniciativas têm surgido, porém a maioria não possui um modelo de avaliação dessa aprendizagem. Assim, o presente trabalho visa responder a pergunta de pesquisa: “Como automatizar a avaliação da aprendizagem de ML voltada a classificação de imagens nos anos finais do ensino fundamental?”. Dessa forma, tem-se como objetivo desenvolver e testar um modelo de análise da aprendizagem de ML na Educação Básica, adotando técnicas de *Deep Learning* para automatizar o processo. Isso é alcançado por meio da fundamentação teórica dos principais conceitos, uma revisão do estado da arte, o desenvolvimento do modelo de avaliação, e a sua automatização, e sua integração e avaliação no contexto de um curso da iniciativa Computação na Escola. Dessa forma, criou-se uma extensão da ferramenta CodeMaster que permite automaticamente avaliar a aprendizagem de *Machine Learning* do aluno, facilitando o processo, e aumentando a chance de adoção do ensino de computação nas escolas brasileiras.

**Palavras chave:** Educação Básica, Avaliação, Inteligência Artificial, *Machine Learning*.

## **LISTA DE FIGURAS**

Figura 1 - Curso Machine Learning para Todos	<b>19</b>
Figura 2 - Processo de Machine Learning	<b>24</b>
Figura 3 - Frequência dos conceitos abordados	<b>31</b>
Figura 4 - Frequência dos 3 níveis de Bloom (1956)	<b>31</b>
Figura 5 - Frequência dos estágios do ciclo use-modify-create	<b>32</b>
Figura 6 - Frequência de níveis de automação da avaliação	<b>33</b>
Figura 7 - Quiz da Aula 5	<b>43</b>
Figura 8 - Diagrama de casos de Uso do CodeMaster, adaptado de Demetrio (2017)	<b>66</b>
Figura 9 - Diagrama de Componentes do CodeMaster, baseado em Demetrio (2017)	<b>68</b>
Figura 10 - Sequência da análise da rubrica Machine Learning para Todos	<b>69</b>
Figura 11 - Diagrama de classes relacionadas à análise dos dados do Machine Learning para Todos	<b>70</b>
Figura 12 - Diagrama Entidade Relacionamento do Banco de Dados, baseado em (DEMETRIO, 2017)	<b>71</b>

## **LISTA DE TABELAS**

Tabela 1 - Rascunho dos objetivos de aprendizado da ideia #3. Traduzido de (AI4K12, 2020)	<b>14</b>
Tabela 2 - Competências necessárias para AI Literacy. Traduzido de (LONG; MAGERKO, 2020)	<b>15</b>
Tabela 3 - Objetivos de aprendizagem do curso Machine Learning Para Todos!. Traduzido de (GRESSE VON WANGENHEIM et al., 2020)	<b>17</b>
Tabela 4 - Plano de ensino do curso Machine Learning Para Todos! Traduzido de Gresse von Wangenheim et al. ( 2020)	<b>18</b>
Tabela 5. Níveis de conhecimento de Bloom (BLOOM, 1956), relacionados com avaliações disponíveis no H5P. Adaptado de (CLICK4IT, 2013)	<b>21</b>
Tabela 6. Classificação tipos de feedback por complexidade (GONÇALVES, 2017)	<b>21</b>
Tabela 7 - Termos da busca	<b>28</b>
Tabela 8 - Frases de busca para cada fonte	<b>28</b>
Tabela 9 - Resultados obtidos	<b>29</b>
Tabela 10. Que abordagens de avaliação existem e quais suas características?	<b>30</b>
Tabela 11 - Que conceitos relacionados à ML são avaliados?	<b>32</b>
Tabela 12 - Como esses conceitos de ML são analisados	<b>33</b>
Tabela 13 - Presença de nota ou feedback	<b>34</b>
Tabela 14 - Quizzes da Aula 1	<b>40</b>
Tabela 15 - Quizzes da Aula 2	<b>41</b>
Tabela 16 - Quizzes da Aula 3	<b>41</b>
Tabela 17 - Quizzes da Aula 4	<b>42</b>
Tabela 18 - Quizzes da Aula 6	<b>42</b>
Tabela 19 - Rubrica de avaliação de aprendizagem com base no desempenho (GRESSE VON WANGENHEIM et al., 2021)	<b>44</b>
Tabela 20 - Cálculo da nota final	<b>45</b>
Tabela 21 - Resumo do Modelo de Machine Learning	<b>48</b>
Tabela 22 - Cenário de resultados de aprendizagem de baixo desempenho	<b>55</b>
Tabela 23 - Cenário de resultados de aprendizagem de médio desempenho	<b>57</b>
Tabela 24 - Cenário de resultados de aprendizagem de alto desempenho	<b>59</b>
Tabela 25 - Mapeamento da nota e a cor da faixa do ninja robô	<b>62</b>
Tabela 26 - Requisitos Funcionais	<b>65</b>
Tabela 27 - Requisitos Não-Funcionais	<b>66</b>
Tabela 28 - Interfaces gráficas do USC07	<b>71</b>

## SUMÁRIO

<b>1. INTRODUÇÃO</b>	<b>6</b>
1.1 OBJETIVOS	8
1.4 ESTRUTURA DO DOCUMENTO	11
<b>2. FUNDAMENTAÇÃO TEÓRICA</b>	<b>12</b>
2.1 ENSINO DE ML NA EDUCAÇÃO BÁSICA	12
2.2 AVALIAÇÃO DA APRENDIZAGEM	18
2.3 DEEP LEARNING	22
<b>3. ESTADO DA ARTE</b>	<b>26</b>
3.1 DEFINIÇÃO DO PROTOCOLO DE REVISÃO	26
3.2 EXECUÇÃO DA BUSCA	28
3.3 ANÁLISE DOS RESULTADOS	28
3.4 DISCUSSÃO	34
<b>4. DESENVOLVIMENTO DO MODELO DE AVALIAÇÃO</b>	<b>36</b>
4.1 ANÁLISE DO CONTEXTO	36
4.2 AVALIAÇÃO DO CURSO MACHINE LEARNING PARA TODOS!	37
4.3 AVALIAÇÃO DE DESEMPENHO	42
4.4 NOTA FINAL	44
<b>5. AUTOMAÇÃO DO MODELO DE AVALIAÇÃO</b>	<b>45</b>
<b>6. INTEGRAÇÃO COM O CODEMASTER</b>	<b>71</b>
<b>7. CONCLUSÃO</b>	<b>78</b>
<b>REFERÊNCIAS</b>	<b>79</b>

# 1. INTRODUÇÃO

Desde de Turing até os dias atuais, a influência da computação no dia a dia das pessoas vem crescendo consideravelmente. Diversos aspectos da vida atual, e quase todos setores da economia têm sido fortemente afetados, ou são agora movimentados, por sistemas computacionais. Com toda essa influência, a tendência é que o conhecimento sobre essas tecnologias seja cada vez mais importante para todos, e não só estudantes de computação, estejam preparados para as carreiras do século XXI (CSTA, 2017).

Uma das áreas da computação com grande influência na sociedade atual é a de Inteligência Artificial (IA) e Aprendizado de máquina (*Machine Learning - ML*). ML é um campo da IA que estuda tecnologias que permitem que máquinas aprendam a partir de dados passados, sem terem de ser programadas especificamente (JAKHAR; KAUR, 2020). *Machine Learning* tem sido utilizado para identificação de *spam*, sistemas de busca, mecanismos de recomendação, entre outros usos que fazem parte da vida diária de muitas pessoas. Uma das tarefas típicas de ML é o reconhecimento de imagens, o foco deste trabalho, usado por exemplo, no reconhecimento facial para desbloqueio de celulares. Porém, a falta de conhecimento da tecnologia por trás de toda essa facilidade, utilizada como uma caixa preta, pode trazer insegurança e estranheza, reduzindo seu impacto positivo e seu potencial de desenvolvimento (EVANGELISTA et al., 2018).

Além disso, com o crescimento desse campo, muitos empregos podem ser afetados ou até extintos, enquanto outros surgirão, relacionados com os novos avanços, e que requerem um conhecimento mais profundo do assunto. Além do impacto da tecnologia em diversas carreiras, a própria área da computação e IA estão entre as que mais crescem no mundo, inclusive no Brasil, onde mesmo em Outubro de 2019, durante a pandemia, a quantidade de vagas abertas da área era de aproximadamente 25 mil (BNE, 2020). Além disso, “Especialista em Inteligência Artificial” ficou entre as 15 profissões emergentes do Brasil do LinkedIn, crescendo 73% em relação ao ano anterior (LINKEDIN, 2020), isto dentro do setor de TI, em que a demanda por profissionais de TI aumenta de forma rápida, mais do que dobrando anualmente (LINKEDIN, 2020). O setor emprega 1,56 milhões de pessoas (BRASSCOM, 2020), um número notável quando leva-se em consideração a falta de uma estratégia de formação de profissionais de TI e a alta evasão em cursos da área de Computação (CARVALHO et al., 2019), o que acaba gerando um déficit de profissionais da área, estimado a ser de 408 mil em 2022 (SOFTEX, 2019). Observa-se também que pela inserção da TI no dia a dia de qualquer cidadão, atuando

em qualquer área de conhecimento, é necessário que todos tenham um conhecimento básico de TI.

Porém, atualmente, o ensino de computação nas escolas brasileiras é inexistente ou insuficiente para as necessidades do mundo moderno. A maioria das instituições brasileiras que apresentam o ensino de computação, focam somente em como utilizar as tecnologias (*IT literacy*), enquanto o mundo precisa que os alunos tenham um conhecimento mais profundo, e uma fluência (*IT fluency*) com a TI, para que adquiram a capacidade de aprender e utilizar eventuais novas tecnologias do futuro, ao longo de sua vida. Por isso, a inclusão do ensino de computação nas escolas se tornou uma tendência mundial (PINTO et al., 2020). Mais recentemente, isto também inclui o ensino de IA/ML como uma das áreas de conhecimento da computação.

Ao redor do mundo observa-se uma tendência recente de abordar também IA/ML na Educação Básica. O ensino de IA/ML vem sendo incluído em vários países como a China, em que Inteligência Artificial é um conteúdo obrigatório para alunos do ensino médio (JING, 2018). Para este fim estão sendo criadas diretrizes para o ensino, como *K-12 Guidelines for Artificial Intelligence* (TOURETZKY et al., 2019), que indicam os conteúdos importantes que se encaixam no contexto de Inteligência Artificial na educação básica, que facilitou e impulsionou o crescimento de cursos no mundo. Nos últimos anos também surgiram diversos cursos de IA/ML para Educação Básica, que utilizam diferentes técnicas de ensino, desde ambientes de programação baseados em blocos, até o uso simples de python e frameworks de *Machine Learning* (MARQUES et al., 2020).

Dentro deste contexto, com o objetivo de introduzir a aprendizagem de conceitos de ML no contexto brasileiro voltado aos anos finais do ensino fundamental, a iniciativa Computação na Escola também iniciou o desenvolvimento de um curso online. Seguindo as diretrizes *K-12 Guidelines for Artificial Intelligence* (TOURETZKY et al., 2019), o curso *Machine Learning para Todos!* (GRESSE VON WANGENHEIM et al., 2020) proporciona uma percepção do uso de *Machine Learning* no dia a dia, compreensão de conceitos básicos de *Machine Learning* e redes neurais, e leva o aluno a aprender a aplicação destes conceitos por meio da criação de um modelo de reconhecimento de imagens predefinido. Esse processo guia o aluno pelas etapas de um projeto de ML, desde a fase de preparação de dados, até a implantação.

Para guiar a progressão da aprendizagem do aluno é importante fornecer um *feedback*, avaliando os seus resultados de aprendizagem. Levando em consideração que muitos cursos guiam o aluno ao nível de criar artefatos de ML, esta avaliação deve ser feita por meio de avaliação de desempenho (ALVES, 2019). Portanto, é necessário o



desenvolvimento de um modelo que permita de forma clara e eficiente, apresentar um *feedback* ao aluno, e ao mesmo tempo realizar uma avaliação objetiva da aprendizagem. Porém, observa-se que a maioria dos cursos atuais aborda a questão da avaliação de aprendizagem voltada somente aos níveis de lembrança e compreensão (BLOOM, 1956), e nota-se que ainda faltam modelos de avaliação da aprendizagem dos alunos com base nos artefatos de ML criados por eles. Esse é um problema não só em relação ao curso Machine Learning para Todos!, mas que é observado também em outros cursos de *Machine Learning* com esse foco de modo geral (MARQUES et al., 2020 - SALVADOR et al., 2021). Além da necessidade de um modelo de avaliação, é importante que ele seja automatizado (FERREIRA, 2020). A automatização permite reduzir o esforço e o conhecimento necessários por parte do professor, levando em conta que o mesmo não necessariamente tem formação específica em ML. Além disso, uma avaliação automática aumentaria a precisão e velocidade da avaliação.

Portanto, notando a atual lacuna de modelos educacionais que avaliam sistematicamente o aprendizado dos conceitos de Inteligência Artificial e Aprendizado de Máquina, este projeto busca responder a seguinte pergunta de pesquisa: Como automatizar a avaliação da aprendizagem de ML voltado a classificação de imagens nos anos finais do ensino fundamental?

## 1.1 OBJETIVOS

### **Objetivo geral**

O objetivo geral deste trabalho é desenvolver e testar um modelo de avaliação da aprendizagem de ML na educação básica. São adotadas técnicas de *deep learning* para automatizar a análise da aprendizagem. O modelo desenvolvido é integrado ao sistema CodeMaster (DEMETRIO, 2017), adicionando a avaliação automatizada da aprendizagem de ML voltado a classificação de imagens no contexto da educação básica (anos finais do Ensino Fundamental). O modelo de aprendizagem é voltado ao curso Machine Learning para Todos (GRESSE VON WANGENHEIM et al., 2020) e baseado nas diretrizes de currículo de ensino de IA na Educação Básica (TOURETZKY et al., 2019), desenvolvido pela Associação para o Avanço da Inteligência Artificial (AAAI) e da Associação dos Professores de Ciência da Computação (CSTA).

## **Objetivos Específicos**

O1. Analisar a fundamentação teórica sobre ensino de ML na educação básica, avaliação da aprendizagem e *deep learning*.

O2. Analisar o estado da arte em relação a análise automática da aprendizagem de ML na educação básica.

O3. Definir o modelo de avaliação .

O4. Desenvolver e testar um sistema para automatizar a avaliação utilizando *deep learning*, integrado ao sistema CodeMaster (DEMETRIO, 2017).

## **Premissas e restrições**

O trabalho é realizado de acordo com o regulamento vigente do Departamento de Informática e Estatística (INE – UFSC) em relação aos Trabalhos de Conclusão de Curso. O modelo de avaliação é instanciado especificamente no curso de Machine Learning para Todos! voltado ao desenvolvimento de um modelo de classificação de imagens de recicláveis.

### **1.2 METODOLOGIA DE PESQUISA**

A metodologia de pesquisa utilizada neste trabalho é dividida nas seguintes etapas.

#### **Etapa 1 – Fundamentação teórica**

Estudando, analisando e sintetizando os conceitos principais e a teoria referente aos temas a serem abordados neste trabalho é apresentado a fundamentação teórica utilizando a metodologia de revisão narrativa (CORDEIRO et al., 2007). Nesta etapa são realizadas as seguintes atividades:

A1.1 – Análise teórica sobre ensino de ML na educação básica

A1.2 - Avaliação da aprendizagem

A1.3 – Análise teórica sobre *deep learning*

#### **Etapa 2 – Estado da arte**

Nesta etapa é realizado um mapeamento sistemático da literatura seguindo o processo proposto por Petersen et al. (2008) para identificar e analisar modelos de

análise automatizado da estética visual de interfaces de usuário de apps atualmente sendo utilizados. Esta etapa é dividida nas seguintes atividades:

- A2.1 – Definição do protocolo da revisão;
- A2.2 – Execução da busca e seleção de artigos relevantes;
- A2.3 – Extração e análise de informações relevantes.

### **Etapa 3 – Desenvolvimento do modelo de avaliação**

Para o desenvolvimento do modelo de avaliação do analisador automático de design de interface será utilizada a metodologia de design instrucional ADDIE (BRANCH, 2009).

- A3.1 - Analisar o contexto.
- A3.2 - Desenvolver os instrumentos de avaliação para o projeto e medidas de avaliação.
- A3.3 - Definir o feedback instrucional.

### **Etapa 4 – Desenvolvimento da automação da avaliação**

Nesta etapa é desenvolvida a automação da avaliação de forma iterativa automatizando a avaliação para cada um dos instrumentos e cada um dos critérios da rubrica.

- A4.1 - Iteração da automação da avaliação dos quizzes
- A4.2 - Iteração da automação da rubrica, usando em alguns casos deep learning

### **Etapa 5 – Integração**

Nesta etapa é integrada essa funcionalidade ao sistema CodeMaster (DEMETRIO, 2017) seguindo um processo de engenharia de software proposto por Pressman (2016). Esta etapa é dividida nas seguintes atividades:

- A5.1 – Análise de requisitos;
- A5.2 – Modelagem da arquitetura do sistema;
- A5.3 – Modelagem detalhada e implementação;
- A5.4 – Testes do sistema.

## **1.4 ESTRUTURA DO DOCUMENTO**

Este trabalho é disposto da seguinte forma: No capítulo 2 são descritos os fundamentos teóricos para facilitar a compreensão dos principais conceitos utilizados nesta pesquisa. É comentado sobre o ensino de ML na educação básica, avaliação da aprendizagem e deep learning. No terceiro capítulo, é apresentada a revisão sistemática do estado da arte, levantando as unidades instrucionais que possuem avaliação. No capítulo 4, é explicado como foi feito o desenvolvimento do modelo de avaliação, com

base na análise do contexto, nos objetivos de aprendizagem e estrutura de aulas do curso Machine Learning para todos. No quinto são apresentadas as soluções de automação da avaliação da rubrica, e o cálculo da nota final. No capítulo 6 é apresentada a conclusão do trabalho.

## 2. FUNDAMENTAÇÃO TEÓRICA

Considerando-se o impacto de IA e ML na sociedade moderna, e sua presença no dia a dia das pessoas, o acesso ao aprendizado sobre o campo, além do ensino superior, ainda é limitado. Com exceção de países como a China, em que Inteligência Artificial é um conteúdo obrigatório para alunos do ensino médio (JING, 2018)

### 2.1 ENSINO DE ML NA EDUCAÇÃO BÁSICA

Considerando-se o impacto de IA e ML na sociedade moderna, e sua presença no dia a dia das pessoas, o acesso ao aprendizado sobre o campo, além do ensino superior, ainda é limitado. Com exceção de países como a China, em que Inteligência Artificial é um conteúdo obrigatório para alunos do ensino médio (JING, 2018), o assunto tipicamente está limitado ao ensino superior, não sendo devidamente contemplado mesmo em diretrizes de aprendizado de computação para a educação básica. No Brasil, o MEC prevê somente para o ensino médio, o aprendizado de conceitos de computação, incluindo o desenvolvimento de pensamento computacional, o qual engloba a resolução de problemas por meio de algoritmos, porém Inteligência Artificial é apenas brevemente mencionada no Itinerário Formativo de Matemática e suas tecnologias (MEC, 2018).

“Os itinerários formativos – estratégicos para a flexibilização da organização curricular do Ensino Médio, pois possibilitam opções de escolha aos estudantes – podem ser estruturados com foco em uma área do conhecimento, na formação técnica e profissional ou, também, na mobilização de competências e habilidades de diferentes áreas, compondo itinerários integrados, nos seguintes termos das DCNEM/2018:

...  
II – matemática e suas tecnologias: aprofundamento de conhecimentos estruturantes para aplicação de diferentes conceitos matemáticos em contextos sociais e de trabalho, estruturando arranjos curriculares que permitam estudos em resolução de problemas e análises complexas, funcionais e não-lineares, análise de dados estatísticos e probabilidade, geometria e topologia, robótica, automação, **inteligência artificial**, programação, jogos digitais, sistemas dinâmicos, dentre outros, considerando o contexto local e as possibilidades de oferta pelos sistemas de ensino; ...” (MEC, 2018)

Porém, o assunto vem ganhando atenção nos últimos anos, e diferentes iniciativas surgiram com o objetivo de determinar diretrizes para um currículo, como *K-12 Guidelines for Artificial Intelligence* (TOURETZKY et al., 2019), ou *CSTA K–12 Computer Science Standards* (CSTA, 2017). Essas diretrizes citam 5 “grandes ideias” que todos estudantes devem saber em relação a IA (TOURETZKY et al., 2019):

- **Percepção** - é importante que o aluno entenda que a percepção de computadores de linguagem oral ou de imagens requer um grande conhecimento do respectivo domínio ;
- **Representação e Raciocínio** - o estudante deve entender o conceito de representação e entender que computadores constroem representações usando

dados, e que delas pode-se derivar novas informações por meio da aplicação de algoritmos de raciocínio;

- **Aprendizado** - o aluno precisa compreender que o aprendizado de máquina é um tipo de inferência estatística que acha padrões nos dados;
- **Interação Natural** - é importante que o estudante saiba que por mais que computadores consigam entender linguagem natural de forma limitada, eles atualmente possuem menos habilidades conversacionais e raciocínio geral de uma criança;
- **Impacto Social** - é imperativo que o aluno seja capaz de identificar maneiras que IA contribui positivamente em suas vidas, além de entender que para a construção ética de um sistema de Inteligência Artificial é preciso atenção nos quesitos de transparência e equidade.

Assim, a ideia #3 se refere a aprendizagem - *Machine Learning*. Especificamente em relação a esta ideia, conforme Touretzky et al. (2019), espera-se que o aluno aprenda:

- O que é aprender?
- Abordagens para ML (algoritmos de regressão, redes neurais artificiais, etc.)
- Tipos de algoritmos de aprendizagem por estilos de aprendizagem
- Fundamentos de redes neurais
- Tipos de arquiteturas de redes neurais
- Como dados de treinamento influenciam a aprendizagem
- Limitações de *Machine Learning*

São especificados também, os objetivos de aprendizados relacionados à ideia #3 (AI4K12, 2020) na Tabela 1.

Tabela 1 - Rascunho dos objetivos de aprendizado da ideia #3. Traduzido de (AI4K12, 2020)

Conceito	3-5 ano escolar	6-8 ano escolar
Humanos vs. Máquinas	Diferenciar entre como pessoas e computadores aprendem	Contrastar as características únicas do aprendizado humano e das maneiras que máquinas operam
Achando padrões em dados	Modelar como aprendizagem supervisionada identifica padrões em dados rotulados	Modelar como aprendizagem supervisionada identifica padrões em dados rotulados
Treinando um modelo	Treinar um modelo de classificação usando <i>machine learning</i> , e então examinar a precisão do mesmo com entradas novas	Treinar e avaliar um modelo de classificação ou previsão usando <i>machine learning</i> em um dataset tabular
Construir vs. usar um raciocinador	Demonstrar como dados de treino são rotulados quando usa-se uma ferramenta de ML	Explicar a diferença entre treinar e usar um modelo de raciocínio
Ajustar representações internas	Analisar um jogo onde se constrói a árvore de decisão, descrevendo a organização da árvore e do algoritmo usado para adicionar nós	Comparar como um algoritmo de aprendizado de árvore de decisão funciona vs. como um algoritmo de aprendizado de redes neurais funciona

Aprendendo por experiência	Explicar como aprendizado de reforço permite que um computador aprenda por experiência (tentativa)	Explicar a diferença entre aprendizado supervisionado e não supervisionado
Estrutura de uma rede neural	Ilustrar como uma rede neural de 1 a 3 neurônios é uma função que computa uma saída	Ilustrar a estrutura de uma rede neural e descrever como suas partes formam um conjunto de funções que computam uma saída
Ajuste de Peso	Demonstrar como pesos são designados em uma rede neural para produzir o comportamento desejado de entrada e saída	Demonstrar como uma regra de aprendizado pode ser usada para ajustar os pesos em uma rede neural de um nível
Conjuntos de <i>feature</i>	Criar um <i>conjunto de dados</i> rotulado com <i>features</i> explícitas de diferentes tipos e usa uma ferramenta de machine learning para treinar um classificador nestes dados	Criar um <i>dataset</i> para treinar um classificador de árvore de decisão ou preditor e explorar o impacto que diferentes <i>features</i> têm na árvore de decisão
<i>Datasets</i> Grandes	Ilustrar como treinar um classificador para um conceito amplo como “cachorro” requer uma grande quantidade de dados para capturar a diversidade do domínio	Ilustrar como objetos em uma imagem podem ser segmentados e rotulados para construir um conjunto de treino para reconhecimento de objetos
Viés	Examinar características e rótulos de dados de treino para detectar possíveis fontes de viés	Explicar como a escolha dos dados de treino molda o comportamento do classificador, e como esse viés pode ser introduzido se o conjunto de treino não for balanceado apropriadamente

Ainda nessa linha de pensamento, Long e Magerko (2020) desenvolveram uma definição concreta para *AI Literacy* (Alfabetização de IA), sugerindo competências chave para *AI Literacy*, conceitos que o aluno deve entender, divididas em 5 temas diferentes: O que é IA?; O que IA pode fazer?; Como IA funciona?; Como IA deve ser usada?; e Como as pessoas enxergam IA? (LONG; MAGERKO, 2020).

Tabela 2 - Competências necessárias para *AI Literacy*. Traduzido de (LONG; MAGERKO, 2020)

Competência	Descrição
Competência 1 (Reconhecendo IA)	Distinguir entre artefatos tecnológicos que usam ou não IA
Competência 2 (Entendendo Inteligência)	Analisar e discutir criticamente características que fazem uma entidade “inteligente”, incluindo discutir diferenças entre humanos, animais e inteligência de máquina
Competência 3 (Interdisciplinaridade)	Reconhecer que existem muitos jeitos diferentes de pensar e desenvolver máquinas “inteligentes”. Identificar uma variedade de tecnologias que usam IA, incluindo sistemas cognitivos, robótica e ML
Competência 4 (Generalista vs. Específico)	Diferenciar entre IA generalista e IA específica
Competência 5 (Pontos Fortes e Fracos de IA)	Identificar tipos de problemas que IA se sobressai e problemas que são mais difíceis para a mesma. Usar essa informação para determinar quando é apropriado usar IA e quando usar habilidades humanas.
Competência 6 (Imaginar IA do Futuro)	Imaginar possíveis aplicações futuras de IA e considerar os efeitos de tal aplicação no mundo
Competência 7 (Representação)	Entender o que representação de conhecimento é e descrever alguns exemplos de representações de conhecimento
Competência 8 (Tomada de Decisão)	Reconhecer e descrever exemplos de como computadores raciocinam e fazem decisões
Competência 9 (Passos de ML)	Entender os passos envolvidos em <i>machine learning</i> e as práticas e desafios de cada passo.

Competência 10 (Papel Humano na IA)	Reconhecer que humanos fazem um papel importante na programação, escolha de modelos e no refinamento de sistemas de IA
Competência 11 (Data Literacy)	Entender conceitos básicos de <i>data literacy</i>
Competência 12 (Aprender pelos Dados)	Reconhecer que computadores geralmente aprendem a partir dos dados
Competência 13 (Interpretar Criticamente os Dados)	Entender que dados não podem ser tomados por si só, e requerem interpretação. Descreva como os exemplos de treino dados em um dataset inicial podem afetar o resultado de um algoritmo.
Competência 14 (Ação e Reação)	Entender que alguns sistemas de IA têm a habilidade de fisicamente agir no mundo. Essa ação pode ser direcionada por raciocínio de alto nível ( como andar em um caminho planejado) ou pode ser reativo (pular para trás para evitar um obstáculo percebido)
Competência 15 (Sensores)	Entender o que sensores são, reconhecer que sensores percebem o mundo usando sensores, e identificar sensores em uma variedade de dispositivos. Reconhecer que diferentes sensores suportam diferentes tipos de representação e raciocínio sobre o mundo.
Competência 16 (Ética)	Identificar e descrever diferentes perspectivas no problemas éticos chave acerca de IA (privacidade, singularidade, tomadas de decisões éticas, etc.)
Competência 17 (Programabilidade)	Compreender que agentes são programáveis.

Já no contexto brasileiro, a Sociedade Brasileira de Computação (SBC), seguindo as tendências mundiais, desenvolveu suas próprias diretrizes para ensino de computação na educação básica (SBC, 2019). Inteligência artificial é um dos objetivos de conhecimento, cujas habilidades desenvolvidas são “Compreender os fundamentos da inteligência artificial e da robótica” (SBC, 2019). Mesmo assim, também é esperado o tema de IA somente para o ensino médio, sem menção sobre ML:

“No eixo de Pensamento Computacional, são trabalhadas a técnica de transformação de problemas e o paradigma de metaprogramação (algoritmos que recebem outros algoritmos como entrada), que são conceitos necessários para a compreensão dos limites da computação, ou seja, dos limites da formalização/racionalização. Este entendimento, aliado aos fundamentos de **inteligência artificial** e robótica, provê a base necessária para uma discussão mais substanciada sobre o que é o Homem e o que é a Máquina, quais as similaridades e diferenças, não somente do ponto de vista físico, mas do ponto de vista filosófico, entendendo também as grandes questões éticas envolvidas na **inteligência artificial**.” (SBC, 2019)

Nos últimos anos, várias unidades instrucionais vêm sendo criadas, como cursos, tutoriais e cursos online (MARQUES et al., 2020). As Unidades Instrucionais (UIs) estudadas variam de oficinas de uma hora, até cursos de um semestre, com competências diferentes que cobrem desde o que é ML até técnicas específicas de *Machine Learning*, e o impacto dela na sociedade (MARQUES et al., 2020). As unidades alcançam níveis diferentes na Taxonomia de Bloom (BLOOM, 1956), desde o primeiro nível, de lembrar e reconhecer os conceitos básicos, até mesmo o nível de aplicação no qual, tipicamente se adota o ciclo de use-modify-create (LEE et al., 2011).

No estágio de *use*, o aluno utiliza da criação de outros (LEE et al., 2011), como por exemplo, experimentar um programa de classificação de *deep learning* com nós pré-definidos, em busca de entender como o mesmo chega nas conclusões, e o quão bom é em fazer isso. No estágio de *modify*, o aluno pode modificar tal programa (LEE et



al., 2011), ajustando configurações como *epoch*, *batch size* e *learning rate*, por exemplo. Finalmente, na fase de *create*, o aluno é guiado a ter ideias novas, de projetos computacionais que queira realizar (LEE et al., 2011). Por exemplo, o estudante cria modelos novos de ML para desenvolver soluções úteis para comunidade, guiando o aluno a criar artefatos com um impacto na sua vida e/ou na comunidade, que permite perceber a presença e utilidade de tais habilidades no dia a dia adotando a ação computacional (TISSENBAUM et al, 2019).

Seguindo as tendências mundiais, e inspiradas pelas diretrizes citadas, a iniciativa Computação na Escola iniciou o desenvolvimento do curso Machine Learning para Todos! (GRESSE VON WANGENHEIM et al., 2020). O curso tem como alvo estudantes entre 10 e 14 anos, sem nenhum conhecimento prévio de ML. Porém, é assumido que o aluno possui entendimento de conceitos básicos de algoritmos e programação, ou desenvolvimentos de apps com o App Inventor em aulas anteriores. Seguindo as diretrizes de AI4K12, e alguns tópicos de *CSTA K–12 Computer Science Standards*, o curso possui objetivos de aprendizagem (GRESSE VON WANGENHEIM et al., 2020) apresentado na Tabela 3.

Tabela 3 - Objetivos de aprendizagem do curso Machine Learning Para Todos!. Traduzido de (GRESSE VON WANGENHEIM et al., 2020)

ID	Objetivo de Aprendizagem
OA1	Conhecer e identificar exemplos de aplicações de ML
OA2	Descrever conceitos básicos e o processo de Machine Learning, o que é rede neural
OA3	Identificar padrões nos dados
OA4	Simular o treinamento de uma rede neural simples
OA5	Coletar, limpar e rotular dados para o treinamento de um modelo de ML, entendendo como o algoritmo pode ser influenciado pelos dados
OA6	Treinar o modelo
OA7	Avaliar o desempenho do modelo de Machine Learning
OA8	Aplicar o modelo, desenvolvendo um aplicativo mobile
OA9	Discutir assuntos de ética e o impacto de ML na sociedade
OA10	Compartilhar o sistema desenvolvido
OA11	Propor iniciativas coletivas e individuais para resolver problemas ambientais na cidade ou comunidade

Dessa forma, o curso leva o aluno a entender não só os conceitos fundamentais de machine learning, mas sua relevância no dia a dia atual, e o impacto que tais soluções

podem ter na comunidade local, e suas implicações éticas, permitindo que o aluno possa ver a diferença que ele pode fazer para seus arredores. Além disso, o curso faz uso do ensino prático, guiando o estudante para que o próprio faça a preparação e o treinamento de um modelo e o implantando em um app, dando ao mesmo uma noção básica e palpável de como o processo é realizado.

Considerando a situação da pandemia, o curso foi esboçado e adaptado para ser aplicado de forma online. O plano de ensino desenvolvido é apresentado na Tabela 4.

Tabela 4 - Plano de ensino do curso Machine Learning Para Todos! Traduzido de Gresse von Wangenheim et al. (2020)

Aula	Conteúdo	Objetivos de aprendizagem	Método Educacional	
Noções Gerais e importância				
1. Aulas do curso online	Motivação de IA e sua aplicação na vida diária	OA1, OA9	Slides interativos, video, demonstrações	Quiz
Conceitos básicos				
2. Aulas do curso online	Conceitos Básicos de ML: o que significa "aprender"	OA2, OA3, OA4	Slides interativos	Quiz
Faça seu primeiro modelo (e aplicativo) de ML!				
3.1 Aulas do curso online	Classificação de recicláveis; preparação de dados (limpeza e rotulação)	OA3, OA5, OA11	Slides interativos Aula, atividade prática	Quiz, avaliação baseada no desempenho ante do dataset preparado
3.2 Aulas do curso online	Treinamento e avaliação do modelo	OA4, OA6, OA7	Slides interativos atividade prática com Google TM	Quiz, avaliação baseada no desempenho do modelo e avaliação de análise e interpretação
Atividade Extra Online	Implantação do modelo criado em um app ( <i>wireframe</i> disponível), customização do app	OA8	Slides Interativos	Avaliação baseada no desempenho do app criado
Atividade Extra Online	Compartilhamento do app criado	OA10	atividade prática ,	Avaliação baseada no desempenho do compartilhamento
Revisão do conteúdo e do processo de ML				
4.1 Aulas do curso online	Revisão dos conceitos e processo de ML	OA1, OA2, OA5	Slides Interativos atividade prática	Quiz
Problemas éticos e impacto social de ML				
4.2 Aulas do curso online	Problemas éticos a respeito de IA/ML, limitações, riscos e oportunidades de emprego	OA1, OA9	Slides interativos, demonstração ( <i>MIT Moral Machine</i> )	Quiz

O curso está disponível online com atividades interativas com h5p.org e coletando dados via o plugin H5PxAPIkatchu (TACKE, 2017).

Figura 1 - Curso Machine Learning para Todos



Fonte: (COMPUTAÇÃO NA ESCOLA, 2020 )

## 2.2 AVALIAÇÃO DA APRENDIZAGEM

A avaliação é uma parte importante do processo de ensino e aprendizagem, pois avalia o desempenho de algo ou alguém. Esse desempenho pode ser medido em uma nota atribuída ao objeto avaliado, e assim, um *feedback* personalizado é criado (BRANCH, 2009). A avaliação é importante tanto para o educando, para que o mesmo conheça seu desempenho, quanto para o educador, para avaliar a retenção de informação dos alunos e a efetividade de seus métodos de transferência de conhecimento (ALVES, 2019). De acordo com Stegeman et al. (2016), o aluno precisa saber: o que um bom desempenho em uma tarefa é; como seu próprio desempenho se compara com um bom; e o que fazer para reduzir a lacuna entre os dois.

Seguindo o design instrucional (BRANCH, 2009), existem dois tipos principais de avaliação: formativa e somativa. Formativa se refere ao processo de coletar dados que podem ser usados para revisar a instrução antes da implementação, e tem como objetivo determinar a potencial eficiência dos recursos de aprendizado durante o desenvolvimento, e identificar recursos que precisam ser revisados (BRANCH, 2009). Já a avaliação somativa é o processo de coletar dados após a implementação, para determinar sua eficiência. Tem como objetivo sintetizar a aprendizagem do aluno, evidenciando a perspectiva de conclusão do processo, visto que é realizada no final do mesmo. Esse tipo

é geralmente o mais usado em atividades abertas de programação, em que uma nota é atribuída ao artefato do aluno, após a conclusão do desenvolvimento.

Avaliações podem ser realizadas de diferentes maneiras. Uma alternativa no modo online é por meio de atividades interativas do H5P, um *framework* colaborativo de código aberto, possui diversos tipos de avaliação:

- Quiz
- Quiz Aritmético
- Caça-palavras
- *flashcards*
- Jogos de imagens (emparelhamento e sequenciamento de imagens)
- Jogo de memória
- Quiz de personalidade
- Cartas de diálogo
- Formulários (arrastar e soltar palavras, completar espaços em branco)
- Ache o *hotspot*
- *Checklists*
- Marque as imagens
- Questionários
- Questões de múltipla escolha

Além de avaliações tipo quiz, especialmente considerando uma metodologia ativa para o ensino de ML, muitas vezes usa-se uma avaliação por desempenho. Tipicamente a avaliação de desempenho é feita utilizando rubricas, um conjunto de critérios, níveis de desempenho e opcionalmente, descritores verbais que explicam os diferentes níveis (STEGEMAN et al., 2016). Por meio de rubricas que utilizam descritores verbais, é possível retornar um *feedback* para o aluno, garantindo que o mesmo saiba o que pode melhorar para atingir um desempenho melhor (STEGEMAN et al., 2016).

Ao desenvolver uma avaliação de desempenho, é importante ter em mente os objetivos de aprendizagem do assunto. A taxonomia de Bloom foi criada para auxiliar na classificação dos objetivos de aprendizagem em crescentes tipos de complexidade e especificidade (BLOOM, 1956). Quando se trata de educação básica, geralmente somente os 3 primeiros níveis são abordados. O primeiro nível, de Conhecimento, envolve reconhecer e lembrar fatos e conceitos básicos, sem necessariamente entender seu significado. (WOOLFOLK, 2007). O segundo nível, Compreensão, engloba a demonstração de um entendimento das ideias relevantes, por meio de resumos, generalizações, etc. O terceiro nível, de aplicação, engloba reunir o conhecimento e compreensão adquiridos e aplicá-los a certas situações, como por meio da soluções de problemas simples.

Tabela 5. Níveis de conhecimento de Bloom (BLOOM, 1956), relacionados com avaliações disponíveis no H5P. Adaptado de (CLICK4IT, 2013)

Domínio Cognitivo (Nível)	Exemplos de tipos de avaliação	Palavras Chave
1. Conhecimento (recordar informação)	quizzes, múltipla e única escolha, jogos de memória, preencher espaço em branco, arrastar palavras	Arranjar definir, descrever, rotular, listar, memorizar, reconhecer, relacionar, reproduzir, selecionar, citar
2. Compreensão (recordar e interpretar informação)	Formulários (onde o aluno explica sua interpretação de conceitos ou de um cenário), atividade de cenários ramificados, emparelhamento de imagens	explicar, reiterar, reformular, criticar, classificar, resumir, ilustrar, traduzir, revisar, reportar, discutir, reescrever, estimar, interpretar, teorizar, parafrasear, referenciar, exemplificar
3. Aplicação (usar informação abstrata em situações concretas)	Avaliação por desempenho	Usar, aplicar, descobrir, administrar, executar, resolver, produzir, implementar, construir, mudar, preparar, conduzir, performar, reagir, responder,

### 2.2.1 Notas e *Feedback*

Uma avaliação pode ser usada para calcular uma nota, fornecer *feedback* ou ambos. A nota age como uma indicação do nível de desempenho, um reflexo da pontuação de uma avaliação, mas não necessariamente fornece um *feedback* por si só. Uma nota pode ser qualitativa, atribuindo conceitos para diferentes níveis, como de 'A' a 'E', por exemplo, ou pode ser quantitativa, como uma nota entre 0 a 10. No Brasil, o MEC não possui nenhum padrão para atribuição de nota, porém a forma mais comum no Ensino Fundamental brasileiro é a quantitativa no intervalo [0, 10] (CME, 2011).

Prover *feedback* regularmente é importante, pois auxilia os educandos a ficarem engajados durante o processo de aprendizado, facilitando e incentivando o aprendizado (GONÇALVES, 2017). Um *feedback* pode ser abordado de duas principais maneiras: de verificação, quando serve para julgar se a resposta do aluno está correta ou não; e de elaboração, que apresenta comentários e sugestões, encorajando o próprio educando a identificar os aspectos de sua resposta ou trabalho que precisam ser melhorados, e como melhorá-los (KULHAVY & STOCK, 1989).

Tabela 6. Classificação tipos de *feedback* por complexidade (GONÇALVES, 2017)

Tipo de <i>Feedback</i>	Descrição
<b>Abordagem por verificação</b>	
Sem <i>feedback</i>	Refere-se às situações onde o aluno é apresentado a um problema ou questão, e precisa responder ao mesmo. Neste caso, nenhuma indicação sobre a exatidão da resposta do aluno é fornecida.
Resposta correta	Apresenta ao aluno a resposta correta para um problema específico, sem apresentar qualquer informação adicional.

Tente novamente	Informa ao aluno se a resposta está correta ou incorreta, e permite a ele uma ou mais tentativas para responder ao problema.
Marcação de erros	Destaca os erros presentes na solução apresentada pelo aluno, sem apresentar a resposta correta.
<b>Abordagem por elaboração</b>	
Isolação de atributo	Apresenta informações abordando individualmente os atributos centrais do conteúdo que está sendo estudado
Dicas e sugestões	Orienta o aluno, indicando o que ele deve fazer em seguida ou apresentando exemplos. Este tipo de feedback evita fazer uma apresentação explícita da solução correta.
Análise de erros	Realiza a análise de erros e diagnósticos. Apresenta informações específicas sobre os erros cometidos pelos alunos e esclarece conceitos mal compreendidos (e.g. o que está errado e por que).
Tutorial	Considerado o tipo mais complexo de feedback (NARCISS & HUTH, 2004), realiza a marcação de erros, apresenta informações relacionadas ao conteúdo, e provê dicas ou exemplos de como proceder. A resposta correta pode ou não ser apresentada.

Além disso, na última década o uso de técnicas de gamificação vêm ganhando espaço na educação. Também chamada de ludificação, consiste na prática de incorporar elementos de video games em serviços, websites, educação, etc. (O'BRIEN, 2010). Estes elementos incluem: barras de progresso, emblemas e níveis de conquistas, classificação de usuários, entre outras (O'BRIEN, 2010). No contexto de educação, estes elementos podem ajudar no engajamento e motivação do aluno durante o processo de aprendizagem (GONÇALVES, 2017).

### 2.2.2 Automação da avaliação

A avaliação muitas vezes é feita manualmente pelo próprio instrutor, o que pode requerer um tempo considerável por parte do mesmo, além de estar sujeita a erros humanos. Existem maneiras de criar avaliações automatizadas, por meio de softwares que avaliam diferentes fatores de qualidade. A automatização permite agilizar o processo, requer menos tempo do educador e reduz a possibilidade de erro humano da avaliação. Kay et al. (1994) afirmam que não é possível avaliar os programas dos alunos de forma consistente e integral sem ajuda automatizada. Portanto, é perceptível que a automatização, de forma geral, melhora o processo de avaliação.

Em relação a artefatos de software, ela pode se dar por meio da análise estática (caixa branca) ou análise dinâmica (caixa-preta), ou ambas. A análise estática envolve coletar informações sobre o programa sem executá-lo, ou seja, analisando puramente o código fonte e suas características. Esse tipo de avaliação geralmente leva em conta aspectos como o estilo de código, erros de programação, métricas de software e design

do programa. Já a análise dinâmica envolve a execução do programa, avaliando características do programa como sua funcionalidade, eficiência e habilidades de testes (ALA-MUTKA, 2005).

## 2.3 DEEP LEARNING

*Neural Networks* (Redes Neural) é uma área do estudo de *Machine Learning* que se inspira no funcionamento dos cérebros dos animais. Uma rede neural consiste de vários processadores interconectados chamados de *neurons* (neurônios), cada um produzindo uma sequência de ativações de valores reais. No contexto de ML, os neurônios podem ser ativados pelas conexões ponderadas dos neurônios anteriores, assim, o aprendizado está em descobrir os pesos que fazem com que a RN exiba o comportamento desejado. Quanto mais níveis de neurônios são adicionados, mais profunda a rede fica (SCHMIDHUBER, 2015). Portanto, *Deep Learning* é a técnica de ML que envolve usar redes neurais profundas, ou seja, com muitas camadas de neurônios, que têm seus pesos ajustados a cada iteração de aprendizado. Essa área tem uma aplicação ampla, podendo ser usada em aprendizagem supervisionada, semi-supervisionada e não supervisionada (SCHMIDHUBER, 2015).

*Supervised Learning* (Aprendizagem Supervisionada) é a tarefa de ML que consiste em aprender uma função que mapeia uma entrada a uma saída, baseada em exemplos em pares de entrada-saída (RUSSELL; NORVIG, 2010). Ou seja, pode ser usada quando tanto as entradas quanto as saídas corretas e erradas são conhecidas. Modelos de aprendizagem supervisionada podem ser utilizados em problemas de classificação, em que é preciso diferenciar entre duas ou mais classes, como por exemplo, definir se um e-mail é spam ou não, identificar se dada imagem contém um gato ou um cachorro, classificar o tipo de determinado lixo contido em uma foto, etc.

*Unsupervised Learning* (Aprendizagem não Supervisionada) é um tipo de atividade de ML que, ao contrário da supervisionada, não possui rótulos ou entradas pré definidas, buscando detectar padrões não percebidos anteriormente, fazendo uso mínimo de intervenção ou supervisão humana (HINTON; SEJNOWSKI, 1999). Técnicas de aprendizagem não supervisionada são usadas em problemas de clusterização, em que é preciso agrupar entradas baseado em determinadas categorias, e de reconhecimento de padrões, como reconhecimento automático de alvos ou obstáculos, processamento de sinais sísmicos, etc (CARPENTER; GROSSBERG, 1988).

Por fim, a *Semi-Supervised Learning* (Aprendizagem Semi-Supervisionada) está em algum lugar entre a supervisionada e a não supervisionada, geralmente usando um

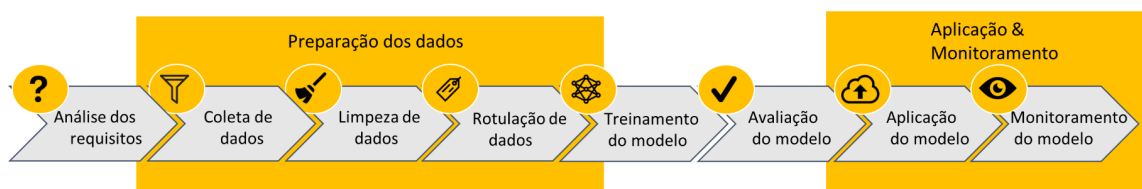
paradigma como base e adotando elementos do outro. Classificação semi-supervisionada, por exemplo, é uma das técnicas de aprendizagem semi-supervisionada que consiste em realizar o treino do modelo com dados rotulados e não rotulados, de forma que ele tenha um desempenho melhor do que um modelo supervisionado treinado somente com dados supervisionados (ZHU; GOLDBERG, 2009). Pode ser usada para um classificador de documentos de texto, treinado com uma quantidade pequena de documentos de texto rotulados, em conjunto com uma grande quantidade de não rotulados, visto que a rotulação manual de centenas de documentos de texto para preparar para o treino seria inviável.

### 2.3.1 Processo de *Machine Learning*

Desenvolver uma aplicação de ML é um processo iterativo, que envolve diferentes passos consecutivos. Os exatos passos e detalhes característicos de cada passo podem variar de acordo com o desenvolvedor, porém, de forma geral incluem (AMERSHI et al., 2019)(AMAZON, 2020):

Figura 2 - Processo de *Machine Learning*, slide retirado de (GRESSE VON WANGENHEIM, 2020)

## Quais são os passos para desenvolver um modelo de *Machine Learning*?



Iniciativa Computação na Escola - Creative Commons Atribuição-NãoComercial-Compartilhável 4.0 Internacional. Imagens: freepik.com

- 1. Análise de requisitos.** A primeira parte do processo, é quando decide-se que tipo de modelo é mais apropriado para o problema, e também uma decisão inicial de quais *features* serão viáveis e mais relevantes para o mesmo .
- 2. Coleta de dados.** Envolve encontrar e integrar conjuntos de dados, caso seja possível, se não, é necessário fazer efetivamente a coleta dos dados (AMERSHI et al., 2019). A quantidade, tipo e formato de dados varia de acordo com o problema e com quem desenvolve a solução .



3. **Limpeza de dados.** Quando se reúne uma grande quantidade de dados, é inevitável que falhas ou imprecisões ocorram, seja na coleta ou integração dos mesmos. A limpeza busca remover dados imprecisos, incompletos ou com muita interferência .
4. **Rotulação de dados.** Para cada entrada, é atribuído um rótulo relevante ao domínio do problema e ao modelo escolhido .
5. **Treinamento do modelo.** Para alguns modelos essa etapa é precedida de um passo de Engenharia de *features*, porém, como esse trabalho foca em *Deep Learning*, ambas etapas geralmente se mesclam devido à natureza das redes neurais. O modelo é treinado e ajustado utilizando os dados selecionados, limpos e rotulados. Durante o processo, diferentes parâmetros de treinamento são ajustados para otimizar o modelo. Alguns desses parâmetros são (RADHAKRISHNAN, 2017):
  - **Learning Rate:** Define o quão rápido e em que proporção a rede neural atualiza seus parâmetros. Um valor de *learning rate* baixo resulta em um processo de aprendizado mais demorado, mas que converge mais suavemente, enquanto um valor alto agiliza o processo, mas pode resultar em um modelo que nunca converge.
  - **Número de Epochs:** É a quantidade de vezes que o conjunto de dados de treinamento inteiro passa pela rede.
  - **Batch Size:** É o número de subconjuntos dos dados de treinamento que são apresentados à rede até que uma atualização de parâmetros é realizada. Geralmente é feita em potências de 2 a partir de 32.
6. **Avaliação do modelo.** O modelo é testado no conjunto de dados de teste, e seu desempenho é avaliado, o qual também depende do domínio e tipo do problema, assim como quem o avalia. Sistemas mais críticos podem depender de extensa avaliação humana (AMERSHI et al., 2019), enquanto outras aplicações podem precisar apenas da análise das correspondências entre os resultados do modelo e a rotulação humana definida em fases anteriores. Por isso, existem diferentes maneiras de avaliar um modelo, que são muitas vezes usadas em conjunto. Entre outras, existem (POWERS, 2008):
  - **Matriz de Confusão:** Ajuda na visualização do desempenho do modelo, mostrando em cada linha as instâncias preditas das respectivas classes, enquanto cada coluna representa a instância real de cada classe, ou vice-versa . Assim, ela permite determinar a quantidade de positivos e

negativos verdadeiros, e falsos positivos e negativos. A partir dela, são derivadas diversas outras estatísticas, como:

- **Recall ou Sensitividade:** é a proporção de positivos reais que são corretamente preditos como positivos (positivos verdadeiros).
- **Precisão ou Confiança:** é a proporção entre positivos preditos pelo modelo e positivos verdadeiros.

**7. Implantação do modelo.** O resultado final pode ser implantado no dispositivo ou sistema alvo, em um ambiente de produção, pronto para receber e processar novas entradas.

**8. Monitoramento do modelo.** O modelo em produção é constantemente monitorado para garantir que está agindo como esperado, e não apresenta falhas. A tolerância a imprecisões e erros depende da criticidade do domínio.

Quando se trata do desenvolvimento de modelos de ML, há inúmeras maneiras de abordar o mesmo problema. Com a popularização da tecnologia e crescente número de estudos sobre o assunto, diferentes arquiteturas, *frameworks* e bibliotecas de código foram desenvolvidas para auxiliar no desenvolvimento de RN, que muitas vezes englobam o processo de ML inteiro.

Rede Neural Convolutiva (CNN) é uma classe de arquitetura de ML, atualmente bastante aplicada em problemas relacionados com análise de imagens. Além das camadas de entrada e saída, consistem de uma camada convolutiva, uma camada de *pooling* e um classificador, que processa os *features* obtidos nas camadas anteriores (VALUEVA et al., 2020).

Quanto a programação de fato, existem diversas bibliotecas de código, como TensorFlow (TENSORFLOW, 2015), uma biblioteca de código aberto para criação, treinamento e exportação de redes neurais, disponível para diferentes plataformas e linguagens. Além disso, ferramentas como o *Google Teachable Machine* (TEACHABLE MACHINE, 2022) permitem a criação de redes de classificação de imagens com abstrações de alto nível, sem a necessidade de conhecimentos de programação.

### 3. ESTADO DA ARTE

O estado da arte representa a atual condição de conhecimento do assunto tratado, e é usado como ponto de partida para a pesquisa. Para demonstrar a relevância do projeto, e ajudar no desenvolvimento do mesmo, é importante analisar o estado da arte atual, realizando uma busca por outras unidades instrucionais de ML que possuam métodos de avaliação.

#### 3.1 DEFINIÇÃO DO PROTOCOLO DE REVISÃO

A pergunta de pesquisa é: Quais modelos de avaliação existem para automaticamente avaliar a aprendizagem de ML na educação básica? Assim, a pergunta é dividida nas seguintes questões de análise :

Q1. Que abordagens de avaliação existem?

Q2. Que conceitos relacionados à ML são analisados?

Q3. Como esses conceitos de ML são analisados?

Q4. Notas são geradas? Se sim, como?

Q5. *Feedback* instrucional é apresentado? Se sim, como?

Q6. A abordagem foi automatizada? Se sim, como?

**Crítérios de inclusão e exclusão:** Foram considerados apenas modelos de avaliação que focam no ensino de conceitos de *Machine Learning*. Foram incluídas unidades instrucionais voltadas ao ensino de ML para jovens do ensino fundamental, que apresentam algum tipo de método de avaliação. Foram excluídas UIs com foco para ensino médio ou superior, assim como publicações como blogs, vídeos ou ferramentas que não disponibilizam uma unidade instrucional. São considerados modelos de avaliação manuais e automatizados, e a busca foi limitada para artigos em inglês publicados nos últimos 10 anos.

**Crítério de qualidade:** Foram considerados apenas materiais em que existe algum método de avaliação explícito, seja manual ou automatizado.

**Fonte dos dados:** Foram examinados artigos e materiais publicados na língua inglesa, disponíveis na internet por meio das bibliotecas digitais mais proeminentes do campo (ACM Digital Library, IEEExplore, ERIC, sciencedirect, ArXiv). Para aumentar a cobertura e alcance da pesquisa, foi usado o Google, que por indexar grandes quantidades de

dados de diferentes fontes, pode auxiliar em encontrar materiais que não foram publicados como artigos científicos.

**Definição da frase de busca:** Observando a dificuldade de encontrar artigos apresentando modelos de avaliação de aprendizagem de ML por buscas com este enfoque foi adotada uma estratégia diferente de busca. Foram buscados de forma mais ampla artigos que apresentam unidades instrucionais voltadas ao ensino de ML na educação básica. Conseqüentemente, a frase de busca foi calibrada e adaptada baseada na sintaxe da respectiva fonte de dados, como apresentado na Tabela 7.

(teach\* OR education OR course OR MOOC OR learn\*) AND ("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning") AND ("k-12" OR school\* OR kids OR children OR teen\*)

Tabela 7 - Termos da busca.

Conceito	Sinônimo
<i>Machine Learning</i>	<i>artificial intelligence, deep learning, data science</i>
K-12	<i>school, kids, children, teen</i>
<i>Instructional Unit</i>	<i>teach, learn, education, course, MOOC</i>

Tabela 8 - Frases de busca para cada fonte.

Fonte	Frase de busca
ACM <a href="https://dlnext.acm.org/search/advanced">https://dlnext.acm.org/search/advanced</a>	[[Abstract: teach*] OR [Abstract: education] OR [Abstract: course] OR [Abstract: mooc] OR [Abstract: learn*]] AND [[Abstract: "machine learning"] OR [Abstract: "data science"] OR [Abstract: "artificial intelligence"] OR [Abstract: "deep learning"]] AND [[Abstract: "k-12"] OR [Abstract: school*] OR [Abstract: kids] OR [Abstract: children] OR [Abstract: teen*]] AND [Publication Date: (01/01/2011 TO *)]
ArXiv <a href="https://arxiv.org/search/advanced">https://arxiv.org/search/advanced</a>	Query: order: -announced_date_first; size: 100; date_range: from 2011-01-01 to 2021-12-31; include_cross_list: True; terms: AND abstract="artificial intelligence" OR "deep learning" OR "data science"; AND abstract=school OR kids OR children OR teen; AND abstract=teach OR learn OR education OR course OR mooc
ERIC <a href="https://eric.ed.gov/?advanced">https://eric.ed.gov/?advanced</a>	(teach* OR education OR course OR MOOC OR learn*) AND ("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning") AND ("k-12" OR school* OR kids OR children OR teen*) Since 2011
IEEE <a href="https://ieeexplore.ieee.org/search/advanced">https://ieeexplore.ieee.org/search/advanced</a>	((("Abstract":teach*) OR ("Abstract":education) OR ("Abstract":course) OR ("Abstract":MOOC) OR ("Abstract":learn*)) AND ((("Abstract":"machine learning") OR ("Abstract":"data science") OR ("Abstract":"artificial intelligence") OR ("Abstract":"deep learning"))) AND ((("Abstract":"k-12") OR ("Abstract":school*) OR ("Abstract":kids) OR ("Abstract":children) OR ("Abstract":teen*)) ) Filters Applied: 2011-2021)

ScienceDirect https://www.sciencedirect.com/search	(teach OR education OR learn) ("machine learning" OR "artificial intelligence" OR "deep learning") ("k-12" OR school OR kids OR children OR teen)
Google https://www.google.com/	"machine learning" teach "K-12" school

### 3.2 EXECUÇÃO DA BUSCA

A busca foi realizada no mês de fevereiro de 2021 pelo autor deste trabalho revisado pela orientadora. Foram obtidos os seguintes resultados da busca e seleção de artigos relevantes, aplicando os critérios de entrada e saída e selecionado como artigos relevantes somente aqueles sobre unidades instrucionais que também apresentam modelos de avaliação.

Tabela 9 - Resultados obtidos

Fonte	Quantidade de resultados da busca	Quantidade de artigos analisados	Quantidade de artigos potencialmente relevantes	Quantidade de artigos relevantes apresentando avaliação da aprendizagem do aluno
ACM	594	200	14	3
ArXiv	85	85	5	1
ERIC	540	200	3	0
IEEE	704	200	7	1
ScienceDirect	38,461	200	7	0
Google	18.100.000	500	50	9
<b>Total (sem duplicatos)</b>				<b>14</b>

### 3.3 ANÁLISE DOS RESULTADOS

#### 3.3.1 Que abordagens de avaliação existem?

Comparado à quantidade de unidades instrucionais analisadas, foram encontrados muito poucos resultados que também apresentam avaliação. Foram encontradas 12 unidades focadas no ensino de IA e ML no nível de ensino fundamental, e 2 que abrangiam o público geral, totalizando 14 resultados. Aqueles com público mais abrangente foram considerados como relevantes por serem adequados para todas idades, incluindo o público alvo de jovens nos anos finais do ensino fundamental.

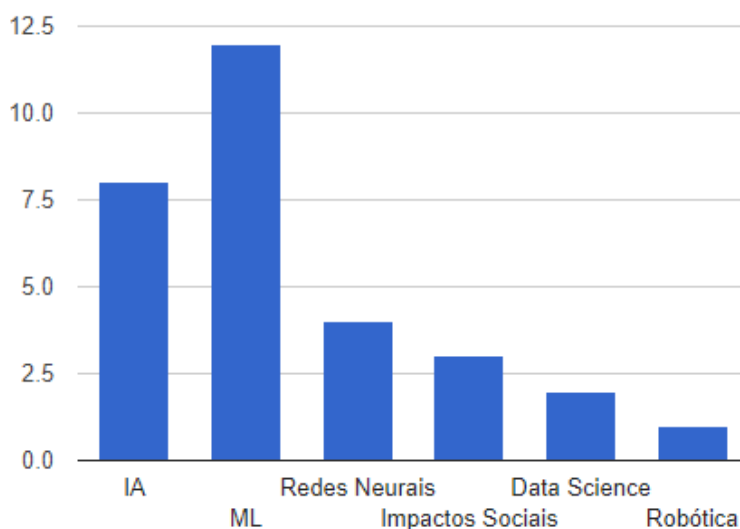
Tabela 10. Que abordagens de avaliação existem e quais suas características?

Referência	Título	Nível educacional
(AI FAMILY CHALLENGE, 2021)	AI Family Challenge	Público geral
(AIINSCHOOLS, 2021)	AlinSchools	Educação Básica
(APPS FOR GOOD, 2021)	Apps for good: ML Standard	Educação Básica
(CODE.ORG, 2021)	AI for Oceans	Educação Básica
(ELEMENTS OF AI, 2021)	Elements of AI	Público geral
(GRESSE VON WANGENHEIM et al, 2020)	Machine Learning for All – Introducing Machine Learning in K-12	Educação Básica
(HEINEMANN et al., 2018)	Drafting a Data Science Curriculum for Secondary Schools	Educação Básica
(KANDLHOFER et al., 2019)	Enabling the Creation of Intelligent Things: Bringing Artificial Intelligence and Robotics to Schools	Educação Básica
(MIT App Inventor, 2021a)	Personal Image Classifier (Appinventor)	Educação Básica
(MIT App Inventor, 2021b)	Introduction to ML: Image Classification (Appinventor)	Educação Básica
(ReadyAI, 2021)	Ready AI AI+Me	Educação Básica
(SAKULKUEAKULSUK et al., 2018)	Kids making AI: Integrating Machine Learning, Gamification, and Social Context in STEM Education	Educação Básica
(SRIKANT; AGGARWAL, 2017)	Introducing Data Science to School Kids	Educação Básica
(VAN BRUMMELEN; HENG; TABUNSHCHYK, 2020)	Teaching Tech to Talk: K-12 Conversational Artificial Intelligence Literacy Curriculum and Development Tools	Educação Básica

### 3.3.2 Que conceitos relacionados à ML são avaliados em quais níveis de aprendizagem?

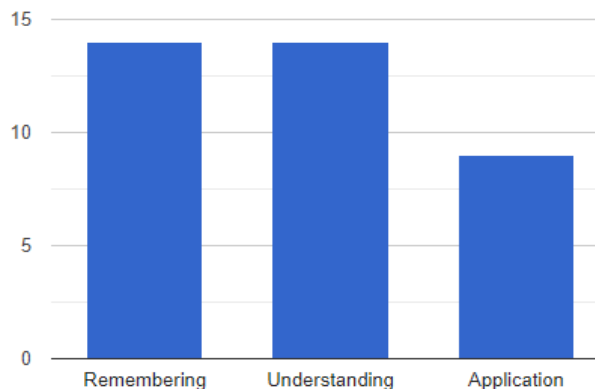
Todas as unidades abordam os conceitos básicos de ML. Porém poucos vão além disso, abordando também outros tópicos básicos de *Data Science* ou robótica. Outros também abordam temas como impactos sociais e implicações éticas da área. De forma geral, os conceitos são apresentados de forma abstrata, utilizando metáforas, e/ou por meio de aplicações práticas, com exemplos reais (Figura 3).

Figura 3 - Frequência dos conceitos abordados



A maioria foca somente nos níveis de aprendizagem (BLOOM, 1956) mais baixos de *remembering* e *understanding*, apenas apresentando os conceitos e exemplificando, a fim de questionar o aluno sobre os mesmos (Figura 4). Uma quantidade considerável também chega ao nível de aplicação, mostrando um pouco do lado prático dos assuntos abordados.

Figura 4 - Frequência dos 3 níveis de Bloom (1956)



Visando a aplicação do conteúdo, a maioria foca no estágio de *use*, guiando os alunos a desenvolver um modelo de ml predefinido, por exemplo. Poucas exceções chegam a guiar o aluno a criar a sua própria solução inteligente com ML (Figura 5).

Figura 5 - Frequência dos estágios do ciclo *use-modify-create*

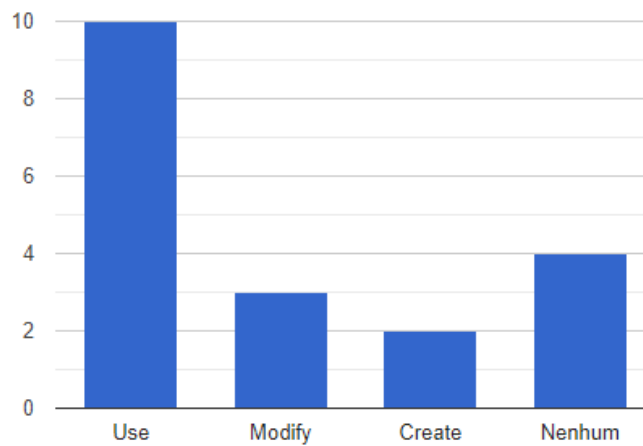


Tabela 11 - Que conceitos relacionados à ML são avaliados?

Referência	Conceitos Abordados	Nível de aprendizagem (Bloom)	Estagio(s) do <i>use-modify-create cycle</i>
(AI Family Challenge, 2021)	Conceitos básicos de IA e ML	Remembering, understanding, application	<i>Use</i>
(AlinSchools, 2021)	Conceitos básicos de IA e ML	Remembering, understanding, application	<i>Use, Modify</i>
(Apps for Good, 2021)	Conceitos básicos de ML	Remembering, understanding, application	<i>Use, Create</i>
(Code.org, 2021)	Conceitos básicos de ML	Remembering, understanding	--
(Elements of AI, 2021)	Conceitos básicos de IA, ML e Redes Neurais	Remembering, understanding	--
(GRESSE VON WANGENHEIM et al, 2020)	Conceitos básicos de IA, ML e Redes Neurais, impactos sociais	Remembering, understanding, application	<i>Use</i>
(HEINEMANN et al., 2018)	Conceitos Básicos de ML, Data science e impacto social	Remembering, understanding, application	<i>Use, Modify</i>
(KANDLHOFER et al., 2019)	Conceitos Básicos e Avançados de IA, Conceitos Básicos de ML, Conceitos Básicos e avançados de Robótica	Remembering, understanding, application	<i>Use, Modify</i>
(MIT App Inventor, 2021a)	Conceitos básicos de Redes Neurais	Remembering, understanding, application	<i>Use</i>
(MIT App Inventor, 2021b)	Conceitos básicos de Redes Neurais	Remembering, understanding, application	<i>Use</i>
(ReadyAI, 2021)	Conceitos básicos de IA e ML	Remembering, understanding	--
(SAKULKUEAKULSU K et al., 2018)	Conceitos básicos de IA e ML	Remembering, understanding	<i>Use</i>
(SRIKANT; AGGARWAL, 2017)	Conceitos básicos de ML e <i>Data Science</i>	Remembering, understanding, application	<i>Use, Create</i>
(VAN BRUMMELEN; HENG; TABUNSHCHYK, 2020)	Conceitos básicos de IA e ML, impactos sociais e éticos	Remembering, understanding,	--



### 3.3.3 Como esses conceitos de ML são analisados?

Grande parte dos resultados relevantes usa algum tipo de quiz, questionário ou testes que incluem majoritariamente questões de múltipla escolha, provavelmente devido ao fato que são mais fáceis e rápidos e corrigir e/ou automatizar. Alguns poucos exemplos apresentam uma avaliação baseada em atividades práticas realizadas ou nos artefatos gerados pelo aluno e seu desempenho.

A maioria das abordagens não possui nenhum tipo de automação. Somente (ELEMENTS OF AI, 2021) e (AI FAMILY CHALLENGE, 2021), (GRESSE VON WANGENHEIM et al, 2020) possuem correção e indicação de progresso automáticas, enquanto (CODE.ORG, 2021) apresenta somente a indicação automática de progresso.

Figura 6 - Frequência de níveis de automação da avaliação

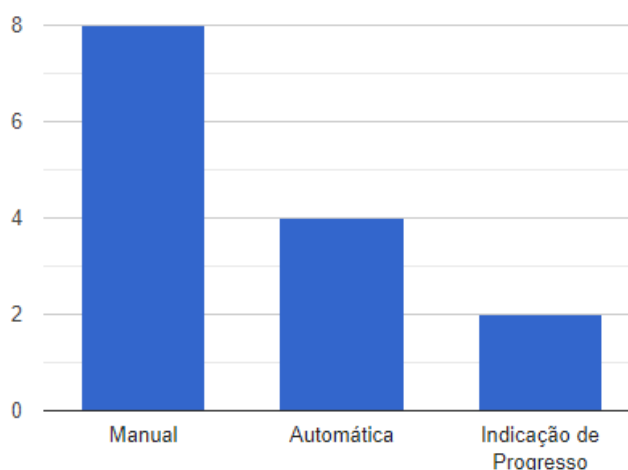


Tabela 12 - Como esses conceitos de ML são analisados

Referência	Tipo de Avaliação	Automação da avaliação
(AI Family Challenge, 2021)	Quiz	Correção automática de respostas
(AlinSchools, 2021)	Realização das atividades	Correção manual pelo avaliador
(Apps for Good, 2021)	Provas com questões variadas	Avaliação manual realizada pelo instrutor
(Code.org, 2021)	Realização das Tarefas	Indicação automática de progresso e certificado. <i>Feedback</i> para instrutores
(Elements of AI, 2021)	Exercícios com questões abertas ou de múltipla escolha	Correção automática de questões de múltipla escolha. Correção realizada pelos colegas. Nenhuma nota atribuída e o progresso é indicado automaticamente
(GRESSE VON WANGENHEIM et al, 2020)	Quizzes e Avaliação baseada nos artefatos criados	Avaliação automática dos quizzes, e manual dos artefatos criados pelos alunos
(HEINEMANN et al., 2018)	Baseada nos resultados do projeto final	Avaliação manual realizada pelo instrutor
(KANDLHOFER et al., 2019)	Questionário	Avaliação manual realizada pelo instrutor
(MIT App Inventor, 2021a)	Teste de Múltipla escolha (3 questões)	Avaliação manual realizada pelo instrutor

(MIT App Inventor, 2021b)	Teste de Múltipla escolha (3 questões)	Avaliação manual realizada pelo instrutor
(ReadyAI, 2021)	Quizzes com tipos de perguntas variadas	Correções de questões e indicação de progresso automática. Certificado.
(SAKULKUEAKULSU K et al., 2018)	Pontos acumulados de acordo com o desempenho dos modelos gerados	Avaliação manual realizada pelo instrutor
(SRIKANT; AGGARWAL, 2017)	Baseado no desempenho do modelo	Avaliação manual realizada pelo instrutor
(VAN BRUMMELEN; HENG; TABUNSHCHYK, 2020)	Questionários com questões abertas	Avaliação manual realizada pelo instrutor

### 3.3.4 *Feedback* instrucional é apresentado?

A maioria dos resultados apresentam algum tipo de *feedback*. Porém eles não são muito bem integrados ou explícitos durante o curso, muitas vezes vindo em forma de discussões sobre os resultados obtidos, ou apenas representando a correção de dado exercício, sem mesmo explicar o motivo das alternativas estarem certas ou erradas. Assim, a maioria dos resultados não provém sugestões de melhorias para os alunos.

Grande parte não possui nenhum tipo de nota, não apresentando nenhuma sugestão de quantificação do aprendizado do aluno. Das poucas unidades que possuem algum tipo de nota, (CODE.ORG, 2021) e (ELEMENTS OF AI, 2021) baseiam a mesma apenas no número de exercícios ou unidades completas, usando a nota como uma forma de representar o nível de completude do curso. AlinSchools (2021) utiliza um sistema de gamificação para a nota, onde cada exercício produz uma certa quantidade de pontos, que são somados e totalizados, para definir uma equipe “vencedora” de alunos.

Tabela 13 - Presença de nota ou *feedback*

Referência	Nota	<i>Feedback</i>
(AI Family Challenge, 2021)	--	Sugestões de recursos instrucionais e atividades adicionais
(AlinSchools, 2021)	Pontuação de 0-100	São promovidas discussões entre os alunos baseadas nos trabalhos de seus colegas
(Apps for Good, 2021)	--	--
(Code.org, 2021)	Pontuação em relação ao % do curso completado	--
(Elements of AI, 2021)	Quantidade de exercícios corretos	<i>Feedback</i> sobre as questões respondidas, explicando o motivo de certa resposta ser a correta
(GRESSE VON WANGENHEIM et al, 2020)	--	<i>Feedback</i> nos quizzes indicando respostas corretas/erradas
(HEINEMANN et al., 2018)	--	--
(KANDLHOFER et al., 2019)	--	--
(MIT App Inventor, 2021a)	--	Discussões no final de cada aula sobre as

		atividades realizadas, podendo incluir <i>feedback</i> de como melhorar tais atividades
(MIT App Inventor, 2021b)	--	Discussões no final de cada aula sobre as atividades realizadas, podendo incluir <i>feedback</i> de como melhorar tais atividades
(ReadyAI, 2021)	--	--
(SAKULKUEAKULSUK et al., 2018)	Sistema de pontos acumulados	--
(SRIKANT; AGGARWAL, 2017)	--	<i>Feedback</i> do instrutor sobre o exercício no final de cada aula
(VAN BRUMMELEN; HENG; TABUNSHCHYK, 2020)	--	--

### 3.4 DISCUSSÃO

Com todos os resultados obtidos que focam no ensino para jovens na faixa etária do ensino fundamental, a maioria usa certos elementos lúdicos e outros meios de atrair e reter a atenção dos alunos. Além disso, muitos visam explicar de forma simples os conceitos básicos de IA e ML, como os meios que computadores percebem o mundo, como aprendem a partir de exemplos, etc.

Alguns dos resultados relevantes vão além de apenas conceitos e fatos básicos sobre o assunto, como também abordam as implicações e impactos sociais das tecnologias, e como fazer uso de forma ética das mesmas. Muitos dos selecionados também vão além de apenas apresentar os conceitos, mas fornecem a oportunidade do aluno de entrar em contato e experimentar com modelos reais, e aplicar soluções que impactem a comunidade a sua volta.

Quanto à avaliação, dentre todos os materiais obtidos, apenas 4 apresentam métodos de avaliação automática, com CODE.ORG (2021) apresentando apenas uma representação automática de progresso. É perceptível não só a falta de métodos de avaliação, como a automação dos mesmos é rara.

Observando os resultados da busca, observa-se que mesmo com a existência de um número considerável de cursos, tutoriais e outros tipos de materiais educacionais focados no ensino de *machine learning* para o ensino fundamental, a maioria deles não apresenta nenhuma forma de *feedback* explícito. Além disso, poucos dos resultados apresentam notas, com um número menor ainda de exemplos que a nota representa mais que a quantidade de atividades completadas.

**Ameaças à validade.** Revisões sistemáticas do estado da arte podem apresentar resultados tendenciosos, visto que resultados positivos têm mais chances de serem

publicados que negativos. Porém, para essa análise as abordagens de avaliação foram consideradas mais importantes que o sucesso ou não do material, portanto, esse risco é considerado mínimo. Há também o risco de que algum resultado relevante tenha sido omitido, seja por tendências nos motores de busca das fontes de dados, ou por algum resultado ter passado despercebido pelo autor. Para mitigar tal risco, a frase de busca foi construída com o objetivo de maximizar a inclusividade da mesma, considerando não só os termos desejados como também seus sinônimos e palavras relacionadas, seguindo modelos de outros trabalhos com buscas semelhantes, e realizando buscas em várias bases. Ameaças relacionadas à seleção de materiais relevantes foram mitigadas por meio da definição detalhada dos critérios de inclusão e exclusão e a revisão e discussão do autor do presente trabalho com a orientadora, até obter consenso.

## 4. DESENVOLVIMENTO DO MODELO DE AVALIAÇÃO

Com o estado da arte revisado, foi possível dar início ao processo de desenvolvimento do modelo de avaliação. Primeiramente, foi realizada a análise do contexto em que o curso está inserido, analisando a realidade brasileira do ensino de computação para o ensino fundamental, e seus desafios. Foi possível determinar que, de modo geral, muitas escolas têm o equipamento necessário para o ensino de computação (QEDU, 2019), porém faltam profissionais especializados para esse ensino (INEP, 2019).

### 4.1 ANÁLISE DO CONTEXTO

O público alvo do curso são alunos de escolas brasileiras com idade entre 10 a 14 anos, correspondente aos anos finais do ensino fundamental. Nesse estágio do aprendizado, é esperado que o aluno seja alfabetizado na língua nativa (Português do Brasil) e possua certa familiaridade, mas não fluência, com a língua inglesa ou espanhol. Além disso, considera-se que o aluno compreende em alto nível o mundo natural, tecnológico e individual, e possui suficiente desempenho na escrita, leitura e compreensão de suas experiências, para ampliar sua percepção e analisar contextos maiores nos anos seguintes (BNCC, 2019). Atualmente, mesmo com a inclusão do tópico de pensamento computacional na área de conhecimento de matemática na Base Nacional Comum Curricular (BNCC), a aplicação do ensino de computação ainda é rara em escolas brasileiras, especialmente quando se trata de IA no ensino fundamental, e conseqüentemente nenhum conhecimento sobre computação ou *machine learning* é esperado, portanto, assume-se que o aluno não os possua.

De acordo com QEDU (2017), os dados da Prova Brasil mostram que 55% dos estudantes do 9º ano das escolas públicas brasileiras têm pelo menos um computador em casa e 42% destes passam mais de três horas por dia lidando com tecnologias. Em Santa Catarina, é perceptível um aumento na presença da tecnologia na vida dos alunos, visto que esses números aumentam para 75% e 60%, respectivamente (QEDU, 2017).

De todas as escolas do Brasil, públicas e particulares, 80% delas possuem internet, com 66% sendo banda larga. Existem 30.201.159 computadores para uso dos alunos, para 179.533 escolas, uma média de aproximadamente 168 computadores por escola, e 34% apresentam laboratório de informática dedicado. Já no Estado de Santa Catarina, 97% têm internet, sendo 86% banda larga. São 2.223.376 equipamentos para 6.263 escolas, com uma média de 355 computadores para uso dos alunos por escola, e 40% possuem um laboratório de informática dedicado (QEDU, 2019). A partir de 2020,

motivadas pela pandemia, observa-se também melhorias em relação a infraestrutura tecnológica do aluno, como a prefeitura de Florianópolis por exemplo, que ofereceu chips com internet de 20GB mensais para mais de 30 mil estudantes da cidade (ESTÚDIO NSC, 2021)

No país inteiro, do 5º ao 9º ano existiam um total de 14.951.330 alunos em 2019 (aproximadamente 83 por escola), 491.148 (aproximadamente 78 por escola) destes sendo do estado de Santa Catarina (QEDU, 2019). Em média, cada turma tem 35 alunos e as aulas têm duração de 45 minutos (GARCIA, 2022).

Por outro lado, observa-se a falta profissionais com formação em computação para ministrar aulas computação. A procura por cursos de formação de professores em computação é consideravelmente mais baixa do que por aqueles considerados tradicionais, como história, matemática etc. Além disso, em 2019, de 10.932 matrículas apenas 786 pessoas concluíram seus cursos nessa área, mostrando uma alta taxa de evasão (INEP, 2019). Portanto, o ensino de computação nas escolas ocorre muitas vezes de forma extracurricular ou interdisciplinar, com professores formados em outras disciplinas ensinado de forma integrada conteúdo de computação nas suas disciplinas consideradas mais tradicionais, como história, geografia etc.

## 4.2 AVALIAÇÃO DO CURSO MACHINE LEARNING PARA TODOS!

Para a criação de uma avaliação que mede o aprendizado de ML do aluno, é preciso que todos os objetivos de aprendizagem sejam englobados em pelo menos uma das avaliações previstas no plano de ensino.

Tabela 3 - Objetivos de aprendizagem do curso Machine Learning Para Todos!. Traduzido de Gresse von Wangenheim et al. (2020)

ID	Objetivo de Aprendizagem
OA1	Conhecer e identificar exemplos de aplicações de ML
OA2	Descrever conceitos básicos e o processo de Machine Learning
OA3	Identificar padrões nos dados
OA4	Simular o treinamento de uma rede neural simples
OA5	Coletar, limpar e rotular dados para o treinamento de um modelo de ML, entendendo como o algoritmo pode ser influenciado pelos dados
OA6	Treinar o modelo
OA7	Avaliar o desempenho do modelo de Machine Learning
OA8	Aplicar o modelo, desenvolvendo um aplicativo mobile
OA9	Discutir assuntos de ética e o impacto de ML na sociedade
OA10	Compartilhar o sistema desenvolvido
OA11	Propor iniciativas coletivas e individuais para resolver problemas ambientais na cidade ou comunidade

De acordo com o plano de ensino (Tabela 4) a avaliação da aprendizagem do aluno foi dividida em duas etapas, a avaliação dos quizzes, e a do desempenho e são usadas em conjunto para calcular a nota final. O material das aulas está disponível no formato H5P (HP5.ORG, 2021) em forma de imagens, vídeos, exercícios, etc.. O material H5P também contém os quizzes, que são relacionados aos assuntos de suas respectivas aulas, tratando dos objetivos de aprendizado do curso. O curso está hospedado no Moodle da UFSC, o sistema de gestão da aprendizagem utilizado pela universidade (MOODLE, 2021), por meio do Moodle Grupos, e também de forma aberta no site de cursos da iniciativa Computação na escola (COMPUTAÇÃO NA ESCOLA, 2020 ).

Especificamente para a avaliação do desempenho foi utilizada a rubrica de desempenho do curso Machine Learning Para Todos! (GRESSE VON WANGENHEIM, 2020), envolvendo diferentes critérios relacionados com as atividades práticas realizadas e seus artefatos. As notas de ambas avaliações são utilizadas para o cálculo da nota final.

Tabela 4 - Plano de ensino do curso Machine Learning Para Todos! Traduzido Gresse von Wangenheim et al. (2020)

Aula	Conteúdo	Objetivos de aprendizagem	Método Educacional	
Noções Gerais e importância				
1. Aula 1 do curso online	Motivação de IA e sua aplicação na vida diária	OA1	Slides interativos, video, demonstrações	Quiz: Slide 4 ( <i>drag-and-drop</i> ), 10 (múltipla escolha), 19 (múltipla escolha), 21 (múltipla escolha), 23 (múltipla escolha)
Conceitos básicos				
2. Aula 2 do curso online	Conceitos Básicos de ML: o que significa "aprender"	OA2, OA3, OA4	Slides interativos	Quiz: Slides 27 (verdadeiro ou falso) e 38 ( <i>drag-and-drop</i> )
Faça seu primeiro modelo (e aplicativo) de ML!				
3.1 Aula 3 do curso online	Classificação de recicláveis; preparação de dados (limpeza e rotulação)	OA3, OA5, OA11	Slides interativos Aula, atividade prática	Quiz: Slides 13 ( <i>drag-and-drop</i> ) e 20 ( <i>drag-and-drop</i> ); Avaliação baseada no desempenho do dataset preparado
3.2 Aula 4 do curso online	Treinamento e avaliação do modelo	OA4, OA6, OA7	Slides interativos, atividade prática com Google™, e quizzes	Quiz: Slides 22 (múltipla escolha), 23 (múltipla escolha), 29 (múltipla escolha), 30 (múltipla escolha) e 42 ( <i>drag-and-drop</i> ); avaliação baseada no desempenho do modelo e avaliação de análise e interpretação
Revisão do conteúdo e do processo de ML				
4.1 Aula 5 do curso online	Revisão dos conceitos e processo de ML	OA1, OA2, OA5	Slides Interativos atividade prática	Quiz: Slide 23 ( <i>drag-and-drop</i> )
Problemas éticos e impacto social de ML				

4.2 Aula 6 do curso online	Problemas éticos a respeito de IA/ML, limitações, riscos e oportunidades de emprego	OA1, OA9	Slides interativos, demonstração ( <i>MIT Moral Machine</i> )	Quiz: Slides 13 (múltipla escolha), 20 (múltipla escolha), e 22 (múltipla escolha)
----------------------------	---	----------	---	--

Os objetivos de aprendizagem de 1 a 4, e o 9 são avaliados por quizzes. Por meio do H5P, é possível introduzir quizzes relevantes ao assunto no meio dos slides, criando um fluxo contínuo em que o aluno é apresentado a um conceito, e logo em seguida avaliado sobre o mesmo.

**OA1 - Conhecer e identificar exemplos de aplicações de ML.** Durante a primeira aula, para avaliar o primeiro objetivo, o aluno é apresentado a vários exemplos de aplicações e questionado sobre qual das opções utiliza inteligência artificial (Anexo A).

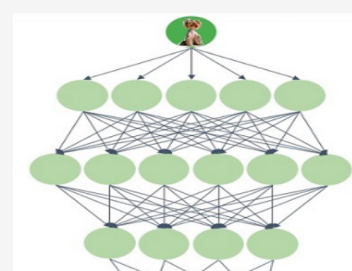
Tabela 14 - Quizzes da Aula 1

<p>Arraste as palavras para as caixas corretas (Dica: Animal)</p> <div style="display: flex; justify-content: space-between; align-items: center;"> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> <div style="border: 1px solid #ccc; width: 100px; height: 15px; background-color: #e0e0e0;"></div> </div> <div style="display: flex; justify-content: flex-end; margin-top: 10px;"> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">M</div> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">O</div> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">A</div> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">C</div> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">C</div> <div style="border: 1px solid #ccc; padding: 2px 5px; margin-bottom: 5px;">A</div> </div> <div style="text-align: center; margin-top: 10px;"> <div style="background-color: #007bff; color: white; padding: 5px 10px; border-radius: 5px; display: inline-block;">  Verificar         </div> </div>	<p>O app identifica (marque todas respostas corretas)</p> <div style="margin-bottom: 5px;"><input type="checkbox"/> grau da confiança na classificação do objeto</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> nome dos tipos de objetos</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> posição do objeto</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> quantas vezes o objeto já foi detectado</div> <div style="text-align: center; margin-top: 10px;"> <div style="background-color: #007bff; color: white; padding: 5px 10px; border-radius: 5px; display: inline-block;">  Verificar         </div> </div>
<p>Quais destas aplicações já usam Inteligência Artificial?</p> <div style="margin-bottom: 5px;"><input type="checkbox"/> Recomendações de produtos</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> Desbloquear o celular com imagem do rosto</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> Tradutor de texto online</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> Calculadora</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> Carros autônomos</div> <div style="text-align: center; margin-top: 10px;"> <div style="background-color: #007bff; color: white; padding: 5px 10px; border-radius: 5px; display: inline-block;">  Verificar         </div> </div>	<p>O que as máquinas ainda <b>não</b> fazem</p> <div style="margin-bottom: 5px;"><input type="radio"/> sentir saudades</div> <div style="margin-bottom: 5px;"><input type="radio"/> aprender e tomar decisões</div> <div style="margin-bottom: 5px;"><input type="radio"/> ver</div> <div style="margin-bottom: 5px;"><input type="radio"/> ouvir e responder</div> <div style="text-align: center; margin-top: 10px;"> <div style="background-color: #007bff; color: white; padding: 5px 10px; border-radius: 5px; display: inline-block;">  Verificar         </div> </div>
<p>Inteligência Artificial é <span style="float: right;">🔊</span></p> <div style="margin-bottom: 5px;"><input type="checkbox"/> um computador implantado no cérebro de humanos</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> o humano usando uma máquina artificial</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> uma pessoa artificialmente repetindo conhecimento</div> <div style="margin-bottom: 5px;"><input type="checkbox"/> a máquina reproduzindo coisas que normalmente precisam de inteligência humana</div> <div style="text-align: center; margin-top: 10px;"> <div style="background-color: #007bff; color: white; padding: 5px 10px; border-radius: 5px; display: inline-block;">  Verificar         </div> </div>	



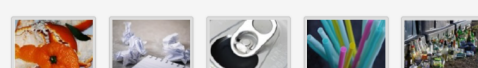


**OA2 - Descrever conceitos básicos e o processo de Machine Learning.** Para o segundo objetivo são realizados 3 quizzes: um sobre como as máquinas aprendem, outro sobre como se dá o processo de ML, e por fim, um quiz que teste a capacidade do aluno de identificar as diferentes camadas de uma rede neural. O primeiro e o último encontram-se na segunda aula, sobre os conceitos básicos de IA e ML, e o quiz de processos é realizado na aula 5, dedicada sobre o processo de *Machine Learning*.

Tabela 15 - Quizzes da Aula 2

<p>Máquinas aprendem com exemplos</p> <p><input type="radio"/> Verdadeiro <input type="radio"/> Falso</p> <p><a href="#">Verificar</a></p>	<p>ML Quiz 2.2 - Você sabe como são chamadas as camadas? Arraste os nomes às respectivas camadas</p>  <p>Camada de Entrada</p> <p>Camada de Saída</p> <p>Camada Escondida</p>
--	---

**OA3 e OA4 - Identificar padrões nos dados e simular o treinamento de uma rede neural simples.** Na terceira aula, sobre a preparação dos dados, os objetivos de aprendizagem 3 e 4, são avaliados por meio de um quiz onde imagens de diferentes tipos de lixo são apresentadas e devem ser categorizadas em tipos (vidro, papel, etc.) (Tabela 16).

Tabela 16 - Quizzes da Aula 3

<p>ML Quiz 3.1 - Classificação de lixo - arraste as imagens as categorias de lixo</p> <p>Metal <input type="checkbox"/> Papel <input type="checkbox"/> Plástico <input type="checkbox"/> Vidro <input type="checkbox"/> Não Reciclável <input type="checkbox"/></p>  <p><a href="#">Verificar</a></p>	<p>ML Quiz 3.2 - Limpeza de dados - arraste as imagens as categorias de tipo de imagens</p> <p>Dados bons <input type="checkbox"/></p>  <p>Dados desarrumados <input type="checkbox"/></p> 
--	---

**OA9 - Discutir assuntos de ética e o impacto de ML na sociedade.** Por fim, são realizados dois quizzes, sendo o primeiro sobre assuntos éticos onde entre várias situações, o aluno as categoriza entre fato, e questões subjetivas sujeitas a viés. O segundo sobre o impacto na sociedade, em que o aluno é apresentado a diferentes profissões, e deve marcar quais podem ser parcialmente ou completamente substituídas por máquinas no futuro.

Tabela 17 - Quizzes da Aula 4

<p><b>Accuracy per class</b></p> <table border="1"> <thead> <tr> <th>CLASS</th> <th>ACCURACY</th> <th># SAMPLES</th> </tr> </thead> <tbody> <tr> <td>vidro</td> <td>0.50</td> <td>4</td> </tr> <tr> <td>papel</td> <td>0.75</td> <td>4</td> </tr> <tr> <td>metal</td> <td>0.60</td> <td>5</td> </tr> <tr> <td>plástico</td> <td>1.00</td> <td>5</td> </tr> <tr> <td>não reciclável</td> <td>0.67</td> <td>3</td> </tr> </tbody> </table> <p>Analisando esta tabela de acurácia, você pode observar, que</p> <p><input type="radio"/> todas as categorias estão sendo reconhecidas com uma acurácia acima de 90%</p> <p><input type="radio"/> alguma(s) categorias são reconhecidos com acurácia abaixo de 90%</p> <p><input type="radio"/> não sei</p> <p><input checked="" type="radio"/> Verificar</p>	CLASS	ACCURACY	# SAMPLES	vidro	0.50	4	papel	0.75	4	metal	0.60	5	plástico	1.00	5	não reciclável	0.67	3	<p>Estes valores da acurácia indicam que:</p> <p><input type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado</p> <p><input type="radio"/> não sei</p> <p><input checked="" type="radio"/> Verificar</p>																		
CLASS	ACCURACY	# SAMPLES																																			
vidro	0.50	4																																			
papel	0.75	4																																			
metal	0.60	5																																			
plástico	1.00	5																																			
não reciclável	0.67	3																																			
<table border="1"> <thead> <tr> <th>Classes</th> <th>vidro</th> <th>papel</th> <th>metal</th> <th>plástico</th> <th>não reciclável</th> </tr> </thead> <tbody> <tr> <th>vidro</th> <td>1</td> <td>0</td> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <th>papel</th> <td>0</td> <td>3</td> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <th>metal</th> <td>0</td> <td>0</td> <td>5</td> <td>0</td> <td>0</td> </tr> <tr> <th>plástico</th> <td>0</td> <td>0</td> <td>1</td> <td>0</td> <td>2</td> </tr> <tr> <th>não reciclável</th> <td>0</td> <td>0</td> <td>0</td> <td>0</td> <td>3</td> </tr> </tbody> </table> <p>Analisando esta matriz de confusão, você pode observar, que:</p> <p><input type="radio"/> todas as imagens do conjunto de teste estão sendo classificadas corretamente</p> <p><input type="radio"/> alguma(s) das imagens de vidro são classificadas incorretamente como plástico</p> <p><input type="radio"/> alguma(s) das imagens de metal são classificadas incorretamente como papel</p> <p><input type="radio"/> alguma(s) das imagens de plástico são classificadas incorretamente como metal</p>	Classes	vidro	papel	metal	plástico	não reciclável	vidro	1	0	0	0	0	papel	0	3	0	1	0	metal	0	0	5	0	0	plástico	0	0	1	0	2	não reciclável	0	0	0	0	3	<p>Esta análise da matriz de confusão indica que:</p> <p><input type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado</p> <p><input type="radio"/> não sei</p> <p><input checked="" type="radio"/> Verificar</p>
Classes	vidro	papel	metal	plástico	não reciclável																																
vidro	1	0	0	0	0																																
papel	0	3	0	1	0																																
metal	0	0	5	0	0																																
plástico	0	0	1	0	2																																
não reciclável	0	0	0	0	3																																
<p>Arraste as palavras aos campos corretos</p> <p>Um <input type="text"/> representa quantas vezes o conjunto de dados passa pela rede que estamos treinando, para que ela analise eles e aprenda.</p> <p>Um <input type="text"/> é um subconjunto dos dados.</p> <p><input type="text"/> é o parâmetro que define quão rápido a rede neural irá aprender.</p> <p><input checked="" type="radio"/> Verificar resposta</p>																																					

Tabela 18 - Quizzes da Aula 6

<p>Possíveis riscos da Inteligência Artificial incluem (marque todas as alternativas corretas)</p> <p><input type="checkbox"/> Mudanças significantes no mercado de trabalho</p> <p><input type="checkbox"/> Tomada de decisão sem transparência</p> <p><input type="checkbox"/> Decisões tendenciosas</p> <p><input type="checkbox"/> Desalinhamento de valores</p> <p><input checked="" type="radio"/> Verificar resposta</p>	<p>Quais profissões têm alta chance de ser automatizadas pela adoção da Inteligência Artificial (marque todas as alternativas corretas)</p> <p><input type="checkbox"/> Atendentes de escritório</p> <p><input type="checkbox"/> Motoristas e entregadores</p> <p><input type="checkbox"/> Atendentes na caixa de supermercado</p> <p><input type="checkbox"/> Profissionais de manicure</p> <p><input checked="" type="radio"/> Verificar resposta</p>
---	---

Quais das seguintes profissões são promissoras mesmo em um futuro caracterizado pela automação com Inteligência Artificial (marque todas as alternativas corretas)

Desenvolvedor de sistemas de software com Inteligência Artificial

Telemarketing

Engenheiro industrial

Criador de novas vacinas

Figura 7 - Quiz da Aula 5

ML Quiz 5.1 - Arraste as atividades do processo de ML na ordem correta

Atividade 1	Atividade 2	Atividade 3	Atividade 4	Atividade 5

Cada aula recebe uma nota de 0 a 10 baseada no desempenho dos respectivos quizzes. A nota total dos quizzes é a média das notas das aulas.

O restante dos objetivos serão avaliados pela rubrica de desempenho, que realiza a avaliação baseada nos artefatos produzidos pelo aluno ao longo do curso, como o dataset, o modelo treinado, etc.

### 4.3 AVALIAÇÃO DE DESEMPENHO

Com base na rubrica proposta por Gresse von Wangenheim et al. (2021) foi instanciada uma versão da rubrica voltada ao curso de Machine Learning para Todos!. Considerando a complexidade de análise de certos critérios propostos na rubrica (GRESSE VON WANGENHEIM et al., 2021), foi neste momento excluída a automação dos seguintes critérios:

- Relevância das imagens (C2)
- Limpeza dos dados (C4)

Estes apresentam-se um desafio, demandando tempo de estudo de desenvolvimento, tendo em mente que cada um precisaria do desenvolvimento, teste e implantação do seu próprio modelo de classificação de imagens. Por isso, estes são considerados para trabalhos futuros.

Tabela 19 - Rubrica de avaliação de aprendizagem com base no desempenho (GRESSE VON WANGENHEIM et al., 2021)

ID	Critério	Níveis de desempenho			Coleta de dados	Extrair dos dados coletados	Proposta de automação
		Baixo - 0 pt.	Aceitável - 1 pt.	Bom - 2 pt.			
<b>Gerenciamento de dados (OA3/OA12)</b>							
C1	Quantidade de imagens	Menos de 5 imagens por categoria	5 a 10 imagens por categoria	Mais de 10 imagens por categoria	Passo 1. Imagens do arquivo .tm	Quantidade total de imagens do conjunto de dados	Contar a quantidade de imagens de cada classe
C2	Distribuição do conjunto de dados	Alguma categoria que representa mais do que 30% dos dados, ou menos que 20%)	Categorias representam entre 30% e 20% dos dados	Todas as categorias representam 25% dos dados	Passo 1. Imagens do arquivo .tm	Quantidade de imagens para cada categoria do conjunto de dados	Comparar a quantidade de cada classe com a quantidade total de imagens
C3	Rotulagem das imagens	Menos de 20% das imagens rotuladas corretamente	De 20% a 99% das imagens rotuladas corretamente	Todas as imagens rotuladas corretamente	Passo 1. Imagens do arquivo .tm	Quantidade de imagens rotuladas corretamente	Aplicar um modelo de alta precisão para a tarefa de classificação das 4 tipos de reciclagem (metal, papel, plástico e vidro) a partir dos dados extraídos
<b>Treinamento de modelo de ML/Transfer Learning e Fine-Tuning (OA4)</b>							
C4	Treinamento - Transfer Learning	O modelo não foi treinado	O modelo foi treinado com os parâmetros padrão	O modelo foi treinado com parâmetros ajustados (época e taxa de aprendizagem)	Passo 1. Arquivo manifest.json do .tm (parâmetros de treinamento)	Valor dos parâmetros da época, taxa de aprendizagem, batch size	Verificar qual foi a arquitetura do comando de definição do learner, retirar qual o número de épocas e taxa de aprendizagem do comando de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML (OA5)</b>							
C5	Análise de acurácia por categoria	Categorias com baixa acurácia não identificadas	--	Todas as categorias com baixa acurácia identificadas corretamente	Passo 3	Valores de acurácia e respostas do Passo 3	Verificar se o comando de display de categorias com baixa acurácia foi executado e se respostas no widget de interpretação usando ipywidgets estão corretas
C6	Interpretação de acurácia	Interpretação da acurácia incorreta	--	Interpretação da acurácia incorreta	Passo 3	Valores de acurácia e respostas do Passo 3	Verificar se as respostas de interpretação estão corretas
C7	Análise da matriz de confusão	Mais do que uma classificação incorreta não identificadas na matriz de confusão	Uma classificação incorreta não identificada na matriz de confusão	Todas as classificações incorretas identificadas na matriz de confusão	Passo 4	Matriz de confusão e respostas do Passo 4	Verificar as classificações indicadas como incorretas pelo aluno e compará-las com as classificações incorretas reais indicadas na matriz de confusão
C8	Interpretação da matriz de confusão	Interpretação errada da análise da matriz de confusão	--	Interpretação correta da análise da matriz de confusão	Passo 4	Matriz de confusão e respostas do Passo 4	Verificar se as respostas de interpretação estão corretas
C9	Ajustes/melhorias feitas	Sem novas iterações de desenvolvimento	Uma nova iteração com alterações no conjunto de dados e/ou parâmetros de treinamento	Diversas novas iterações com alterações no conjunto de dados e/ou parâmetros de treinamento	Passo 5	Respostas do Passo 5	Verificar quantidade de iterações em que houve alterações no conjunto de dados e/ou parâmetros de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML/Testar e aperfeiçoar programas (OA5/OA11)</b>							
C10	Testes com novos objetos	Nenhum novo objeto testado	1-2 novos objetos testados	Mais de dois novos objetos testados	Passo 2	Quantidade de testes realizados	Verificar a resposta sobre quantidade de vezes que o comando de treinamento do

							modelo foi executado com novos objetos
C11	Análise dos resultados de testes	Indicação errada da quantidade de erros nos testes	--	Indicação correta da quantidade de erros nos testes	Passo 2	Quantidade de erros na análise dos resultados dos testes	Verificar se as respostas indicam o número correto de erros nos tests
C12	Interpretação dos testes	Interpretação errada	---	Interpretação correta	Passo 6	Resposta da interpretação	Verificar se respostas de interpretação estão de acordo com o desempenho

Cada item recebe uma pontuação de 0 a 2, dependendo do desempenho. A partir das pontuações é calculada a nota final do aluno, dividindo a pontuação máxima por 2,4, para chegar numa nota de 0 a 10 em relação a rubrica.

Fórmula	Nota máxima
$(C1 + C2 + C3 + C4 + C5 + C6 + C7 + C8 + C9 + C10 + C11 + C12) / 2,4$	10

#### 4.4 NOTA FINAL

A nota final do curso, é feita por meio de uma média ponderada da nota dos quizzes e da nota da rubrica, com um peso de 1 e 4, respectivamente, com o objetivo de enfatizar a importância do desempenho dos artefatos do aluno (Tabela 21).

Tabela 20 - Cálculo da nota final

	Nota Máxima	Peso	Nota final
Quizzes	10	1	$((1 * \text{nota dos quizzes}) + (4 * \text{nota da rubrica})) / 5$
Rubrica	10	4	

## 5. AUTOMAÇÃO DO MODELO DE AVALIAÇÃO

Após o desenvolvimento do modelo, iniciou-se a automação da avaliação, que, assim como a avaliação, teve seu desenvolvimento dividido em duas partes, a automação da avaliação dos quizzes, e a do desempenho.

### 5.1 AUTOMAÇÃO DA AVALIAÇÃO DOS QUIZZES

A automação dos quizzes inseridos no material didático usando H5P (H5P.ORG, 2021), é realizada pelos próprios mecanismos da ferramenta, que disponibiliza correção e *feedbacks* automáticos a cada questão respondida.

Para fins de utilizar os resultados também como parte de uma avaliação geral no curso, esses podem ser coletados por meio do plugin H5PxAPIkatchu (TACKE, 2017) e/ou no contexto do sistema de gerenciamento de aprendizado do moodle nos módulos de avaliação. Assim, por meio dos quizzes automatizados, é possível avaliar a compreensão do aluno sobre os conceitos básicos de IA e ML, exemplos de aplicações que utilizam dessa tecnologia, seus impactos no mundo a sua volta, e suas implicações éticas (OA1, OA2, OA3, OA4 e OA9).

Os dados dos questionários do moodle são manualmente extraídos pela função de exportação da plataforma, no formato de um CSV.

### 5.2 AUTOMAÇÃO DA AVALIAÇÃO DA RUBRICA

A automação da avaliação de desempenho é baseada na rubrica apresentada na Tabela 19, usando como entrada os artefatos criados pelo aluno ao longo do processo de aprendizagem como resultado de atividades práticas. Para ter acesso a estes artefatos, o aluno os envia nos respectivos formulários, para que eles possam ser coletados automaticamente em conjunto com as respostas.

Os dados coletados referentes aos artefatos criados pelos alunos incluem o arquivo do modelo de ML treinado na *Google Teachable Machine* (TEACHABLE MACHINE, 2022) e baixado em formato *.tm*. A partir deste arquivo podem ser extraídas as imagens utilizadas no treinamento. Este arquivo também contém um arquivo *manifest.json*, que contém os parâmetros usados para o treinamento. Além disso, os demais dados necessários para realizar a avaliação baseada no desempenho foram coletados inicialmente por formulário via Limesurvey (LIMESURVEY, 2022), e posteriormente implementados no CodeMaster (DEMETRIO, 2017).

**C1 - Quantidade de imagens (OA5).** O item é avaliado iterando sobre a pasta descompactada de imagens do arquivo .tm e contando quantas imagens foram utilizadas para cada categoria. A categoria a qual a imagem pertence é indicada pelo seu nome. Com base na informação de quantidade de imagens por categoria é alocando a pontuação conforme definido na rubrica. Este item avalia o objetivo OA5 - Coletar, limpar e rotular dados para o treinamento de um modelo de ML, entendendo como o algoritmo pode ser influenciado pelos dados.

```
avalia_quantidade():  
  
para cada imagem no diretório:  
    soma quantidade de imagens em uma pontuação da respectiva classe  
    soma quantidade de imagens em uma pontuação global  
  
para cada quantidade de classes:  
    se quantidade < 5 -> pontuação = 0  
    se 5 <= quantidade < 11 -> pontuação = 1  
    senão -> pontuação = 2  
  
retorna quantidade total, quantidade de cada categoria, e pontuação
```

**C2 - Distribuição do conjunto de dados (OA5).** Com base nessas quantidade de arquivos por categoria, é possível identificar o total de imagens, e, dividindo as quantidades das classes pelo total, também encontra-se a proporção de cada categoria, automatizando assim a avaliação do item C3. Conforme a rubrica, são atribuídos 2 pontos a modelos que possuem a mesma proporção de imagens por categoria (25% para cada categoria). Para projetos com proporções entre 30% e 20% é dado 1 ponto, enquanto proporções que não se encaixam nessas categorias recebem 0 pontos para esse critério.

```
avalia_distribuição(total, quantidade_categoria):  
  
pontuação = 0  
  
se total > 0:  
    para cada quantidade de classes:  
        valor é dividido pelo total  
        se resultado == 0.25 -> pontuação = 2  
        se 0.30 > resultado > 0.20 -> pontuação = 1  
  
retorna pontuação
```

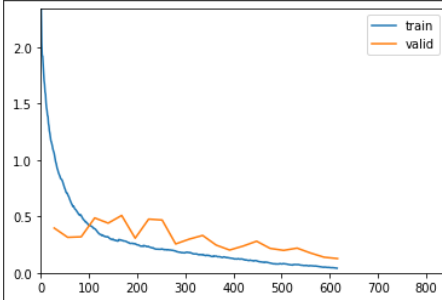
**C3 - Rotulagem das imagens (OA5).** Para automatizar o item C4, foi desenvolvido um modelo de *Deep Learning* usando FastAI e resnetX, com 95% de acurácia. Assim, é verificado se as imagens são rotuladas da mesma forma que o modelo prevê. Se todas as rotulações se igualam com a classificação dada pela rede, o aluno recebe 2 pontos, e com menos de 20%, são atribuídos 0 pontos. Para valores entre isso, é dado um ponto. O modelo desenvolvido é descrito na Tabela 21.

Tabela 21 - Resumo do Modelo de Machine Learning

Summary sheet - Modelo de ML											
Nome do modelo	Classificador de lixo reciclável para o curso Machine Learning para todos!										
Data	Julho 2021										
Versão	1.0										
Objetivo do modelo de ML											
Tarefa	Classificar/predizer imagens de tipos de lixo reciclável, para a avaliação automática de modelos dos alunos do curso.										
Contexto de uso	Uso para analisar a classificação dos modelos dos alunos do curso										
Público alvo	Cidadãos (8+ anos) Foco em alunos do Ensino Fundamental										
Riscos	Risco de classificar erroneamente, prejudicando a nota dos alunos										
Tipo da tarefa	Single-label classificação de imagens										
Categorias	4 categorias de lixo reciclável   Metal, Plástico, Papel, Vidro										
Conjunto de dados											
Descrição dos dados	Conjunto de imagens de objetos recicláveis capturada de uma câmera de celular.										
Origem dos dados (coleta própria/uso de conjunto de dados pré-existente, p.ex. do kaggle)	Conjunto de dados de lixo disponibilizado pela CnE Conjunto de dados disponibilizados no GitHub (garythung/trashnet)										
Quantidade total de dados	Total de 2.257 imagens										
Distribuição dos dados por categoria	<table border="1"> <caption>Quantidade por categoria</caption> <thead> <tr> <th>Categoria</th> <th>Quantidade</th> </tr> </thead> <tbody> <tr> <td>paper</td> <td>~650</td> </tr> <tr> <td>plastic</td> <td>~550</td> </tr> <tr> <td>metal</td> <td>~480</td> </tr> <tr> <td>glass</td> <td>~570</td> </tr> </tbody> </table>	Categoria	Quantidade	paper	~650	plastic	~550	metal	~480	glass	~570
Categoria	Quantidade										
paper	~650										
plastic	~550										
metal	~480										
glass	~570										
Labeling	Pelos autores dos conjuntos de dados, revisados pelo autor do projeto.										
Tipos de aumento de dados aplicados (tipo rotate, crop, etc.)	flip, rotate, zoom, warp, lighting										
Tamanho de imagens	224x224 pixels										
Tamanho do batch	64										
Dataset splitting	80% para treinamento (x imagens) 20% para validação (x imagens)										
Treinamento - Transfer learning											
Tipo de modelo	ResNet50										
Quantidade de épocas	22										
Taxa de aprendizagem	5 <sup>e-3</sup>										



**Curva de loss**



**Loss/taxa de erro por época**

epoch	train_loss	valid_loss	accuracy	time
0	1.072642	0.396833	0.858093	10:19
1	0.706180	0.314555	0.902439	09:52
2	0.513271	0.319105	0.911308	09:55
3	0.393647	0.486572	0.858093	10:00
4	0.320379	0.440417	0.886918	10:00
5	0.289484	0.508492	0.864745	10:06
6	0.259397	0.306916	0.906874	10:07
7	0.235439	0.476172	0.880266	10:05
8	0.212714	0.467893	0.891353	10:08
9	0.194930	0.256099	0.929047	10:08
10	0.179679	0.298449	0.895787	10:09
11	0.159581	0.331459	0.913526	10:11
12	0.148581	0.246130	0.924612	10:10
13	0.132379	0.201985	0.942350	10:15
14	0.121057	0.237372	0.920177	10:14
15	0.108177	0.280245	0.915743	10:15
16	0.087150	0.216615	0.924612	10:31
17	0.081858	0.199198	0.949002	10:23
18	0.072990	0.218984	0.935698	10:10
19	0.063586	0.175503	0.955654	10:09
20	0.049263	0.138500	0.953437	10:10
21	0.041024	0.126400	0.957871	10:10

**Avaliação - Transfer learning**

**Acurácia total** 0.958

**Acurácia por categoria**

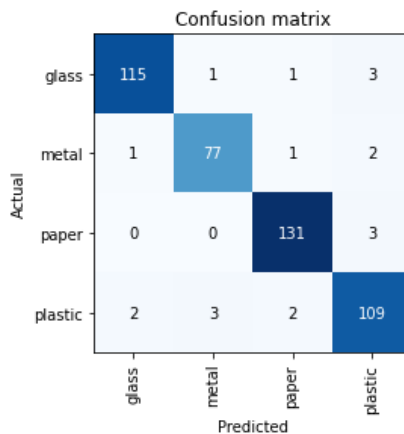
papel	0,99
plástico	0,94
metal	0,95
vidro	0,96

**Precisão** 0.96

**Recall** 0.96

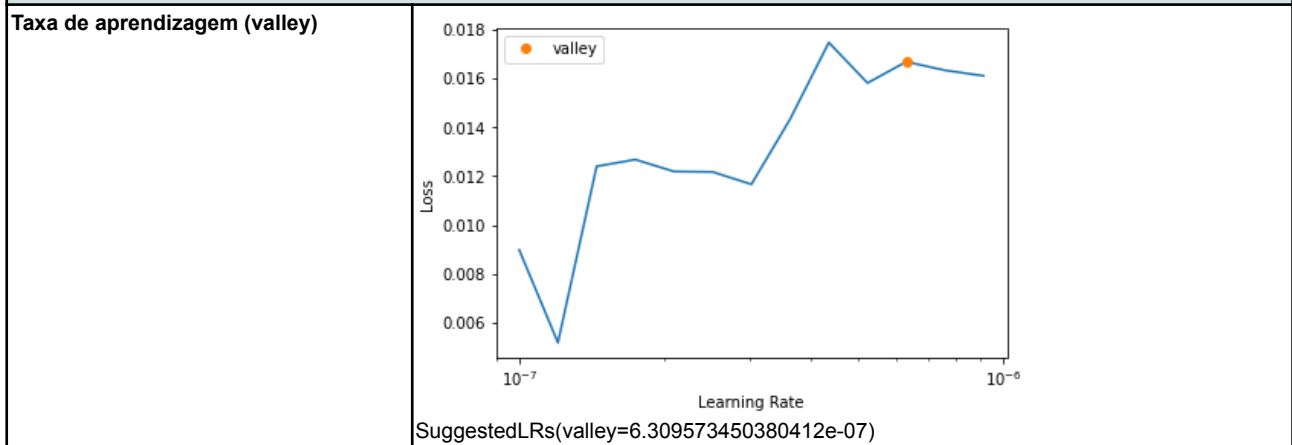
**F1 score** 0.96

**Matriz de confusão**






























Top 9 losses	Prediction/Actual/Loss/Probability		
	plastic/paper / 6.64 / 1.00 	metal/plastic / 5.73 / 1.00 	plastic/paper / 5.71 / 1.00 
	glass/metal / 4.15 / 0.98 	paper/metal / 3.34 / 0.95 	metal/plastic / 3.22 / 0.91 
	metal/glass / 2.94 / 0.95 	glass/plastic / 2.80 / 0.94 	plastic/paper / 2.60 / 0.92 

**Treinamento - Fine Tuning**



**Avaliação - Fine Tuning**

<b>Acurácia total</b>	0,969	
<b>Acurácia por categoria</b>	papel	0,98
	plástico	0,94
	metal	0,97
	vidro	0,97
<b>Precisão</b>	0,97	
<b>Recall</b>	0,97	
<b>F1 score</b>	0,97	

<b>Matriz de confusão</b>	<p style="text-align: center;"><b>Confusion matrix</b></p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td style="border: none;"></td> <td style="border: none;">glass</td> <td style="border: none;">metal</td> <td style="border: none;">paper</td> <td style="border: none;">plastic</td> </tr> <tr> <td style="border: none; vertical-align: middle;">Actual</td> <td style="background-color: #0056b3; color: white;">117</td> <td style="background-color: #0099cc;">1</td> <td style="background-color: #0099cc;">1</td> <td style="background-color: #0099cc;">1</td> </tr> <tr> <td style="border: none; vertical-align: middle;">metal</td> <td style="background-color: #0099cc;">1</td> <td style="background-color: #0070c0;">79</td> <td style="background-color: #0099cc;">1</td> <td style="background-color: #0099cc;">0</td> </tr> <tr> <td style="border: none; vertical-align: middle;">paper</td> <td style="background-color: #0099cc;">0</td> <td style="background-color: #0099cc;">0</td> <td style="background-color: #0056b3; color: white;">132</td> <td style="background-color: #0099cc;">2</td> </tr> <tr> <td style="border: none; vertical-align: middle;">plastic</td> <td style="background-color: #0099cc;">2</td> <td style="background-color: #0099cc;">3</td> <td style="background-color: #0099cc;">2</td> <td style="background-color: #0056b3; color: white;">109</td> </tr> <tr> <td style="border: none;"></td> <td style="border: none; text-align: center;">glass</td> <td style="border: none; text-align: center;">metal</td> <td style="border: none; text-align: center;">paper</td> <td style="border: none; text-align: center;">plastic</td> </tr> <tr> <td style="border: none;"></td> <td colspan="4" style="border: none; text-align: center;">Predicted</td> </tr> </table>		glass	metal	paper	plastic	Actual	117	1	1	1	metal	1	79	1	0	paper	0	0	132	2	plastic	2	3	2	109		glass	metal	paper	plastic		Predicted			
	glass	metal	paper	plastic																																
Actual	117	1	1	1																																
metal	1	79	1	0																																
paper	0	0	132	2																																
plastic	2	3	2	109																																
	glass	metal	paper	plastic																																
	Predicted																																			
<b>Top 9 losses</b>	<p style="text-align: center;"><b>Prediction/Actual/Loss/Probability</b></p> <table style="width: 100%; text-align: center;"> <tr> <td style="width: 33%;">           plastic/paper / 6.36 / 1.00   </td> <td style="width: 33%;">           metal/plastic / 4.75 / 0.99   </td> <td style="width: 33%;">           paper/metal / 4.42 / 0.98   </td> </tr> <tr> <td>           glass/metal / 3.79 / 0.98   </td> <td>           metal/plastic / 3.70 / 0.91   </td> <td>           plastic/paper / 3.40 / 0.97   </td> </tr> <tr> <td>           metal/glass / 2.81 / 0.94   </td> <td>           metal/plastic / 2.77 / 0.75   </td> <td>           glass/plastic / 2.76 / 0.92   </td> </tr> </table>	plastic/paper / 6.36 / 1.00 	metal/plastic / 4.75 / 0.99 	paper/metal / 4.42 / 0.98 	glass/metal / 3.79 / 0.98 	metal/plastic / 3.70 / 0.91 	plastic/paper / 3.40 / 0.97 	metal/glass / 2.81 / 0.94 	metal/plastic / 2.77 / 0.75 	glass/plastic / 2.76 / 0.92 																										
plastic/paper / 6.36 / 1.00 	metal/plastic / 4.75 / 0.99 	paper/metal / 4.42 / 0.98 																																		
glass/metal / 3.79 / 0.98 	metal/plastic / 3.70 / 0.91 	plastic/paper / 3.40 / 0.97 																																		
metal/glass / 2.81 / 0.94 	metal/plastic / 2.77 / 0.75 	glass/plastic / 2.76 / 0.92 																																		
<b>Predição</b>																																				
<b>Quantidade de testes realizadas</b> (via upload de imagens novos)	5																																			
<b>Acurácia dos testes</b>	75%																																			
<b>Limitações e considerações éticas</b>																																				
<b>Limitações</b>	Esse modelo é limitado a somente as 4 categorias de lixo reciclável, e portanto, é incapaz de analisar objetos não recicláveis, como lixos orgânicos ou hospitalares.																																			
<b>Considerações éticas referente aos dados</b>	Não são incluídas imagens com conteúdo não apropriado para menores de 18 anos.																																			
<b>Possíveis riscos e danos</b>	Elementos como formato e rótulo dos objetos podem variar dependendo da região, visto que existem muitos produtos que são apenas disponíveis em determinados países, regiões, estados ou cidades. Tal fato pode afetar a identificação de novos dados.																																			
<b>Deployment</b>																																				
<b>Formato da exportação do modelo</b>	ONNX																																			
<b>Referências</b>																																				
<b>Autores e afiliação</b>	C. Gresse von Wangenheim, Marcos Silva Laydner INCoD/INE/UFSC																																			

Licença	-
Conjunto de dados (link da página do site GQS)	<a href="http://softwarequalitygroup.paginas.ufsc.br/datasets/recyclable-trash-dataset/">http://softwarequalitygroup.paginas.ufsc.br/datasets/recyclable-trash-dataset/</a>
Modelo (link no GitHub da UFSC)	<a href="https://codigos.ufsc.br/100000000394729/automacao-avaliacao-ensino-basico/-/blob/master/Classificador_de_lixo_reciclavel.ipynb">https://codigos.ufsc.br/100000000394729/automacao-avaliacao-ensino-basico/-/blob/master/Classificador_de_lixo_reciclavel.ipynb</a>
Publicações (referências de publicações)	-

**C4 - Treinamento - Transfer Learning (OA6).** O C4 é avaliado por meio do arquivo manifest.json, analisando os parâmetros de *learning rate*, *epochs* e *batch size*, e comparando-os com os valores padrões do *Google Teachable Machine*. Caso todos os valores forem alterados, o aluno recebe 2 pontos, ou 1 ponto se pelo menos um dos atributos foi modificado. Se todos os parâmetros estiverem no valor padrão, a nota é 0.

```

avalia_parametros():

dicionário de parâmetros são carregados do json

se epocas == 50 e tamanho do batch == 16 e taxa de aprendizado == 0.001 ->
retorna 1
senão retorna 2

```

**C5 e C6 - Análise de acurácia por categoria e Interpretação de acurácia (OA7).** São coletadas as informações referentes a este critério a partir do formulário do passo 3 (Tabela 28, linha 7). O aluno informa os valores de acurácia geral do modelo, assim como de cada categoria. Além disso, o aluno aponta as categorias que estão abaixo de 90% de acurácia, e se o modelo deve ser melhorado ou não. O primeiro critério verifica se as classes que possuem baixa acurácia foram apontadas corretamente pelo aluno, comparando ambas respostas. O segundo verifica a conclusão do desempenho do modelo, analisando a resposta da última questão, e considerando as respostas anteriores. O aluno recebe 2 pontos se corretamente identificar que se nenhuma das classes tem acurácia menor que 90%, o modelo está funcionando adequadamente. Alternativamente, o aluno deve demonstrar que se o modelo apresenta classes com acurácias baixas, o mesmo precisa ser melhorado. Caso contrário, são atribuídos 0 pontos.

```

avalia_acuracia():

para cada classe com acurácia menor que 0.9:

```

```
se o aluno a indicou como menor que 0.9 -> somar ao total de avaliações  
com baixo desempenho corretas
```

```
se número de classes de baixo desempenho indicadas corretamente = número de  
classes de baixo desempenho real -> c5 = 2
```

```
se número de classes de baixo desempenho indicadas corretamente > 0 -> c5 = 1  
senão -> c5 = 0
```

```
se classes abaixo de 0.9 == 0 e aluno indicou "o modelo está funcionando  
perfeitamente" -> c6 = 2
```

```
se classes abaixo de 0.9 > 0 e aluno indicou "o modelo precisa ser melhorado"  
-> c6 = 2
```

```
senão -> c6 = 0
```

```
retorna c5 e c6
```

**C7 e C8 - Análise da matriz de confusão e Interpretação da matriz de confusão (OA7).** As informações sobre a matriz de confusão foram extraídas a partir das informações coletadas por meio do formulário online do passo 4 (Tabela 28, linha 9). O primeiro critério envolve a identificação dos erros de classificação baseado na matriz de confusão. As respostas do aluno sobre quais classes são classificadas incorretamente é comparada com a matriz de confusão indicada pelo aluno. A pontuação do item é alocada de acordo com a rubrica. O segundo critério avalia a interpretação do aluno sobre o que a matriz de confusão diz sobre seu modelo. O aluno recebe 2 pontos se corretamente identificar que se nenhuma das classes possui classificações incorretas, o modelo está funcionando adequadamente. Alternativamente, o aluno deve demonstrar que se o modelo apresenta classes com erros de classificação, o mesmo precisa ser melhorado. Caso contrário, são atribuídos 0 pontos.

```
avalia_matriz_de_confusão():
```

```
para cada linha da matriz:
```

```
    para cada coluna da matriz:
```

```
        se número da coluna != número da linha:
```

```
            se valor da célula != 0 -> matriz de confusão não perfeita
```

```
            salvar classe confundida
```

```
para cada classe x:
```

```
    para cada classe y confundida com classe x:
```

```
        se aluno indicou como confundida -> continua
```

```
        senão -> soma ao número de erros
```

```
se número de erros == 0 -> c7 = 2
se número de erros == 1 -> c7 = 1
senão -> c7 = 0
```

```
se matriz de confusão perfeita e aluno indicou "o modelo está funcionando
perfeitamente" -> c8 = 2
se matriz de confusão não perfeita e aluno indicou "o modelo precisa ser
melhorado" -> c8 = 2
senão -> c8 = 0
```

**C9 - Ajustes/melhorias feitas (OA6).** Este item é automatizado a partir dos dados originados do **formulário do passo 5**. Analisa-se a resposta do aluno sobre novas tentativas de melhorar o modelo. Se o aluno indica uma tentativa, ele recebe 1 ponto, enquanto para duas ou mais, são dados 2 pontos e 0 pontos caso o aluno responda que não tentou melhorar o modelo.

```
avalia_ajustes():

se aluno indicou "sim, várias vezes" -> return 2
se aluno indicou "sim, uma vez" -> return 1
senão -> return 0
```

**C10 e C11 - Testes com novos objetos e Análise dos resultados de testes (OA7).** A automação dos itens C10 e C11 são feitas por meio da análise dos resultados do formulário online do passo 2 (Tabela 28, linha 5). O primeiro critério de C10 checa a resposta do aluno sobre a quantidade de novos testes realizados, atribuindo a pontuação de acordo com a rubrica. O segundo critério verifica se a indicação dada pelo aluno sobre a quantidade de erros nos testes é condizente com os resultados apontados no formulário. A pontuação

**C12 - Interpretação dos testes (OA7).** o aluno recebe 2 pontos se corretamente identificar que se seu modelo não errou nenhum teste, está funcionando adequadamente, ou que se errou um ou mais testes, o modelo precisa ser melhorado. Caso contrário, nenhum ponto é atribuído a esse item. Este é analisado comparando o número de erros dos testes e a resposta do aluno no formulário sobre o desempenho do seu modelo.

```

avalia_testes():

se número de testes realizados > 2 -> c11 = 2
se número de testes realizados >= 1 -> c11 = 1
senão -> c10 = 0

para cada teste:
    se teste correto -> soma ao valor real de testes corretos

se valor real de testes corretos = valor de testes corretos indicados pelo
aluno -> c11 = 2
senão -> c11 = 0

se quantidade de testes corretos = quantidades de testes realizados e aluno
indicou "o modelo está funcionando perfeitamente" -> c12 = 2
se quantidade de testes corretos != quantidades de testes realizados e aluno
indicou "o modelo precisa ser melhorado" -> c12 = 2
senão -> c12 = 0

retorna c10, c11, 12

```

### 5.3 AVALIAÇÃO PRELIMINAR DO MODELO DE AVALIAÇÃO

Para executar uma avaliação preliminar da correção do modelo de avaliação, foram criados três cenários apresentando resultados de aprendizagem em 3 níveis: baixo, médio e alto desempenho, conforme apresentado nas Tabelas 22-24. Analisando estes exemplos de resultados com a automação da rubrica de forma intermediária por meio de questionários desenvolvidos no Limesurvey (LIMESURVEY, 2022), foram calculadas as pontuações e notas.

Tabela 22 - Cenário de resultados de aprendizagem de baixo desempenho

	Pontuação
Pontuação total nos quizzes	10

**Artefatos criados no TM**

Papel ⋮

---

1 Image Samples

Webcam

Upload

Plástico ⋮

---

1 Image Samples

Webcam

Upload

Metal ⋮

---

1 Image Samples

Webcam

Upload

Vidro ⋮

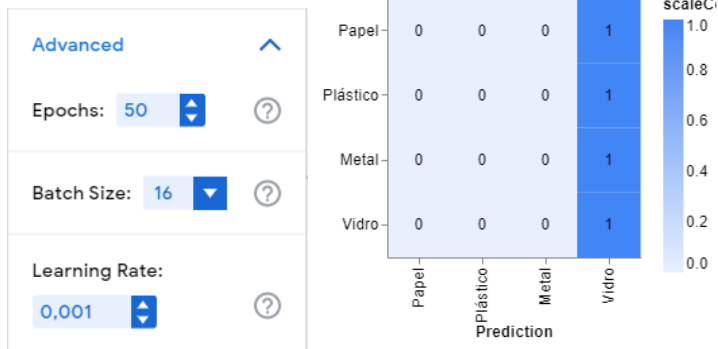
---

5 Image Samples

Webcam

Upload

Confusion Matrix



**Accuracy per class**

CLASS	ACCURACY	# SAMPLES
Papel	0.00	1
Plástico	0.00	1
Metal	0.00	1
Vidro	1.00	1



Interpretações	Categoria de lixo do objeto				Categoria indicada pelo modelo			
	Metal	Papel	Plástico	Vidro	Metal	Papel	Plástico	Vidro
	Objeto novo 1	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Objeto novo 2	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Objeto novo 3	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Objeto novo 4	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Objeto novo 5	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
<p>Analisando a tabela de acurácia, você pode observar, que</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> todas as categorias estão sendo reconhecidas com uma acurácia acima de 90%</p> <p><input type="radio"/> alguma(s) categorias são reconhecidas com acurácia abaixo de 90%, quais?</p>								
<p>Isto indica o que?</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado, como?</p>								
<p>Analisando a matriz de confusão, você pode observar, que:</p> <p>Comentar apenas quando você selecionar uma resposta.</p> <p><input checked="" type="checkbox"/> todas as imagens do conjunto de teste estão sendo classificadas corretamente</p> <p><input type="checkbox"/> alguma(s) das imagens de metal são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de papel são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de plástico são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de vidro são erradamente classificadas como</p>								
<p>Isto indica o que?</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado, como?</p> <p style="text-align: right;">Por favor,</p>								
<b>Pontuação total</b>								13
<b>Nota</b>								1,5

Tabela 23 - Cenário de resultados de aprendizagem de médio desempenho

		Pontuação
<b>Pontuação total nos quizzes</b>		30
<b>Artefatos criados no TM</b>		20

**Papel** ✎

8 Image Samples

Webcam Upload

---

**Plástico** ✎

6 Image Samples

Webcam Upload

---

**Metal** ✎

6 Image Samples

Webcam Upload

---

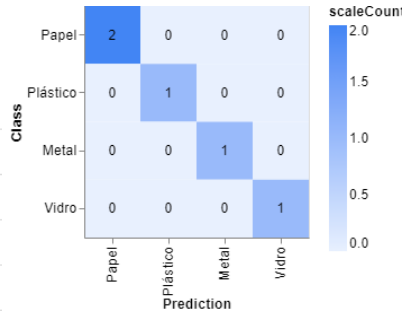
**Vidro** ✎

6 Image Samples

Webcam Upload

**Accuracy per class**

CLASS	ACCURACY	# SAMPLES
Papel	1.00	2
Plástico	1.00	1
Metal	1.00	1
Vidro	1.00	1



**Advanced** ^

Epochs: 50 ?

Batch Size: 32 ?

Learning Rate: 0,001 ?

<p><b>Interpretações</b></p>	<p>Quantas vezes o modelo acertou?</p> <p>Escolha uma das seguintes respostas:</p> <p>2</p> <hr/> <p>Estes resultados indicam que?</p> <p>Escolha uma das seguintes respostas:</p> <p><input type="radio"/> o modelo está funcionando perfeitamente</p> <p><input checked="" type="radio"/> o modelo precisa ser melhorado</p> <hr/> <p>Analisando a tabela de acurácia, você pode observar, que</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> todas as categorias estão sendo reconhecidas com uma acurácia acima de 90%</p> <p><input type="radio"/> alguma(s) categorias são reconhecidas com acurácia abaixo de 90%, quais?</p> <hr/> <p>Isto indica o que?</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado, como?</p> <hr/> <p>Analisando a matriz de confusão, você pode observar, que:</p> <p>Comentar apenas quando você selecionar uma resposta.</p> <p><input checked="" type="checkbox"/> todas as imagens do conjunto de teste estão sendo classificadas corretamente</p> <p><input type="checkbox"/> alguma(s) das imagens de metal são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de papel são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de plástico são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de vidro são erradamente classificadas como</p> <hr/> <p>Isto indica o que?</p> <p>Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado, como?</p> <p>Por favor,</p>	
<p><b>Pontuação total</b></p>		<p>50</p>
<p><b>Nota</b></p>		<p>6,3</p>

Tabela 24. Cenário de resultados de aprendizagem de alto desempenho

	<p><b>Pontuação</b></p>
<p><b>Pontuação total nos quizzes</b></p>	<p>60</p>

**Artefatos criados no TM**

Papel ⋮

13 Image Samples

Webcam

Upload

Plástico ⋮

12 Image Samples

Webcam

Upload

Metal ⋮

11 Image Samples

Webcam

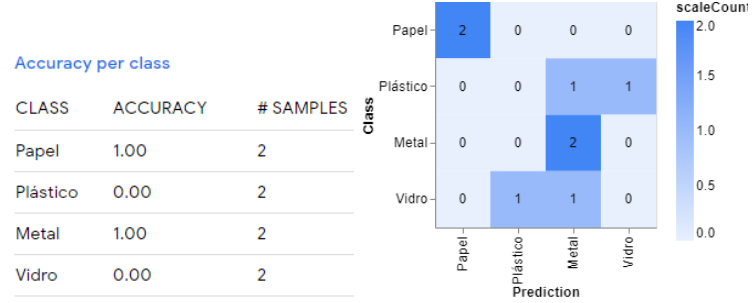
Upload

Vidro ⋮

12 Image Samples

Webcam

Upload



**Advanced** ^

Epochs:  ↕ ?

Batch Size:  ▼ ?

Learning Rate:  ↕ ?

<p><b>Interpretações</b></p>	<p>Quantas vezes o modelo acertou?</p> <p>! Escolha uma das seguintes respostas:</p> <p>5</p> <hr/> <p>Estes resultados indicam que?</p> <p>! Escolha uma das seguintes respostas:</p> <p><input checked="" type="radio"/> o modelo está funcionando perfeitamente</p> <p><input type="radio"/> o modelo precisa ser melhorado</p> <hr/> <p>Analisando a tabela de acurácia, você pode observar, que</p> <p>! Escolha uma das seguintes respostas:</p> <p><input type="radio"/> todas as categorias estão sendo reconhecidas com uma acurácia acima de 90%</p> <p><input checked="" type="radio"/> alguma(s) categorias são reconhecidas com acurácia abaixo de 90%, quais?</p> <p>Por favor, coloque aqui o seu comentário:</p> <p>plástico, vidro</p> <hr/> <p>Isto indica o que?</p> <p>! Escolha uma das seguintes respostas:</p> <p><input type="radio"/> o modelo está funcionando perfeitamente</p> <p><input checked="" type="radio"/> o modelo precisa ser melhorado, como?</p> <p>Por favor, coloque aqui o seu comentário:</p> <p>adicionando mais imagens ao treinamento</p> <hr/> <p>Analisando a matriz de confusão, você pode observar, que:</p> <p>! Comentar apenas quando você selecionar uma resposta.</p> <p><input type="checkbox"/> todas as imagens do conjunto de teste estão sendo classificadas corretamente</p> <p><input type="checkbox"/> alguma(s) das imagens de metal são erradamente classificadas como</p> <p><input type="checkbox"/> alguma(s) das imagens de papel são erradamente classificadas como</p> <p><input checked="" type="checkbox"/> alguma(s) das imagens de plástico são erradamente classificadas como</p> <p><input checked="" type="checkbox"/> alguma(s) das imagens de vidro são erradamente classificadas como</p> <p>metal, vidro</p> <p>metal, plástico</p> <hr/> <p>Isto indica o que?</p> <p>! Escolha uma das seguintes respostas:</p> <p><input type="radio"/> o modelo está funcionando perfeitamente</p> <p><input checked="" type="radio"/> o modelo precisa ser melhorado, como?</p> <p>Por favor, coloque aqui o seu comentário:</p> <p>incluir mais imagens no teste</p>	
<p><b>Pontuação total</b></p>		<p>86</p>
<p><b>Nota</b></p>		<p>10</p>

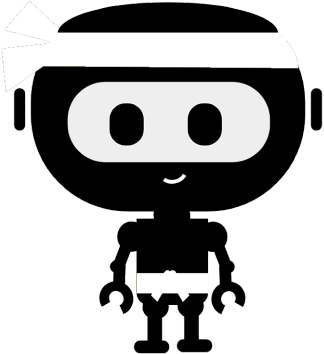
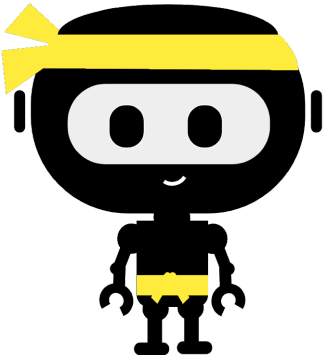
A partir da realização desses testes, observa-se que as pontuações e notas correspondem com os níveis de aprendizagem representados em cada um dos três cenários apresentados.

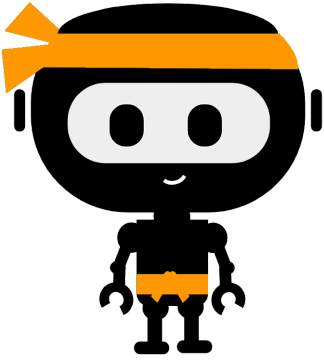
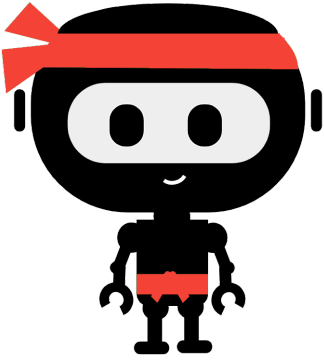
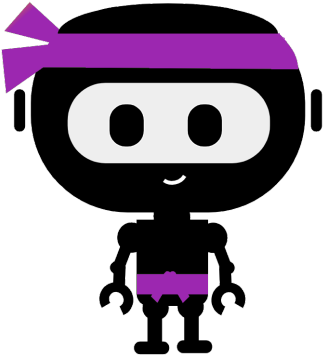
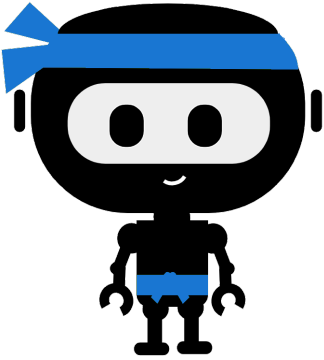
## 6. INTEGRAÇÃO COM O CODEMASTER

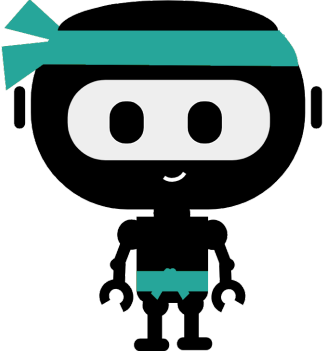
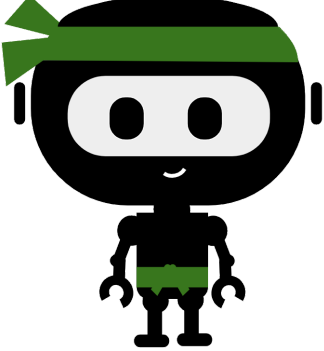
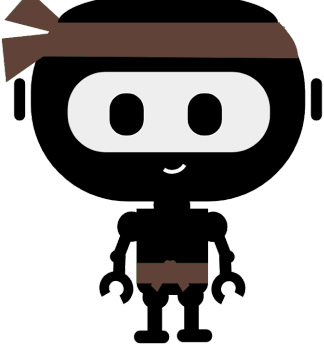
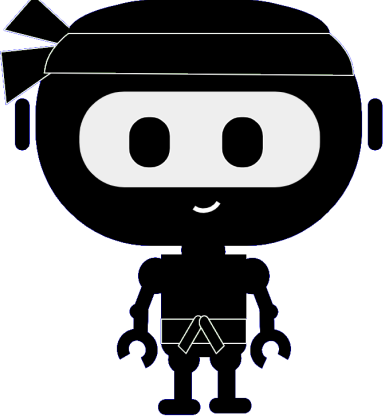
Com a rubrica definida e automatizada, resta apenas integrar as ferramentas desenvolvidas com o CodeMaster (DEMETRIO, 2017), para que os alunos possam responder os formulários e obterem suas notas de forma autônoma, com a vantagem de ter um feedback com elementos lúdicos, na forma do ninja robô, a versão do mascote do CodeMaster - App Inventor com o tema de *ML*. Dessa forma, foi desenvolvida uma extensão do CodeMaster, para agora ter também, a avaliação do curso Machine Learning para Todos! (GRESSE VON WANGENHEIM et al., 2020) .

O cálculo da nota final, combinando a nota da rubrica com a dos quizzes, já apresentado anteriormente, foi integrado com a representação do CodeMaster de cores de faixas diferentes para o ninja robô, mapeado conforme a tabela 25, com o objetivo de tornar essa nota mais divertida e interessante aos alunos. As imagens dos respectivos ninjas robôs, criados por M. Nathalie Ferreira.

Tabela 25 - Mapeamento da nota e a cor da faixa do ninja robô

Nota	Cor da Faixa	Imagem
0 - 0.9	Branca	
1.0 – 1.9	Amarela	

2.0 – 2.9	Laranja	
3.0 – 3.9	Vermelha	
4.0 – 4.9	Roxa	
5.0 – 5.9	Azul	

6.0 – 6.9	Turquesa	
7.0 – 7.9	Verde	
8.0 – 8.9	Marrom	
9.0 - 10.0	Preta	



## 6.1 ANÁLISE DE REQUISITOS

Com base nos conceitos desenvolvidos anteriormente no trabalho, foram identificados e analisados os requisitos da extensão da plataforma, além de elaborados os fluxos de execução do curso.

### 6.1.1 Requisitos Funcionais

Os requisitos funcionais, apresentados na Tabela 26, foram elaborados com base nos requisitos funcionais do CodeMaster original (DEMETRIO, 2017), e nos formulários anteriormente feitos no Limesurvey (LIMESURVEY, 2022).

Tabela 26 - Requisitos Funcionais

ID	Requisito	Descrição	Artefato Entrada	Artefato Saída
RF01	Realizar <i>Upload</i> de um projeto do <i>Google Teachable Machine</i>	A ferramenta deve permitir que qualquer usuário envie um projeto <i>Google Teachable Machine</i> no formato <i>.tm</i> .	Arquivo <i>.tm</i> exportado do <i>Google Teachable Machine</i> , selecionado no computador do usuário	Projeto <i>Google Teachable Machine</i> em uma matriz de bytes.
RF02	Descompactar Projeto <i>Google Teachable Machine</i>	A ferramenta deve ser capaz de descompactar o projeto <i>Google Teachable Machine</i> no formato <i>.tm</i> e localizar os arquivos de imagens e o <i>manifest.json</i>	Projeto <i>Google Teachable Machine</i> formato <i>.tm</i>	Lista das imagens utilizadas no projeto e <i>manifest.json</i> do mesmo
RF03	Avaliar o projeto <i>Google Teachable Machine</i>	A ferramenta deve avaliar o projeto enviado baseado na rubrica desenvolvida	Lista das imagens utilizadas no projeto e <i>manifest.json</i> do mesmo	Pontuação dos critérios relacionados ao projeto
RF04	Receber respostas por meio do <i>front-end</i>	A ferramenta deve ser capaz de capturar as respostas dos formulários, e enviá-las para a avaliação	Respostas dos formulários	Listas de strings com cada campo de resposta
RF05	Avaliar respostas dos formulários	A ferramenta deve ser capaz de avaliar as respostas dos formulários com base na rubrica desenvolvida	Listas de strings com cada campo de resposta	Pontuação dos critérios relacionados ao respectivo formulário
RF06	Manter o registro das pontuações dos critérios do aluno	A ferramenta deve manter em seu banco de dados os registros das pontuações de cada critério feitas por um aluno, cuja identificação é criptografada	Pontuação do aluno em cada critério	O sistema armazena um Hash do email do aluno, junto com as suas pontuações em cada critério
RF07	Calcular a nota final	A ferramenta deve ser capaz de reunir a pontuação de cada critério armazenado no banco e calcular a nota final do aluno	Pontuação do aluno em cada critério	Nota final do aluno

<b>RF08</b>	Apresentar a pontuação e nota final	A ferramenta deve apresentar a nota final e a cor da faixa do ninja robô, mostrando a pontuação de cada critério	Pontuação do aluno em cada critério e nota final	Interface com usuário apresenta informação quantitativa de cada elemento analisado, a nota final e a faixa do ninja robô
-------------	-------------------------------------	--	--	--

### 6.1.2 Requisitos Não-funcionais

Os requisitos não-funcionais, apresentados na Tabela 27, foram elaborados com base nos requisitos não-funcionais do CodeMaster original (DEMETRIO, 2017).

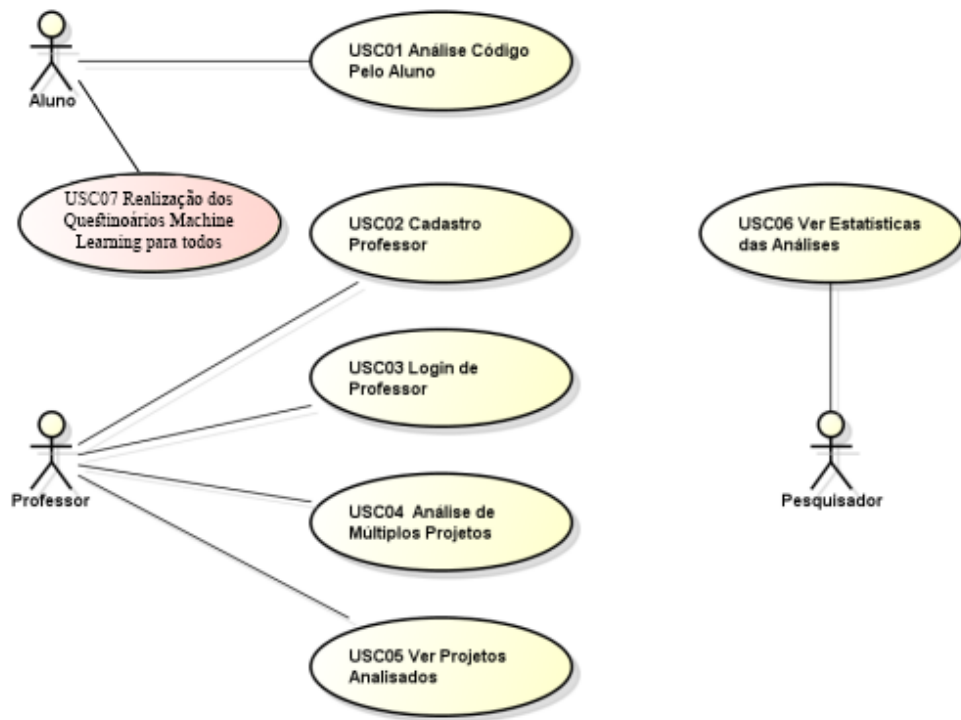
Tabela 27 - Requisitos Não-Funcionais

ID	Requisito	Descrição
<b>RNF01</b>	Linguagem de Programação Java para o <i>back-end</i>	O <i>back-end</i> da ferramenta deve ser desenvolvida na linguagem de programação Java, pois é uma linguagem muito conhecida facilitando a manutenção do sistema por profissionais qualificados.
<b>RNF02</b>	Linguagem de Programação Typescript com Angular para <i>front-end</i>	O <i>front-end</i> da ferramenta deve ser desenvolvida na linguagem de programação Typescript, utilizando o framework Angular, uma combinação bastante utilizada atualmente, que facilita e agiliza o desenvolvimento <i>web</i>
<b>RNF03</b>	Linguagem de Programação Python para avaliação	A avaliação deve ser implementada em python, pois ela facilita o desenvolvimento envolvendo modelos de <i>machine learning</i>
<b>RNF04</b>	Comunicação REST	As partes do sistema (front, back e avaliações) deve se comunicar por meio de APIs REST
<b>RNF05</b>	Criptografia não reversível do e-mail dos alunos	Para atrelar a pontuação ao aluno, o e-mail do mesmo é usado para identificá-lo. Para manter a privacidade e segurança dos dados do mesmo, o e-mail deve passar por um <i>hash</i> para ser salvo no banco
<b>RNF06</b>	Projeto <i>Google Teachable Machine</i>	O sistema deve analisar projetos <i>Google Teachable Machine</i> no formato <i>.tm</i>
<b>RNF07</b>	Identidade Visual da CnE	A interface de usuário deve ter padrões de identidade visual definidos pela Iniciativa Computação na Escola
<b>RNF08</b>	Sistema Web	A ferramenta deve ser acessada via navegador Web com conexão à internet

## 6.2 CASO DE USO

Analisando os requisitos elaborados, foi necessário desenvolver um caso de uso a mais para a plataforma, sendo adicionado aos já criados e detalhados em Demetrio (2017).

Figura 8 - Diagrama de casos de Uso do CodeMaster, adaptado de Demetrio (2017)



## Caso de Uso 7 - USC07 Realização dos Questionários Machine Learning para Todos

### Atores:

- Aluno

### Fluxo principal:

1. Aluno acessa o site do CodeMaster
2. Aluno clica no item de menu “Machine Learning para Todos”.
3. Aluno clica no item da interface “Parte 1: Arquivo .tm”
4. Aluno preenche o formulário e clica em enviar
5. Aluno clica no item da interface “Parte 2: Testes com novos objetos”
6. Aluno preenche o formulário e clica em confirmar
7. Aluno clica no item da interface “Parte 3: Acurácia”
8. Aluno preenche o formulário e clica em confirmar
9. Aluno clica no item da interface “Parte 4: Matriz de Confusão”
10. Aluno preenche o formulário e clica em confirmar
11. Aluno clica no item da interface “Parte 5: Melhorias do modelo”
12. Aluno preenche o formulário e clica em confirmar
13. Aluno clica no item da interface “Parte 6: Avaliação”
14. Sistema retorna o resultado final da avaliação da rubrica

### Fluxo Alternativo:

Aluno não preenche um dos campos corretamente e o sistema apresenta a mensagem: “Todos os campos devem ser preenchido corretamente”

## Fluxo de Exceção:

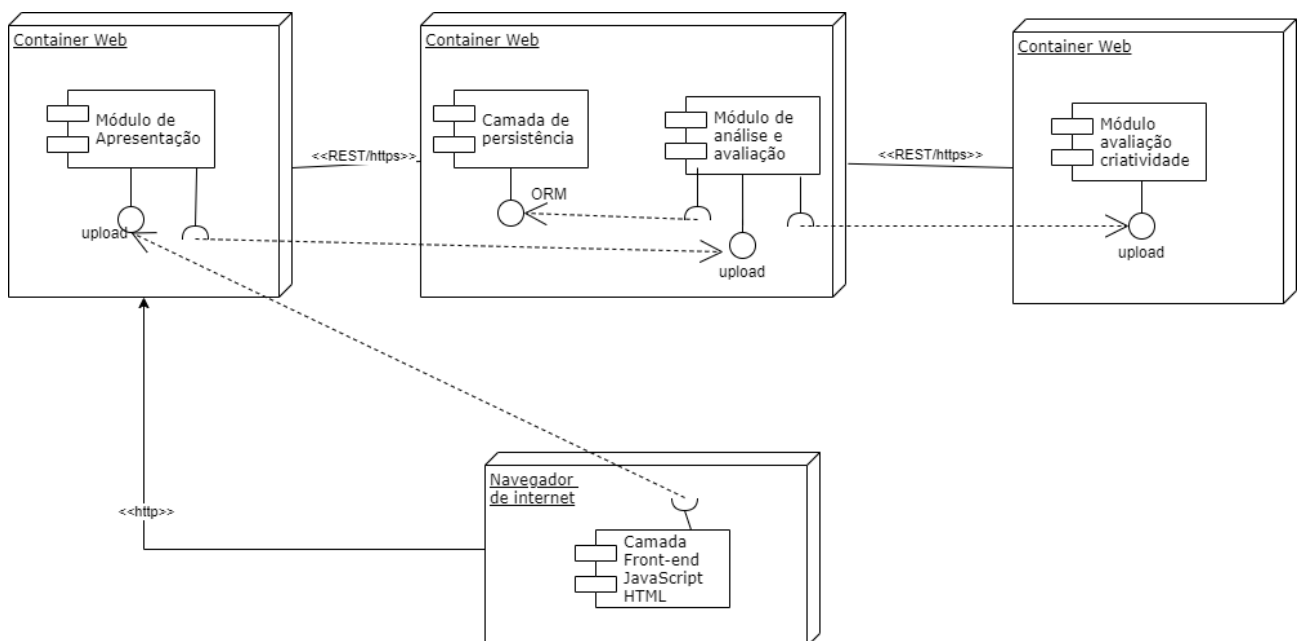
**FE01** - Aluno envia um arquivo que não é um projeto *Google Teachable Machine*. Sistema informa uma mensagem: “O arquivo deve ser um projeto *Google Teachable Machine* no formato *.tm*”

**FE02** - Aluno tenta acessar a parte 6 antes de preencher as etapas anteriores. Sistema informa uma mensagem: “Todas as partes devem ser completas para poder ver seu resultado”

## 6.3 MODELAGEM E IMPLEMENTAÇÃO

O sistema foi expandido a partir de uma versão atualizada de Demetrio (2017), conforme apresentado na Figura 9. O modelo de arquitetura do sistema que objetiva a separação das camadas de apresentação e análise em diferentes módulos foi mantido, com uma maior separação do módulo de apresentação, com a migração JSP para um container separado utilizando Typescript utilizando Angular como framework de desenvolvimento web.

Figura 9 - Diagrama de Componentes do CodeMaster, baseado em Demetrio (2017)



O módulo de análise a avaliação é um serviço web REST que, dentro do contexto da implementação do Machine Learning para Todos, é responsável por receber as respostas do aluno, encaminhá-las para o módulo de avaliação e criatividade para serem avaliadas, receber as notas dos critérios, e persisti-las no banco de dados. REST (RODRIGUEZ, 2008) é uma arquitetura que permite a separação de sistemas web em

módulos (serviços). Para implementar esse serviço o CodeMaster V2.0 utiliza o *framework* Spring (SPRING, 2021), que abstrai os detalhes de baixo nível da implementação de comunicação entre os servidores e simplifica a implementação de um serviço REST, tornando simples a implementação, implantação e manutenção de serviços *web stand-alone*.

O módulo de apresentação é o responsável pelo controle da interface de usuário, e no contexto deste trabalho, envia as respostas e artefatos do aluno para o módulo de análise, além de apresentar o resultado final para o usuário. O componente de *front-end* utiliza as tecnologias Angular (ANGULAR, 2021) e Node.js (NODEJS, 2021) para facilitar o seu desenvolvimento como uma aplicação *web stand-alone*, e simplificar a comunicação com os outros servidores.

O módulo de avaliação e criatividade, neste trabalho, recebe as respostas do aluno, as avalia a partir dos algoritmos definidos na seção 5.2, e retorna a pontuação dos respectivos critérios. Este módulo utiliza Flask (FLASK, 2021), uma biblioteca para Python que, assim como as outras tecnologias usadas nos outros módulos, facilita o desenvolvimento de uma aplicação *web* e na comunicação com os outros módulos REST.

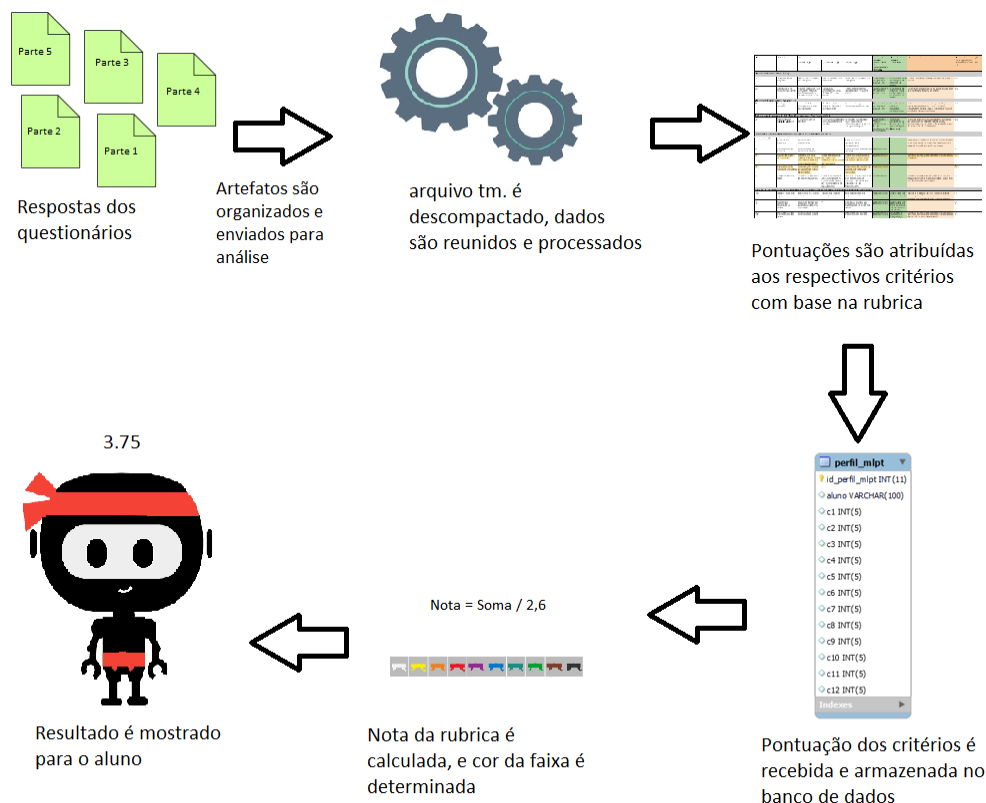


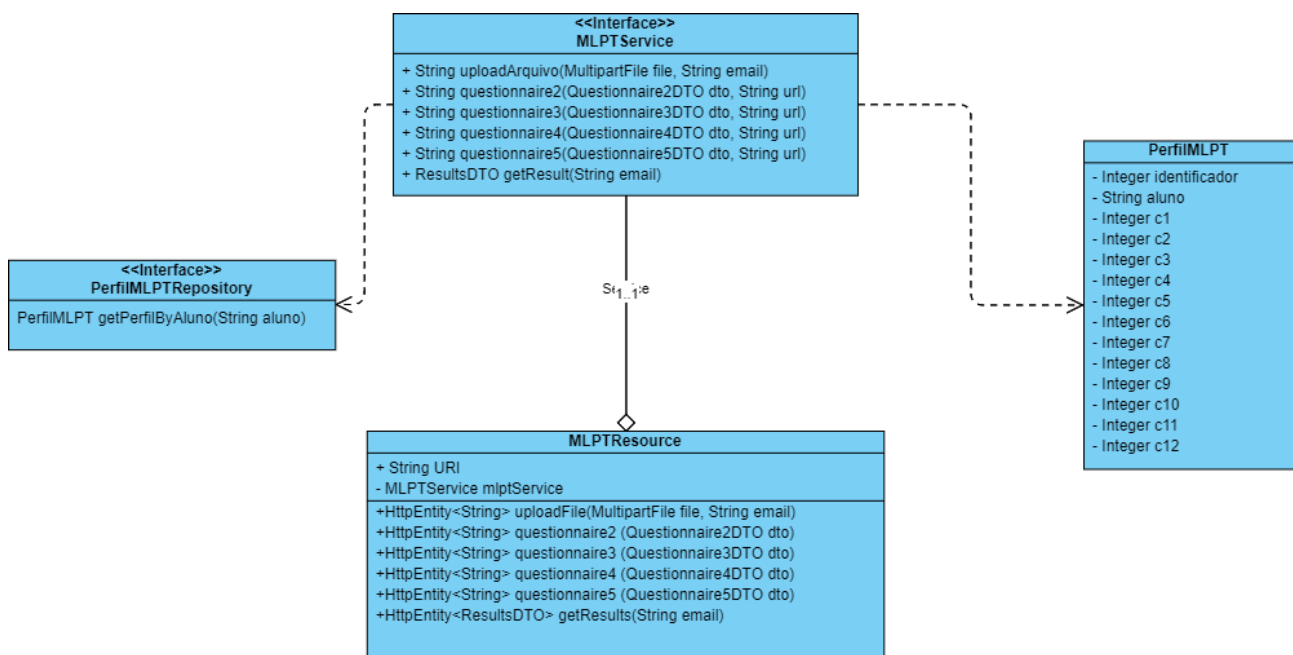
Figura 10 - Sequência da análise da rubrica Machine Learning para Todos

## 6.4 ARQUITETURA DO PROJETO

A arquitetura de cada módulo e a comunicação entre eles é detalhada em (DEMETRIO, 2017), e por isso, para propósito deste projeto, será detalhada apenas a arquitetura das mudanças feitas neste.

A Figura 11 mostra o diagrama das classes desenvolvidas no módulo de análise e avaliação. A classe *MLTPResource* é responsável por receber os dados vindos do módulo de visualização por meio de *endpoints* REST, e passa esses dados para o respectivo método no *MLPTService*. Este por sua vez, recebe o DTO com os dados, e se necessário, os organiza para alguma estrutura de dados como lista ou dicionário, que facilite a avaliação que será realizada. Os dados então são enviados novamente por REST para o módulo de avaliação de criatividade, que responde com as pontuações. Essas pontuações são atribuídas à classe de modelo *PerfilMLPT*, e persistidas com a ajuda do *PerfilMLPTRepository*, que faz a comunicação com a camada de ORM. Finalmente, se o resultado foi salvo no banco e o fluxo finalizou corretamente, é informado ao *frontend* o sucesso da requisição, caso contrário, é informado o erro ocorrido.

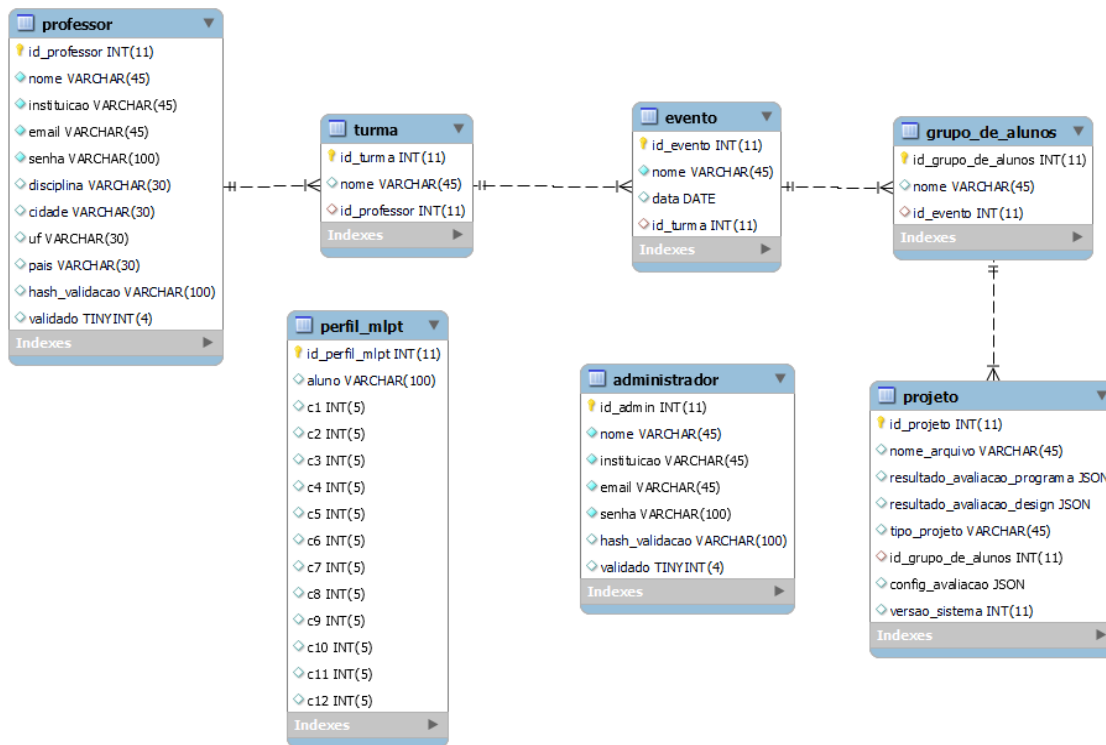
Figura 11 - Diagrama de classes relacionadas à análise dos dados do Machine Learning para Todos



O Diagrama de Entidade Relacionamento da figura 12 consiste no diagrama original com a nova tabela *PerfilMLPT* adicionada neste trabalho. A camada de persistência é utilizada apenas pelo módulo de análise e avaliação. Esta camada utiliza um banco de dados MySQL (MYSQL, 2017). O MySQL permite a criação de bancos de dados relacionais. O banco de dados do CodeMaster foi projetado tendo em mente uma

estrutura simples e objetiva, para permitir futuras implementações no sistema sem afetar os dados já presentes no sistema (DEMETRIO, 2017).

Figura 12 - Diagrama Entidade Relacionamento do Banco de Dados, baseado em (DEMETRIO, 2017)


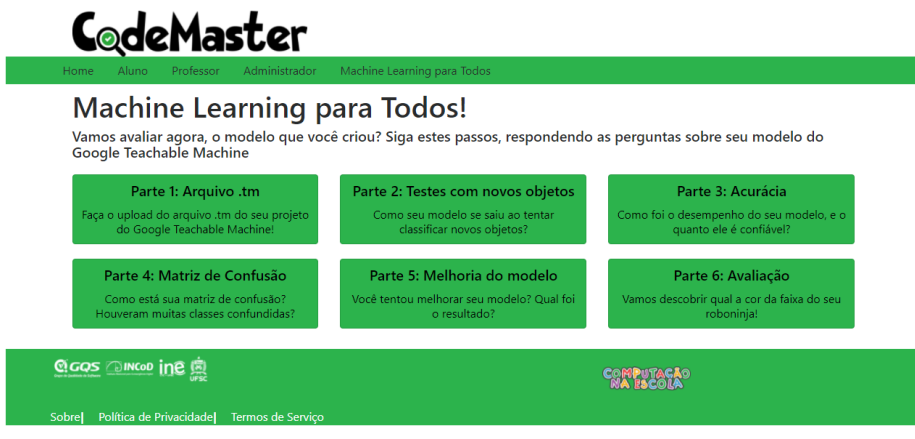



A pontuação do aluno para cada critério é armazenada na nova tabela *PerfilMLPT*, com o e-mail do aluno servindo de identificador para sua pontuação no campo “aluno”. Para uma maior segurança, o e-mail do aluno cadastrado não é salvo em formato original, mas em sim um hash do e-mail, obtido por meio do algoritmo SHA-256, desta forma é possível a partir do e-mail obter o hash, mas a partir do hash é impossível obter o e-mail. Isso impede a descoberta do e-mail dos usuários por pessoas mal intencionadas que possam vir a acessar o banco de dados, e protege a identidade dos alunos.

O módulo de avaliação de criatividade recebe os dados do módulo de análise e avaliação por meio de *endpoints* REST no arquivo *server.py*, que manda os dados para o *evaluator.py*. Este por sua vez avalia os dados de acordo com os algoritmos definidos em 5.2. Quando relevante, o arquivo *.tm* é descompactado pelo *extractor.py* para então ser analisado. A pontuação resultando dos respectivos critérios é então retornada para o módulo de análise e avaliação

## 6.5 DESIGN DE INTERFACE

Tabela 28 - Interfaces gráficas do USC07

<p>1. Aluno acessa o site do CodeMaster</p>	
<p>2. Aluno clica no item de menu "Machine Learning para Todos".</p>	
<p>3. Aluno clica no item da interface "Parte 1: Arquivo .tm"</p>	



4. Aluno preenche o formulário e clica em finalizar

The screenshot shows the CodeMaster website header with navigation links: Home, Aluno, Professor, Administrador, and Machine Learning para Todos. The main heading is "Parte 1: Upload do arquivo .tm". Below it, there is an "E-mail" field containing "teste@teste.com". A list of instructions follows: "1) Clique no botão 'Escolher arquivo' e selecione o arquivo .tm exportado do Google Teachable Machine". A file selection box shows "Choose File" and "teste.tm". A second instruction reads: "2) Clique no botão 'Enviar' e volte para os slides do curso." At the bottom, there is a green "Enviar" button. The footer contains logos for GOS, INCoD, INE, and UFPE, along with the text "COMPUTAÇÃO NA ESCOLA" and links for "Sobre", "Política de Privacidade", and "Termos de Serviço".

5. Aluno clica no item da interface "Parte 2: Testes com novos objetos"

The screenshot shows the CodeMaster website header with navigation links: Home, Aluno, Professor, Administrador, and Machine Learning para Todos. The main heading is "Parte 2: Resultados dos Testes". Below it, there is an "E-mail" field. A list of questions follows: "1) Quantos objetos novos foram testados?" with a dropdown menu; "2) Quais os resultados dos testes realizados?" with a table; "3) Quantos objetos novos o modelo acertou?" with a dropdown menu; and "4) Estes resultados indicam o que?" with a dropdown menu. The table for question 2 has two columns: "Categoria real" and "Categoria indicada pelo seu modelo". A red warning message states: "Todos os campos são obrigatórios e devem ser preenchidos corretamente (email deve ser válido)". The footer contains logos for GOS, INCoD, INE, and UFPE, along with the text "COMPUTAÇÃO NA ESCOLA" and links for "Sobre", "Política de Privacidade", and "Termos de Serviço".

6. Aluno preenche o formulário e clica em finalizar

The screenshot shows the CodeMaster website header with navigation links: Home, Aluno, Professor, Administrador, and Machine Learning para Todos. The main heading is "Parte 2: Resultados dos Testes". Below it, there is an "E-mail" field containing "teste@teste.com". A list of questions follows: "1) Quantos objetos novos foram testados?" with a dropdown menu showing "2"; "2) Quais os resultados dos testes realizados?" with a table; "3) Quantos objetos novos o modelo acertou?" with a dropdown menu showing "1"; and "4) Estes resultados indicam o que?" with a dropdown menu showing "O modelo precisa ser melhorado". The table for question 2 has two columns: "Categoria real" and "Categoria indicada pelo seu modelo". The table contains two rows of data: "Plastico" vs "Metal" and "Vidro" vs "Vidro". A green "Finalizar" button is visible. The footer contains logos for GOS, INCoD, INE, and UFPE, along with the text "COMPUTAÇÃO NA ESCOLA" and links for "Sobre", "Política de Privacidade", and "Termos de Serviço".

7. Aluno clica no item da interface “Parte 3: Acurácia”

# CodeMaster

Home Aluno Professor Administrador Machine Learning para Todos

## Parte 3: Acurácia

E-mail

1) Qual a acurácia do seu modelo? (0 a 1)

**Training**

Model Trained

Advanced

Epochs: 57

Batch Size: 16

Learning Rate: 0.00094

Reset Defaults

Under the hood

Accuracy per epoch

Como encontrar a acurácia

2) Qual a acurácia de cada classe?

**Training**

Model Trained

Advanced

Epochs: 57

Batch Size: 16

Learning Rate: 0.00094

Reset Defaults

Under the hood

Accuracy per class

Calculate accuracy per class

CLASS	ACCURACY	# SAMPLES
Vidro	0.00	2
Metal	0.50	2
Plastico	0.00	2
Papel	0.50	2

Acurácia das classes

Metal

Papel

Plástico

Vidro

3) Analisando os valores de acurácia, você pode observar que

4) Isto indica o que?

Todos os campos são obrigatórios e devem ser preenchidos corretamente (Valores de 0 a 1, email válido)

CCQS INCoD ine UFPE

COMPUTAÇÃO NA ESCOLA

Sobre | Política de Privacidade | Termos de Serviço

8. Aluno preenche o formulário e clica em finalizar

# CodeMaster

Home Aluno Professor Administrador Machine Learning para Todos

## Parte 3: Acurácia

E-mail:

1) Qual a acurácia do seu modelo? (0 a 1)

Training

Model Trained

Advanced

Epochs: 57

Batch Size: 16

Learning Rate: 0.00094

Reset Defaults

Under the hood

Accuracy per epoch

0.0 0.2 0.4 0.6 0.8 1.0

0 10 20 30 40 50 60

Epochs

acc

test acc

x 34

test acc: 0.30000011921

Como encontrar a acurácia

2) Qual a acurácia de cada classe?

Training

Model Trained

Advanced

Epochs: 57

Batch Size: 16

Learning Rate: 0.00094

Reset Defaults

Under the hood

Accuracy per class

Calculate accuracy per class

CLASS	ACCURACY	# SAMPLES
Vidro	0.00	2
Metal	0.50	2
Plástico	0.00	2
Papel	0.50	2

Acurácia das classes

Metal:

Papel:

Plástico:

Vidro:

3) Analisando os valores de acurácia, você pode observar que

4) Isto indica o que?

GGOS INCoD ine UFPA

COMPUTAÇÃO NA ESCOLA

Sobre | Política de Privacidade | Termos de Serviço

9. Aluno clica no item da interface “Parte 4: Matriz de Confusão”

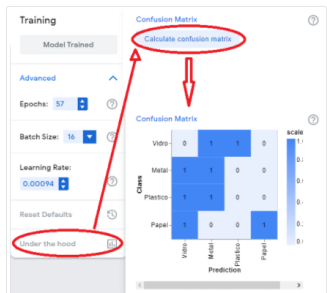
# CodeMaster

Home Aluno Professor Administrador Machine Learning para Todos

## Parte 4: Matriz de Confusão

E-mail

1) Digite sua matriz de confusão:



Como encontrar a matriz de confusão

Categoria	Metal	Papel	Plástico	Vidro
Metal	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Papel	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Plástico	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Vidro	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

2) Analisando a matriz de confusão, você pôde observar que:

- Todas as imagens do conjunto de teste estão sendo classificadas corretamente
- Algumas imagens de metal são erroneamente classificadas como:
- Algumas imagens de papel são erroneamente classificadas como:
- Algumas imagens de plástico são erroneamente classificadas como:
- Algumas imagens de vidro são erroneamente classificadas como:

3) Isto indica o que?

Todos os campos são obrigatórios e devem ser preenchidos corretamente (email deve ser válido, e ser utilizado o mesmo para todas as partes)

GGQS INCoD ine UFPE

COMPUTAÇÃO NA ESCOLA

Sobre | Política de Privacidade | Termos de Serviço

10. Aluno preenche o formulário e clica em confirmar

CodeMaster  
Home Aluno Professor Administrador Machine Learning para Todos

### Parte 4: Matriz de Confusão

E-mail: teste@teste.com

1) Digite sua matriz de confusão:

Training  
Model Trained  
Advanced  
Epochs: 57  
Batch Size: 16  
Learning Rate: 0.00094  
Reset Defaults  
Under the hood

Confusion Matrix  
Calculate confusion matrix

Confusion Matrix

Classes	Vidro	Metal	Papel	Plástico
Vidro	0	1	1	0
Metal	1	3	0	0
Papel	1	1	0	0
Plástico	1	0	0	1

Como encontrar a matriz de confusão

Categoria	Metal	Papel	Plástico	Vidro
Metal	2	0	0	1
Papel	0	4	0	0
Plástico	1	0	2	2
Vidro	0	0	1	6

2) Analisando a matriz de confusão, você pôde observar que:

Todas as imagens do conjunto de teste estão sendo classificadas corretamente

Algumas imagens de metal são erroneamente classificadas como:  Papel  Plástico  Vidro

Algumas imagens de papel são erroneamente classificadas como:

Algumas imagens de plástico são erroneamente classificadas como:  Metal  Papel  Vidro

Algumas imagens de vidro são erroneamente classificadas como:  Metal  Papel  Plástico

3) Isto indica o que?

O modelo precisa ser melhorado

Finalizar

GQS INCoD ine UFSC COMPUTAÇÃO NA ESCOLA

11. Aluno clica no item da interface “Parte 5: Melhoria do modelo”

CodeMaster  
Home Aluno Professor Administrador Machine Learning para Todos

### Parte 5: Melhoria do modelo


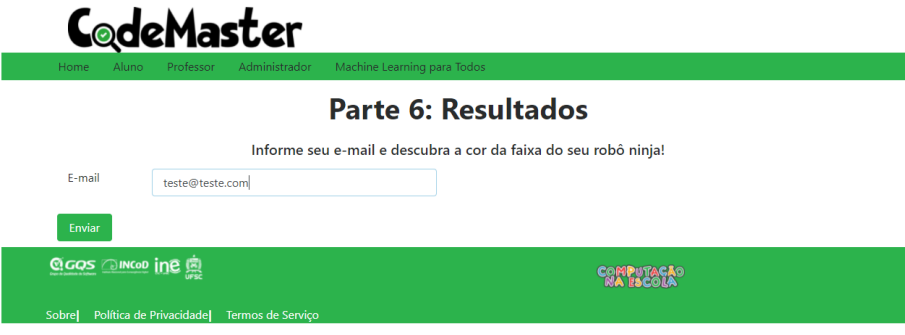

E-mail:

Você tentou melhorar o modelo?

Todos os campos são obrigatórios e devem ser preenchidos corretamente (email deve ser válido)

GQS INCoD ine UFSC COMPUTAÇÃO NA ESCOLA

Sobre | Política de Privacidade | Termos de Serviço

<p>12. Aluno preenche o formulário e clica em confirmar</p>	 <p>The screenshot shows the CodeMaster interface for 'Parte 5: Melhoria do modelo'. It features a navigation bar with 'Home', 'Aluno', 'Professor', 'Administrador', and 'Machine Learning para Todos'. The main heading is 'Parte 5: Melhoria do modelo'. Below it, there is an 'E-mail' input field containing 'teste@teste.com' and a dropdown menu for 'Você tentou melhorar o modelo?' with the selected option 'sim, várias vezes'. A green 'Finalizar' button is at the bottom. The footer includes logos for GQS, INCoD, INE, and UFSC, along with the text 'COMPUTAÇÃO NA ESCOLA' and links for 'Sobre', 'Política de Privacidade', and 'Termos de Serviço'.</p>																										
<p>13. Aluno clica no item da interface “Parte 6: Avaliação”</p>	 <p>The screenshot shows the CodeMaster interface for 'Parte 6: Resultados'. It features the same navigation bar as the previous screenshot. The main heading is 'Parte 6: Resultados'. Below it, there is a prompt 'Informe seu e-mail e descubra a cor da faixa do seu robô ninja!' and an 'E-mail' input field containing 'teste@teste.com'. A green 'Enviar' button is at the bottom. The footer is identical to the previous screenshot.</p>																										
<p>14. Sistema retorna o resultado final da avaliação da rubrica</p>	 <p>The screenshot shows the CodeMaster interface displaying the final evaluation results. On the left, a card shows 'Nota: 7.31' and 'O nível do seu projeto é... verde!' with a green robot character. Below the robot is a row of seven colored arrows (grey, yellow, orange, red, purple, blue, green) and a link 'Clique aqui para descobrir como melhorar sua pontuação!'. On the right, a table shows the criteria and their scores:</p> <table border="1" data-bbox="829 996 1380 1467"> <thead> <tr> <th>Critério</th> <th>Pontuação</th> </tr> </thead> <tbody> <tr> <td>Quantidade de imagens</td> <td>2</td> </tr> <tr> <td>Distribuição do conjunto de dados</td> <td>1</td> </tr> <tr> <td>Rotulagem das imagens</td> <td>1</td> </tr> <tr> <td>Treinamento - Transfer Learning</td> <td>2</td> </tr> <tr> <td>Análise de acurácia por categoria</td> <td>0</td> </tr> <tr> <td>Interpretação de acurácia</td> <td>2</td> </tr> <tr> <td>Análise da matriz de confusão</td> <td>1</td> </tr> <tr> <td>Interpretação da matriz de confusão</td> <td>2</td> </tr> <tr> <td>Ajustes/melhorias feitas</td> <td>1</td> </tr> <tr> <td>Testes com novos objetos</td> <td>1</td> </tr> <tr> <td>Análise dos resultados de testes</td> <td>2</td> </tr> <tr> <td>Interpretação dos testes</td> <td>2</td> </tr> </tbody> </table> <p>The footer is identical to the previous screenshots.</p>	Critério	Pontuação	Quantidade de imagens	2	Distribuição do conjunto de dados	1	Rotulagem das imagens	1	Treinamento - Transfer Learning	2	Análise de acurácia por categoria	0	Interpretação de acurácia	2	Análise da matriz de confusão	1	Interpretação da matriz de confusão	2	Ajustes/melhorias feitas	1	Testes com novos objetos	1	Análise dos resultados de testes	2	Interpretação dos testes	2
Critério	Pontuação																										
Quantidade de imagens	2																										
Distribuição do conjunto de dados	1																										
Rotulagem das imagens	1																										
Treinamento - Transfer Learning	2																										
Análise de acurácia por categoria	0																										
Interpretação de acurácia	2																										
Análise da matriz de confusão	1																										
Interpretação da matriz de confusão	2																										
Ajustes/melhorias feitas	1																										
Testes com novos objetos	1																										
Análise dos resultados de testes	2																										
Interpretação dos testes	2																										

Ao final do TCC o sistema automatizado de avaliação estará disponível no site do CodeMaster (<http://apps.computacaonaescola.ufsc.br:8080>). O software estará disponível no gitlab da ufsc como parte da ferramenta CodeMaster, registrado em nome da UFSC com número de registro BR512018051790-0 pelo INPI.

## 7. CONCLUSÃO

Neste trabalho, foi desenvolvida uma avaliação automatizada da aprendizagem de ML voltada a classificação de imagens para os anos finais do ensino fundamental, focada no curso Machine Learning Para todos!. Primeiro, foi realizada a análise teórica sobre o ensino de ML na educação básica no Brasil e no mundo, da aprendizagem e a automação da mesma, e sobre *deep learning* e o processo de *machine learning* (O1). Após isso, foi feita a análise do estado da arte em relação a análise automática da aprendizagem de ML na educação básica, revelando que o assunto é pouco explorado, e reforçando a necessidade deste trabalho (O2). Com essa análise, foi possível definir um modelo de avaliação, tendo em mente diferentes tipos de avaliação, a rubrica do curso, e a importância do *feedback* (O3). O próximo passo foi automatizar a avaliação desse modelo desenvolvido, usando, entre outras técnicas, *deep learning* (O4). Por fim, a avaliação automatizada para o curso Machine Learning para Todos! foi integrada no sistema CodeMaster (O5).

Considerando os resultados do levantamento do estado da arte, o possível impacto deste trabalho é grande. Um recurso de aprendizado de *Machine Learning* com foco para alunos no final do ensino fundamental com avaliação automatizada é raro, especialmente em português. Por isso, este é um passo importante na base da *AI Literacy* para os jovens estudantes brasileiros.

Como trabalhos futuros, alguns aspectos da avaliação podem ser melhorados, adições podem ser feitas, para ter-se um sistema mais completo e robusto. Pode-se implementar os dois critérios da rubrica que foram retirados, criando métodos para a detecção da limpeza, qualidade e relevância dos dados, e garantir que imagens impróprias não sejam incluídas no treinamento do modelo. Além disso, seria benéfico a adição de um método de avaliação em massa, pelo qual um professor envia um arquivo com múltiplas respostas de seus alunos, e o sistema retorna a nota de cada um.

## REFERÊNCIAS

ALA-MUTKA, K. M. **A Survey of Automated Assessment Approaches for Programming Assignments**. Computer Science Education, v. 15, n. 2, p. 83-102, 2005.

AMERSHI, S. et al. **Software Engineering for Machine Learning: A Case Study**. Proc. of the IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice, Montreal, Quebec, Canada, 2019.

ANDRADE, D. F.; TAVARES, H. R.; VALLE, R. C. **Teoria da Resposta ao Item: Conceitos e Aplicações**. São Paulo: ABE — Associação Brasileira de Estatística, 4º SINAPE, 2000.

ANGULAR. **The modern web developer's platform**. 2021. Disponível em: <<https://angular.io>>. Acesso em: Dezembro de 2021

AI4K12. **AI4K12 Big Idea #3 Progression Chart - Working Draft of Big Ideia 3**. 19 de Novembro de 2020. Disponível em: <[https://drive.google.com/file/d/1QL6I\\_I5cdNTVnYBIZ3\\_Lxur2DgFjmG\\_d/view](https://drive.google.com/file/d/1QL6I_I5cdNTVnYBIZ3_Lxur2DgFjmG_d/view)>. Acesso em: Dezembro de 2020

ALVES, N. C. **CodeMaster: Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual**. Dissertação (Programa de Pós-Graduação em Ciência da Computação (PPGCC)) – Universidade Federal de Santa Catarina, 2019

BRASIL. **Base Nacional Comum Curricular (BNCC). Educação é a Base**. Brasília, MEC/CONSED/UNDIME, 2017. Disponível em: <[http://basenacionalcomum.mec.gov.br/images/BNCC\\_publicacao.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_publicacao.pdf)>. Acesso em março de 2021.

BLOOM, B.S. et al. **Taxonomy of educational objectives**. Vol. 1: Cognitive domain. New York: McKay, v. 20, p. 24, 1956.

BRANCH, R. M. **Instructional Design: The ADDIE Approach**. New York: Springer, 2009.

BNE. **Vagas de Emprego na Área de Informática**, 2020. Disponível em: <<https://www.bne.com.br/vagas-de-emprego-na-area-de-informatica?Area=Inform%C3%A1tica&Sort=0&Page=1>>. Acesso em: Outubro 2020.



BRASSCOM. **Índice BRASSCOM de Convergência Digital**, 2012. Disponível em: <[https://brasscom.org.br/wp-content/uploads/2017/08/ibcd\\_2012.pdf](https://brasscom.org.br/wp-content/uploads/2017/08/ibcd_2012.pdf)>. Acesso em: Setembro de 2020.

BRASSCOM. **Relatório Setorial**, 2019. Disponível em: <<https://brasscom.org.br/relatorio-setorial-de-tic-2019/>>. Acesso em: Setembro de 2020.

CARVALHO, L. et al. **Detecção precoce de evasão em cursos de graduação presencial em Computação: um estudo preliminar**. Anais do Workshop sobre Educação em Computação ,SBC, 2019

CARPENTER, G. A.; GROSSBERG, S. **The ART of adaptive pattern recognition by a self-organizing neural network**. Computer, v. 21, n. 3, p. 77–88, 1988.

CLICK4IT. **General Guidelines for developing assessments**, 2013. Disponível em: <[http://www.click4it.org/images/e/e1/Toolkit\\_Assessment.pdf](http://www.click4it.org/images/e/e1/Toolkit_Assessment.pdf)>. Acesso em: 01 de Dezembro de 2020

CME. **RESOLUÇÃO CME Nº02/2011**. Conselho Municipal de Educação de Florianópolis, 2011.

COMPUTAÇÃO NA ESCOLA, 2020. Disponível em: <<https://cursos.computacaonaescola.ufsc.br>> .Acesso em: Setembro 2020

CORDEIRO, A. M. et al . **Revisão sistemática: uma revisão narrativa**. Revista do Colégio Brasileiro de Cirurgiões, v. 34, n. 6, p. 428-431, 2007.

CORTINA, J. M. **What is coefficient alpha? An examination of theory and applications**. Journal of Applied Psychology. v. 78, p. 98-104, 1993.

CRONBACH, L. J. **Coefficient alpha and the internal structure of tests**. Psychometrika, v. 16, n. 3. p. 297–334, 1951.

CSTA. ACM. **CSTA K–12 Computer Science Standards, Revised 2017**, 2017. Disponível em: <<https://www.doe.k12.de.us/cms/lib/DE01922744/Centricity/Domain/176/CSTA%20Computer%20Science%20Standards%20Revised%202017.pdf>>. Acesso em: Setembro de 2020.

DEMETRIO, M. F.. **Desenvolvimento de um analisador e avaliador de código de App Inventor para ensino de computação**. Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Ciências Da Computação, UFSC, Florianópolis, 2017.

ESTÚDIO NSC. Diário Catarinense. **Mais de 30 mil estudantes de Florianópolis receberão chips com internet para auxílio nos estudos**, 25 de Fevereiro de 2021. Disponível em: <https://www.nsctotal.com.br/noticias/mais-de-30-mil-estudantes-de-florianopolis-recebera-o-chips-com-internet-para-auxilio-nos>>. Acesso em: Abril 2021

EVANGELISTA, I.; BLESIO, G.; BENATTI, E. **Why Are We Not Teaching Machine Learning at High School? A Proposal**. Proc. of the World Engineering Education Forum - Global Engineering Deans Council, Albuquerque, NM, USA, 2018.

FERREIRA, M. N. F.. **Um modelo para uma unidade de ensino sobre princípios do design visual de interfaces de app: um estudo de caso**. Dissertação de mestrado, UFSC, Florianópolis, 2020

FLASK. **Web development, one drop at a time**. 2021. Disponível em: <https://flask.palletsprojects.com/en/2.0.x/>>. Acesso em: Dezembro de 2021

GARCIA, A. B. **Avaliação de aprendizagem de conceitos de *machine learning* nos anos finais do ensino fundamental**. Dissertação do Programa de Pós-Graduação em Métodos e Gestão em Avaliação – PPGMGA, UFSC, Florianópolis, 2022.

GLORFELD, L. W. **An improvement on Horn's parallel analysis methodology for selecting the correct number of factors to retain**. Educational and Psychological Measurement, v. 55, n. 3, p. 377-393, 1995.

GONÇALVES, R. Q. **Ensino de Gerenciamento de Projetos de Software Mediado por Ferramentas**. 2017. Tese de Doutorado - Universidade Federal de Santa Catarina.

GOODFELLOW, I. et al. **Generative Adversarial Nets**. Université de Montréal, 2014.

GRESSE VON WANGENHEIM, C.; et al. **CodeMaster – Automatic Assessment and Grading of App Inventor and Snap! Programs**. Informatics in Education, 17(1), 2018, 117-150.

GRESSE VON WANGENHEIM, C.; MARQUES, L. S.; HAUCK, J. C. R. **Machine Learning for All – Introducing Machine Learning in K-12**. SocArXiv, 2020.

HINTON, G., SEJNOWSKI, T. J. **Unsupervised Learning: Foundations of Neural Computation**. Cambridge: The MIT Press, 1999.

INEP. Sinopses Estatísticas da Educação Superior: graduação. Graduação. 2019. Disponível em: <http://inep.gov.br/sinopses-estatisticas-da-educacao-superior>. Acesso em: março de 2021.

JAKHAR, D.; KAUR, I. **Artificial intelligence, machine learning and deep learning: definitions and differences**. Clinical and Experimental Dermatology, v. 45, p. 131–132, jan. 2020.

JING, M. South China Morning Post. **China looks to school kids to win the global AI race**, 3 de Maio de 2018. Disponível em: <<https://www.scmp.com/tech/china-tech/article/2144396/china-looks-school-kids-win-global-ai-race>>. Acesso em: Outubro 2020

KAY, D. et al. . **Automated grading assistance for student programs**. Proc. of the 25th SIGCSE technical symposium on Computer science education, Phoenix, Arizona, EUA, p. 381 – 382, 1994.

KULHAVY, R., & STOCK, W. **Feedback in written instruction: The place of response certitude**. Educational Psychology Review, v. 1, n. 4, pp. 279–308, 1989.

LEE, I. et al. **Computational thinking for youth in practice**. ACM Inroads, v. 2, n. 1, p. 32–37, 2011.

LIMESURVEY. **Turn questions into answers**. Disponível em: <<https://www.limesurvey.org>>. Acesso em: Janeiro de 2022

LINKEDIN. **Profissões Emergentes 2020**, 2020. Disponível em: <[https://business.linkedin.com/content/dam/me/business/en-us/talent-solutions/emerging-jobs-report/Emerging\\_Jobs\\_Report\\_Brazil.pdf](https://business.linkedin.com/content/dam/me/business/en-us/talent-solutions/emerging-jobs-report/Emerging_Jobs_Report_Brazil.pdf)>. Acesso em: Outubro 2020.

LONG, D.; MAGERKO, B. **What is AI Literacy? Competencies and Design Considerations**. Proc. of the CHI Conference on Human Factors in Computing Systems., Honolulu, HI, USA, 2020.

SALVADOR, G., GRESSE VON WANGENHEIM, C., RAUBER, M., GARCIA, A., BORGATTO, A. F. **Avaliação de Aprendizagem de Machine Learning na Educação Básica: Um Mapeamento da Literatura**. Anais do 29º WEI – Workshop sobre Educação em Computação, online, 2021

SCHMIDHUBER, J. **Deep learning in neural networks: An overview**. Neural Networks, v. 61, p. 85–117, 2015.

MARQUES, L. S. **Ensino de Machine Learning na Educação Básica: um Mapeamento Sistemático do Estado da Arte**. SBIE, Porto Alegre, RS, Brasil, p. 278-287 , 2021.

MARQUES, L. S.; GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R. **Teaching Machine Learning in School: A Systematic Mapping of the State of the Art**. Informatics in Education, v. 19, n. 2, p. 283–321, 2020.

MARQUES, L. S. GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R. **Introdução à Machine Learning- Desenvolvimento de uma unidade instrucional para ensino de ML na educação básica.** Relatório técnico ISSN 2236-5281 v. 1 n. 1, GQS/INCoD/INF/UFSC, 2020.

MEC, **Base Nacional Comum Curricular**, 2018. Disponível em: <[http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf)>. Acesso em: Setembro de 2020.

MYSQL, 2021. Disponível em: < <https://www.mysql.com/>>. Acesso em: dezembro de 2021.

NODEJS, **Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.** 2021. Disponível em: <<https://nodejs.org/en/>>. Acesso em, Dezembro de 2021

NOTAI.TECH, **NudeNet: Neural Nets for Nudity Classification, Detection and selective censoring**, 2019. Disponível em: <<https://github.com/notai-tech/NudeNet>>. Acesso em: Setembro de 2021.

O'BRIEN C., The Mercury News, **Get ready for the decade of gamification**, 21 de Outubro de 2010. Disponível em: <<https://www.mercurynews.com/2010/10/21/obrien-get-ready-for-the-decade-of-gamification/>>. Acesso em: Janeiro de 2021

PINTO, N. et al. **Quando a Aluna se Torna a Mestre: Um Relato da Experiência de Alunas de Graduação Aplicando Dinâmicas de Ensino de Computação para Alunas de Ensino Médio.** Anais do Workshop sobre Educação em Computação, SBC, 2020.

POWERS, D. **Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation.** p. 25, 2008

RADHAKRISHNAN, P. **What are Hyperparameters? and How to tune the Hyperparameters in a Deep Neural Network?.** Disponível em: <<https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a>>. Acesso em: Dezembro de 2020.

RODRIGUEZ, A. **Restful web services: The basics.** IBM developerWorks, 2008.

RUSSELL, S. J., NORVIG, P. **Artificial Intelligence: A modern Approach.** 3ª edição. New Jersey: Prentice Hall, 2010

SBC, **Diretrizes para ensino de Computação na Educação Básica**, 2019. Disponível em: <<https://www.sbc.org.br/documentos-da-sbc/send/203-educacao-basica/1220-bncc-em-itinerario-informativo-computacao-2>>. Acesso em: Dezembro de 2020.

SOFTEX. **Relatório Anual**, 2014. Disponível em: <<https://softex.br/booksoftex/>>. Acesso em: Setembro de 2020.

SOUZA, A. C. DE et al. **Propriedades psicométricas na avaliação de instrumentos: avaliação da confiabilidade e da validade**. Epidemiologia e Serviços de Saúde, v. 26, n. 3, p. 649–659, 2017.

SPRING, **Spring makes Java simple. Modern. Productive. Reactive. Cloud-ready**. 2021. Disponível em: <<https://spring.io/projects/spring-boot>>. Acesso em: Dezembro de 2021

STEGEMAN, M.; BARENDSSEN, E.; SMETSERS, S. **Designing a rubric for feedback on code quality in programming courses**. Proc. of the 16th Koli Calling International Conference on Computing Education Research, p. 160-164, 2016.

TACKE, O. **SNORDIAN's H5PxAPIkatchu**, 2017. Disponível em: <<https://wordpress.org/plugins/h5pxapikatchu/>>. Acesso em: Dezembro 2020

TEACHABLE MACHINE. **Train a computer to recognize your own images, sounds, & poses**. 2021. Disponível em: <<https://teachablemachine.withgoogle.com>>. Acesso em: Janeiro de 2022

TISSENBAUM, M. et al. **From computational thinking to computational action**. Communications of the ACM, v. 62, n. 3, p. 34–36, 2019.

TOURETZKY, D. S. et al. **K-12 Guidelines for Artificial Intelligence: What Students Should Know**. Proc. of the ISTE Conference, Philadelphia, PA, USA. 2019

VALUEVA, M. V. et al. **Application of the residue number system to reduce hardware costs of the convolutional neural network implementation**. Mathematics and Computers in Simulation, v. 177, p. 232–243, 2020.

WOOLFOLK, A. **Educational psychology**. 10ª edição. Boston: Pearson/Allyn and Bacon, 2007.

ZHU, X.; GOLDBERG, A. B. **Introduction to Semi-Supervised Learning**. Synthesis Lectures on Artificial Intelligence and Machine Learning, v. 3, n. 1, p. 1–130, 2009.

**Anexo A - Link para o código fonte**

<https://codigos.ufsc.br/100000000394729/CodeMaster/-/tree/tcc/marcos.laydner>

## AVALIAÇÃO AUTOMATIZADA DO APRENDIZADO DE MACHINE LEARNING PARA O ENSINO FUNDAMENTAL

Marcos Silva Laydner

Departamento de Informática e Estatística, Universidade Federal de Santa Catarina  
Florianópolis/SC, Brasil  
[marcos.laydner@grad.ufsc.br](mailto:marcos.laydner@grad.ufsc.br)

### Resumo

Com a crescente evolução e popularização do uso de tecnologias de *Machine Learning* no dia a dia, é imperativo que a população geral tenha um entendimento básico sobre como essas tecnologias funcionam, de forma que sua utilização seja melhor aproveitada. Para isso, é importante que o ensino desses conceitos seja feito cedo já na Educação Básica. Com isso, algumas iniciativas têm surgido, porém a maioria não possui um modelo de avaliação dessa aprendizagem. Assim, o presente trabalho visa responder a pergunta de pesquisa: “Como automatizar a avaliação da aprendizagem de ML voltada a classificação de imagens nos anos finais do ensino fundamental?”. Dessa forma, tem-se como objetivo desenvolver e testar um modelo de análise da aprendizagem de ML na Educação Básica, adotando técnicas de *Deep Learning* para automatizar o processo. Isso é alcançado por meio da fundamentação teórica dos principais conceitos, uma revisão do estado da arte, o desenvolvimento do modelo de avaliação, e a sua automatização, e sua integração e avaliação no contexto de um curso da iniciativa Computação na Escola. Dessa forma, criou-se uma extensão da ferramenta CodeMaster que permite automaticamente avaliar a aprendizagem de *Machine Learning* do aluno, facilitando o processo, e aumentando a chance de adoção do ensino de computação nas escolas brasileiras.

**Palavras chave:** Educação Básica, Avaliação, Inteligência Artificial, Machine Learning.

### 1. Introdução

Desde de Turing até os dias atuais, a influência da computação no dia a dia das pessoas vem crescendo consideravelmente. Diversos aspectos da vida atual, e quase todos setores da economia têm sido fortemente afetados, ou são agora movimentados, por sistemas computacionais. Com toda essa influência, a tendência é que o conhecimento sobre essas tecnologias seja cada vez mais importante para todos, e não só estudantes de computação, estejam preparados para as carreiras do século XXI (CSTA, 2017).

Uma das áreas da computação com grande influência na sociedade atual é a de Inteligência Artificial (IA) e Aprendizado de máquina (Machine Learning - ML). ML é um campo da IA que estuda tecnologias que permitem que máquinas aprendam a partir de dados passados, sem terem de ser programadas especificamente (JAKHAR; KAUR, 2020). Machine Learning tem sido utilizado para identificação de spam, sistemas de busca, mecanismos de recomendação,

entre outros usos que fazem parte da vida diária de muitas pessoas. Uma das tarefas típicas de ML é o reconhecimento de imagens, o foco deste trabalho, usado por exemplo, no reconhecimento facial para desbloqueio de celulares. Porém, a falta de conhecimento da tecnologia por trás de toda essa facilidade, utilizada como uma caixa preta, pode trazer insegurança e estranheza, reduzindo seu impacto positivo e seu potencial de desenvolvimento (EVANGELISTA et al., 2018).

Ao redor do mundo observa-se uma tendência recente de abordar também IA/ML na Educação Básica. O ensino de IA/ML vem sendo incluído em vários países como a China, em que Inteligência Artificial é um conteúdo obrigatório para alunos do ensino médio (JING, 2018). Para este fim estão sendo criadas diretrizes para o ensino, como K-12 Guidelines for Artificial Intelligence (TOURETZKY et al., 2019), que indicam os conteúdos importantes que se encaixam no contexto de Inteligência Artificial na educação básica, que facilitou e impulsionou o crescimento de cursos no mundo. Nos últimos anos também surgiram diversos cursos de IA/ML para Educação Básica, que utilizam diferentes técnicas de ensino, desde ambientes de programação baseados em blocos, até o uso simples de python e frameworks de Machine Learning (MARQUES et al., 2020).

Porém, atualmente, o ensino de computação nas escolas brasileiras é inexistente ou insuficiente para as necessidades do mundo moderno. A maioria das instituições brasileiras que apresentam o ensino de computação, focam somente em como utilizar as tecnologias (IT literacy), enquanto o mundo precisa que os alunos tenham um conhecimento mais profundo, e uma fluência (IT fluency) com a TI, para que adquiram a capacidade de aprender e utilizar eventuais novas tecnologias do futuro, ao longo de sua vida. Por isso, a inclusão do ensino de computação nas escolas se tornou uma tendência mundial (PINTO et al., 2020). Mais recentemente, isto também inclui o ensino de IA/ML como uma das áreas de conhecimento da computação.

Portanto, notando a atual lacuna de modelos educacionais que avaliam sistematicamente o aprendizado dos conceitos de Inteligência Artificial e Aprendizado de Máquina, este projeto busca responder a seguinte pergunta de pesquisa: Como automatizar a avaliação da aprendizagem de ML voltado a classificação de imagens nos anos finais do ensino fundamental?

## **2. Fundamentação**

Este capítulo apresenta os conceitos referentes a deep learning, e da avaliação da aprendizagem de alunos. Além disso, são apresentados diferentes desafios e iniciativas sobre o ensino de ML para o ensino fundamental

### **2.1 Ensino de ML na Educação Básica**

Considerando-se o impacto de IA e ML na sociedade moderna, e sua presença no dia a dia das pessoas, o acesso ao aprendizado sobre o campo, além do ensino superior, ainda é limitado. Com exceção de países como a China, em que Inteligência Artificial é um conteúdo obrigatório para alunos do ensino médio (JING, 2018), o assunto tipicamente está limitado ao ensino superior, não sendo devidamente contemplado mesmo em diretrizes de aprendizado de computação para a educação básica.

Porém, o assunto vem ganhando atenção nos últimos anos, e diferentes iniciativas surgiram com o objetivo de determinar diretrizes para um currículo, como K-12 Guidelines for Artificial Intelligence (TOURETZKY et al., 2019), ou CSTA K-12 Computer Science Standards (CSTA, 2017). Essas diretrizes citam 5 “grandes ideas” que todos estudantes devem saber em relação a IA (TOURETZKY et al., 2019):



- **Percepção** - é importante que o aluno entenda que a percepção de computadores de linguagem oral ou de imagens requer um grande conhecimento do respectivo domínio ;
- **Representação e Raciocínio** - o estudante deve entender o conceito de representação an entender que computadores constroem representações usando dados, e que delas pode-se derivar novas informações por meio da aplicação de algoritmos de raciocínio;
- **Aprendizado** - o aluno precisa compreender que o aprendizado de máquina é um tipo de inferência estatística que acha padrões nos dados;
- **Interação Natural** - é importante que o estudante saiba que por mais que computadores consigam entender linguagem natural de forma limitada, eles atualmente possuem menos habilidades conversacionais e raciocínio geral de uma criança;
- **Impacto Social** - é imperativo que o aluno seja capaz de identificar maneiras que IA contribui positivamente em suas vidas, além de entender que para a construção ética de um sistema de Inteligência Artificial é preciso atenção nos quesitos de transparência e equidade.

Seguindo as tendências mundiais, e inspiradas pelas diretrizes citadas, a iniciativa Computação na Escola iniciou o desenvolvimento do curso Machine Learning para Todos! (GRESSE VON WANGENHEIM et al., 2020). O curso tem como alvo estudantes entre 10 e 14 anos, sem nenhum conhecimento prévio de ML. Porém, é assumido que o aluno possui entendimento de conceitos básicos de algoritmos e programação, ou desenvolvimentos de apps com o App Inventor em aulas anteriores. A partir deste, foi realizada a automação da avaliação.

## 2.2 Avaliação da Aprendizagem

A avaliação é uma parte importante do processo de ensino e aprendizagem, pois avalia o desempenho de algo ou alguém. Esse desempenho pode ser medido em uma nota atribuída ao objeto avaliado, e assim, um feedback personalizado é criado (BRANCH, 2009). A avaliação é importante tanto para o educando, para que o mesmo conheça seu desempenho, quanto para o educador, para avaliar a retenção de informação dos alunos e a efetividade de seus métodos de transferência de conhecimento (ALVES, 2019). De acordo com Stegeman et al. (2016), o aluno precisa saber: o que um bom desempenho em uma tarefa é; como seu próprio desempenho se compara com um bom; e o que fazer para reduzir a lacuna entre os dois.

Ao desenvolver um modelo e avaliação, é importante ter em mente os objetivos de aprendizagem do assunto abordado. A taxonomia de Bloom foi criada para auxiliar na classificação dos objetivos de aprendizagem em crescentes tipos de complexidade e especificidade (BLOOM, 1956). Quando se trata de educação básica, geralmente somente os 3 primeiros níveis são abordados. O primeiro nível, de Conhecimento, envolve reconhecer e lembrar fatos e conceitos básicos, sem necessariamente entender seu significado. (WOOLFOLK, 2007). O segundo nível, Compreensão, engloba a demonstração de um entendimento das ideias relevantes, por meio de resumos, generalizações, etc. O terceiro nível, de aplicação, engloba reunir o conhecimento e compreensão adquiridos e aplicá-los a certas situações, como por meio da soluções de problemas simples.

Tabela 5. Níveis de conhecimento de Bloom (BLOOM, 1956), relacionados com avaliações disponíveis no H5P. Adaptado de (CLICK4IT, 2013)

Domínio Cognitivo (Nível)	Exemplos de tipos de avaliação	Palavras Chave
1. Conhecimento (recordar informação)	quizzes, múltipla e única escolha, jogos de memória, preencher espaço em branco, arrastar palavras	Arranjar definir, descrever, rotular, listar, memorizar, reconhecer, relacionar, reproduzir, selecionar, citar

2. Compreensão (recordar e interpretar informação)	Formulários (onde o aluno explica sua interpretação de conceitos ou de um cenário), atividade de cenários ramificados, emparelhamento de imagens	explicar, reiterar, reformular, criticar, classificar, resumir, ilustrar, traduzir, revisar, reportar, discutir, reescrever, estimar, interpretar, teorizar, parafrasear, referenciar, exemplificar
3. Aplicação (usar informação abstrata em situações concretas)	Avaliação por desempenho	Usar, aplicar, descobrir, administrar, executar, resolver, produzir, implementar, construir, mudar, preparar, conduzir, performar, reagir, responder,

## 2.3 Deep Learning

Neural Networks (Redes Neural) é uma área do estudo de Machine Learning que se inspira no funcionamento dos cérebros dos animais. Uma rede neural consiste de vários processadores interconectados chamados de neurons (neurônios), cada um produzindo uma sequência de ativações de valores reais. No contexto de ML, os neurônios podem ser ativados pelas conexões ponderadas dos neurônios anteriores, assim, o aprendizado está em descobrir os pesos que fazem com que a RN exiba o comportamento desejado. Quanto mais níveis de neurônios são adicionados, mais profunda a rede fica (SCHMIDHUBER, 2015). Portanto, Deep Learning é a técnica de ML que envolve usar redes neurais profundas, ou seja, com muitas camadas de neurônios, que têm seus pesos ajustados a cada iteração de aprendizado. Essa área tem uma aplicação ampla, podendo ser usada em aprendizagem supervisionada, semi-supervisionada e não supervisionada (SCHMIDHUBER, 2015).

Quando se trata do desenvolvimento de modelos de ML, há inúmeras maneiras de abordar o mesmo problema. Com a popularização da tecnologia e crescente número de estudos sobre o assunto, diferentes arquiteturas, frameworks e bibliotecas de código foram desenvolvidas para auxiliar no desenvolvimento de RN, que muitas vezes englobam o processo de ML inteiro.

Quanto a programação de fato, existem diversas bibliotecas de código, como TensorFlow (TENSORFLOW, 2015), uma biblioteca de código aberto para criação, treinamento e exportação de redes neurais, disponível para diferentes plataformas e linguagens. Além disso, ferramentas como o Google Teachable Machine (TEACHABLE MACHINE, 2022) permitem a criação de redes de classificação de imagens com abstrações de alto nível, sem a necessidade de conhecimentos de programação.

## 3. Desenvolvimento e Automação do Modelo de Avaliação

### 3.1 Análise do Contexto

O público alvo do curso são alunos de escolas brasileiras com idade entre 10 a 14 anos, correspondente aos anos finais do ensino fundamental. Nesse estágio do aprendizado, é esperado que o aluno seja alfabetizado na língua nativa (Português do Brasil) e possua certa familiaridade, mas não fluência, com a língua inglesa ou espanhol. Além disso, considera-se que o aluno compreende em alto nível o mundo natural, tecnológico e individual, e possui suficiente desempenho na escrita, leitura e compreensão de suas experiências, para ampliar sua percepção e analisar contextos maiores nos anos seguintes (BNCC, 2019). Atualmente, mesmo com a inclusão do tópico de pensamento computacional na área de conhecimento de matemática na Base Nacional Comum Curricular (BCCN), a aplicação do ensino de computação ainda é rara em escolas brasileiras, especialmente quando se trata de IA no ensino fundamental, e conseqüentemente nenhum conhecimento sobre computação ou machine learning é esperado, portanto, assume-se que o aluno não os possua.

De acordo com QEDU (2017), os dados da Prova Brasil mostram que 55% dos estudantes do 9º ano das escolas públicas brasileiras têm pelo menos um computador em casa e 42% destes passam mais de três horas por dia lidando com tecnologias. Em Santa Catarina, é perceptível um aumento na presença da tecnologia na vida dos alunos, visto que esses números aumentam para 75% e 60%, respectivamente (QEDU, 2017).

Por outro lado, observa-se a falta profissionais com formação em computação para ministrar aulas computação. A procura por cursos de formação de professores em computação é consideravelmente mais baixa do que por aqueles considerados tradicionais, como história, matemática etc. Além disso, em 2019, de 10.932 matrículas apenas 786 pessoas concluíram seus cursos nessa área, mostrando uma alta taxa de evasão (INEP, 2019). Portanto, o ensino de computação nas escolas ocorre muitas vezes de forma extracurricular ou interdisciplinar, com professores formados em outras disciplinas ensinado de forma integrada conteúdo de computação nas suas disciplinas consideradas mais tradicionais, como história, geografia etc.

### 3.2 Plano de Ensino do Curso Machine Learning para Todos!

De acordo com o plano de ensino (Tabela 4) a avaliação da aprendizagem do aluno foi dividida em duas etapas, a avaliação dos quizzes, e a do desempenho e são usadas em conjunto para calcular a nota final. O material das aulas está disponível no formato H5P (HP5.ORG, 2021) em forma de imagens, vídeos, exercícios, etc.. O material H5P também contém os quizzes, que são relacionados aos assuntos de suas respectivas aulas, tratando dos objetivos de aprendizado do curso. O curso está hospedado no Moodle da UFSC, o sistema de gestão da aprendizagem utilizado pela universidade (MOODLE, 2021), por meio do Moodle Grupos, e também de forma aberta no site de cursos da iniciativa Computação na escola (COMPUTAÇÃO NA ESCOLA, 2020). Já a parte da avaliação de desempenho, descrita pela rubrica, foi implementada como uma extensão do CodeMaster (DEMETRIO, 2017). A ferramenta é uma aplicação web gratuita que automaticamente avalia projetos criados com o App Inventor (APP INVENTOR, 2022) e Snap! (SNAP!, 2022), levando em conta critérios de programação e conceitos de design de interface e usabilidade.

Tabela 4 - Plano de ensino do curso Machine Learning Para Todos! Traduzido Gresse von Wangenheim et al. (2020)

Aula	Conteúdo	Objetivos de aprendizagem	Método Educacional	
Noções Gerais e importância				
1. Aula 1 do curso online	Motivação de IA e sua aplicação na vida diária	OA1	Slides interativos, video, demonstrações	Quiz: Slide 4 ( <i>drag-and-drop</i> ), 10 (múltipla escolha), 19 (múltipla escolha), 21 (múltipla escolha), 23 (múltipla escolha)
Conceitos básicos				
2. Aula 2 do curso online	Conceitos Básicos de ML: o que significa "aprender"	OA2, OA3, OA4	Slides interativos	Quiz: Slides 27 (verdadeiro ou falso) e 38 ( <i>drag-and-drop</i> )
Faça seu primeiro modelo (e aplicativo) de ML!				
3.1 Aula 3 do curso online	Classificação de recicláveis; preparação de dados (limpeza e rotulação)	OA3, OA5, OA11	Slides interativos Aula, atividade prática	Quiz: Slides 13 ( <i>drag-and-drop</i> ) e 20 ( <i>drag-and-drop</i> ); Avaliação baseada no desempenho do dataset preparado
3.2 Aula 4 do curso online	Treinamento e avaliação do modelo	OA4, OA6, OA7	Slides interativos, atividade prática com Google™, e quizzes	Quiz: Slides 22 (múltipla escolha), 23 (múltipla escolha), 29 (múltipla escolha), 30 (múltipla escolha) e 42 ( <i>drag-and-drop</i> ); avaliação baseada no desempenho do modelo e avaliação de análise e interpretação
Revisão do conteúdo e do processo de ML				
4.1 Aula 5 do curso	Revisão dos conceitos e processo de ML	OA1, OA2, OA5	Slides Interativos atividade prática	Quiz: Slide 23 ( <i>drag-and-drop</i> )

online				
Problemas éticos e impacto social de ML				
4.2 Aula 6 do curso online	Problemas éticos a respeito de IA/ML, limitações, riscos e oportunidades de emprego	OA1, OA9	Slides interativos, demonstração ( <i>MIT Moral Machine</i> )	Quiz: Slides 13 (múltipla escolha), 20 (múltipla escolha), e 22 (múltipla escolha)

### 3.3 Avaliação do Aprendizado

Para a criação de uma avaliação que mede o aprendizado de ML do aluno, é preciso que todos os objetivos de aprendizagem sejam englobados em pelo menos uma das avaliações previstas no plano de ensino.

Tabela 3 - Objetivos de aprendizagem do curso Machine Learning Para Todos!. Traduzido de Gresse von Wangenheim et al. (2020)

ID	Objetivo de Aprendizagem
OA1	Conhecer e identificar exemplos de aplicações de ML
OA2	Descrever conceitos básicos e o processo de Machine Learning
OA3	Identificar padrões nos dados
OA4	Simular o treinamento de uma rede neural simples
OA5	Coletar, limpar e rotular dados para o treinamento de um modelo de ML, entendendo como o algoritmo pode ser influenciado pelos dados
OA6	Treinar o modelo
OA7	Avaliar o desempenho do modelo de Machine Learning
OA8	Aplicar o modelo, desenvolvendo um aplicativo mobile
OA9	Discutir assuntos de ética e o impacto de ML na sociedade
OA10	Compartilhar o sistema desenvolvido
OA11	Propor iniciativas coletivas e individuais para resolver problemas ambientais na cidade ou comunidade

Os objetivos de aprendizagem de 1 a 4, e o 9 são avaliados por quizzes. Por meio do H5P, é possível introduzir quizzes relevantes ao assunto no meio dos slides, criando um fluxo contínuo em que o aluno é apresentado a um conceito, e logo em seguida avaliado sobre o mesmo. Cada aula possui seus slides, dentro dos quais estão os quizzes. As aulas recebem uma nota de 0 a 10 cada, baseada no desempenho dos respectivos quizzes. A nota total dos quizzes é a média das notas das aulas.

A automação dos quizzes inseridos no material didático usando H5P (H5P.ORG, 2021), é realizada pelos próprios mecanismos da ferramenta, que disponibiliza correção e feedbacks automáticos a cada questão respondida.

Para fins de utilizar os resultados também como parte de uma avaliação geral no curso, esses podem ser coletados por meio do plugin H5PxAPIkatchu (TACKE, 2017) e/ou no contexto do sistema de gerenciamento de aprendizado do moodle nos módulos de avaliação. Assim, por meio dos quizzes automatizados, é possível avaliar a compreensão do aluno sobre os conceitos básicos de IA e ML, exemplos de aplicações que utilizam dessa tecnologia, seus impactos no mundo a sua volta, e suas implicações éticas (OA1, OA2, OA3, OA4 e OA9).

Os dados dos questionários do moodle são manualmente extraídos pela função de exportação da plataforma, no formato de um CSV.

### 3.4 Avaliação do Desempenho

Com base na rubrica proposta por Gresse von Wangenheim et al. (2021) foi instanciada uma versão da rubrica voltada ao curso de Machine Learning para Todos!. Considerando a complexidade de análise de certos critérios propostos na rubrica (GRESSE VON WANGENHEIM et al., 2021), foi neste momento excluída a automação dos seguintes critérios:

- Relevância das imagens (C2)
- Limpeza dos dados (C4)

Estes apresentam-se um desafio, demandando tempo de estudo de desenvolvimento, tendo em mente que cada um precisaria do desenvolvimento, teste e implantação do seu próprio modelo de classificação de imagens. Por isso, estes são considerados para trabalhos futuros.

Tabela 19 - Rubrica de avaliação de aprendizagem com base no desempenho (GRESSE VON WANGENHEIM et al., 2021)

ID	Critério	Níveis de desempenho			Coleta de dados	Extrair dos dados coletados	Proposta de automação
		Baixo - 0 pt.	Aceitável - 1 pt.	Bom - 2 pt.			
<b>Gerenciamento de dados (OA3/OA12)</b>							
C1	Quantidade de imagens	Menos de 5 imagens por categoria	5 a 10 imagens por categoria	Mais de 10 imagens por categoria	Passo 1. Imagens do arquivo .tm	Quantidade total de imagens do conjunto de dados	Contar a quantidade de imagens de cada classe
C2	Distribuição do conjunto de dados	Alguma categoria que representa mais do que 30% dos dados, ou menos que 20%)	Categorias representam entre 30% e 20% dos dados	Todas as categorias representam 25% dos dados	Passo 1. Imagens do arquivo .tm	Quantidade de imagens para cada categoria do conjunto de dados	Comparar a quantidade de cada classe com a quantidade total de imagens
C3	Rotulagem das imagens	Menos de 20% das imagens rotuladas corretamente	De 20% a 99% das imagens rotuladas corretamente	Todas as imagens rotuladas corretamente	Passo 1. Imagens do arquivo .tm	Quantidade de imagens rotuladas corretamente	Aplicar um modelo de alta precisão para a tarefa de classificação das 4 tipos de reciclagem (metal, papel, plástico e vidro) a partir dos dados extraídos
<b>Treinamento de modelo de ML/Transfer Learning e Fine-Tuning (OA4)</b>							
C4	Treinamento - <i>Transfer Learning</i>	O modelo não foi treinado	O modelo foi treinado com os parâmetros padrão	O modelo foi treinado com parâmetros ajustados (época e taxa de aprendizagem)	Passo 1. Arquivo manifest.json do .tm (parâmetros de treinamento)	Valor dos parâmetros da época, taxa de aprendizagem, <i>batch size</i>	Verificar qual foi a arquitetura do comando de definição do learner, retirar qual o número de épocas e taxa de aprendizagem do comando de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML (OA5)</b>							
C5	Análise de acurácia por categoria	Categorias com baixa acurácia não identificadas	--	Todas as categorias com baixa acurácia identificadas corretamente	Passo 3	Valores de acurácia e respostas do Passo 3	Verificar se o comando de display de categorias com baixa acurácia foi executado e se respostas no widget de interpretação usando ipywidgets estão corretas
C6	Interpretação de acurácia	Interpretação da acurácia incorreta	--	Interpretação da acurácia incorreta	Passo 3	Valores de acurácia e respostas do Passo 3	Verificar se as respostas de interpretação estão corretas
C7	Análise da matriz de confusão	Mais do que uma classificação incorreta não identificadas na matriz de confusão	Uma classificação incorreta não identificada na matriz de confusão	Todas as classificações incorretas identificadas na matriz de confusão	Passo 4	Matriz de confusão e respostas do Passo 4	Verificar as classificações indicadas como incorretas pelo aluno e compará-las com as classificações incorretas reais indicadas na matriz de confusão
C8	Interpretação da matriz de confusão	Interpretação errada da análise da matriz de confusão	--	Interpretação correta da análise da matriz de confusão	Passo 4	Matriz de confusão e respostas do Passo 4	Verificar se as respostas de interpretação estão corretas
C9	Ajustes/melhorias feitas	Sem novas iterações de desenvolvimento	Uma nova iteração com alterações no conjunto de dados e/ou parâmetros de treinamento	Diversas novas iterações com alterações no conjunto de dados e/ou parâmetros de treinamento	Passo 5	Respostas do Passo 5	Verificar quantidade de iterações em que houve alterações no conjunto de dados e/ou parâmetros de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML/Testar e aperfeiçoar programas (OA5/OA11)</b>							
C10	Testes com novos objetos	Nenhum novo objeto testado	1-2 novos objetos testados	Mais de dois novos objetos testados	Passo 2	Quantidade de testes realizados	Verificar a resposta sobre quantidade de vezes que o comando de treinamento do modelo foi executado com novos objetos
C11	Análise dos resultados de testes	Indicação errada da quantidade de erros nos testes	--	Indicação correta da quantidade de erros nos testes	Passo 2	Quantidade de erros na análise dos resultados dos testes	Verificar se as respostas indicam o número correto de erros nos tests
C12	Interpretação dos	Interpretação	---	Interpretação correta	Passo 6	Resposta da	Verificar se respostas de interpretação

testes	errada			interpretação	estão de acordo com o desempenho
--------	--------	--	--	---------------	----------------------------------

Cada item recebe uma pontuação de 0 a 2, dependendo do desempenho. A partir das pontuações é calculada a nota final do aluno, dividindo a pontuação máxima por 2,4, para chegar numa nota de 0 a 10 em relação a rubrica.

Fórmula	Nota máxima
$(C1 + C2 + C3 + C4 + C5 + C6 + C7 + C8 + C9 + C10 + C11 + C12) / 2,4$	10

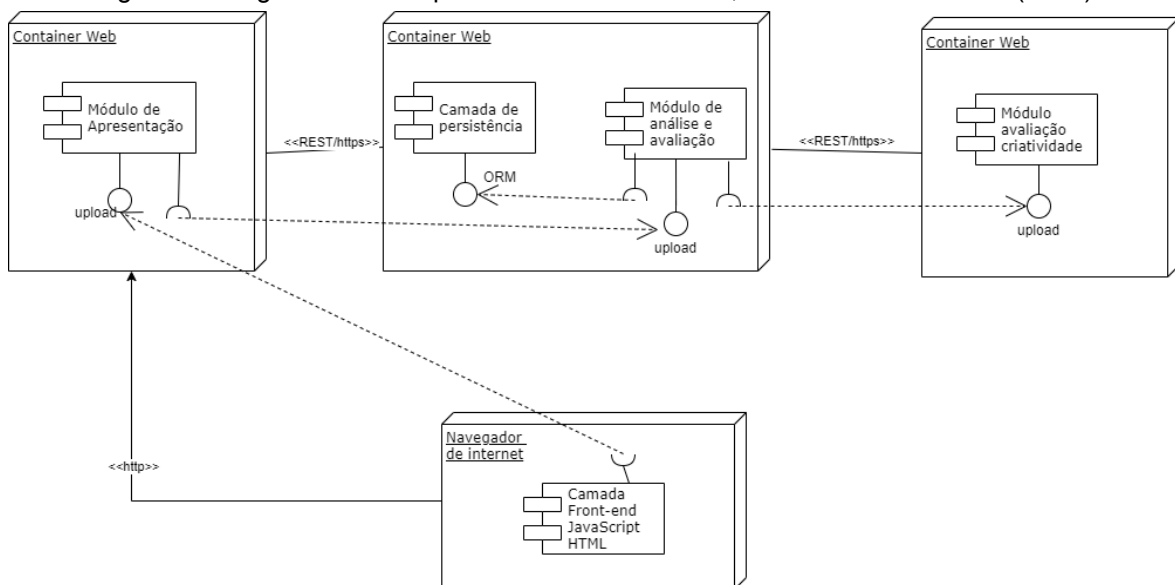
A automação da avaliação de desempenho é baseada na rubrica apresentada na Tabela 19, usando como entrada os artefatos criados pelo aluno ao longo do processo de aprendizagem como resultado de atividades práticas. Para ter acesso a estes artefatos, o aluno os envia nos respectivos formulários, para que eles possam ser coletados automaticamente em conjunto com as respostas.

Os dados coletados referentes aos artefatos criados pelos alunos incluem o arquivo do modelo de ML treinado na Google Teachable Machine (TEACHABLE MACHINE, 2022) e baixado em formato .tm. A partir deste arquivo podem ser extraídas as imagens utilizadas no treinamento. Este arquivo também contém um arquivo manifest.json, que contém os parâmetros usados para o treinamento. Além disso, os demais dados necessários para realizar a avaliação baseada no desempenho foram coletados inicialmente por formulário via Limesurvey (LIMESURVEY, 2022), e posteriormente implementados no CodeMaster (DEMETRIO, 2017).

#### 4. Integração com o CodeMaster

O sistema foi expandido a partir de uma versão atualizada de Demetrio (2017), conforme apresentado na Figura 9. O modelo de arquitetura do sistema que objetiva a separação das camadas de apresentação e análise em diferentes módulos foi mantido, com uma maior separação do módulo de apresentação, com a migração JSP para um container separado utilizando Typescript utilizando Angular como framework de desenvolvimento web.

Figura 9 - Diagrama de Componentes do CodeMaster, baseado em Demetrio (2017)

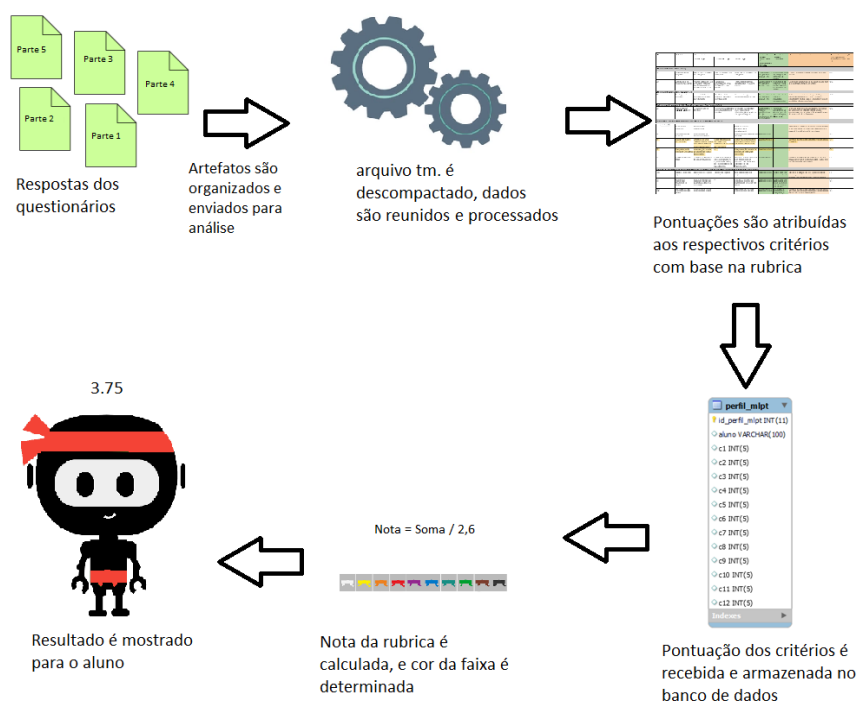


O módulo de análise a avaliação é um serviço web REST que, dentro do contexto da implementação do Machine Learning para Todos, é responsável por receber as respostas do aluno, encaminhá-las para o módulo de avaliação e criatividade para serem avaliadas, receber as notas dos critérios, e persisti-las no banco de dados. REST (RODRIGUEZ, 2008) é uma arquitetura que permite a separação de sistemas web em módulos (serviços). Para implementar esse serviço o CodeMaster V2.0 utiliza o framework Spring (SPRING, 2021), que abstrai os detalhes de baixo nível da implementação de comunicação entre os servidores e simplifica a implementação de um serviço REST, tornando simples a implementação, implantação e manutenção de serviços web stand-alone.

O módulo de apresentação é o responsável pelo controle da interface de usuário, e no contexto deste trabalho, envia as respostas e artefatos do aluno para o módulo de análise, além de apresentar o resultado final para o usuário. O componente de front-end utiliza as tecnologias Angular (ANGULAR, 2021) e Node.js (NODEJS, 2021) para facilitar o seu desenvolvimento como uma aplicação web stand-alone, e simplificar a comunicação com os outros servidores.

O módulo de avaliação e criatividade, neste trabalho, recebe as respostas do aluno, as avalia a partir dos algoritmos definidos na seção 5.2, e retorna a pontuação dos respectivos critérios. Este módulo utiliza Flask (FLASK, 2021), uma biblioteca para Python que, assim como as outras tecnologias usadas nos outros módulos, facilita o desenvolvimento de uma aplicação web e na comunicação com os outros módulos REST.

Figura 10 - Sequência da análise da rubrica Machine Learning para Todos



## 5. Conclusão

Neste trabalho, foi desenvolvida uma avaliação automatizada da aprendizagem de ML voltada a classificação de imagens para os anos finais do ensino fundamental, focada no curso Machine Learning Para todos!. Primeiro, foi realizada a análise teórica sobre o ensino de ML na educação básica no Brasil e no mundo, da aprendizagem e a automação da mesma, e sobre deep learning e o processo de machine learning (O1). Após isso, foi feita a análise do estado da arte em relação a análise automática da aprendizagem de ML na educação básica, revelando que o assunto é

pouco explorado, e reforçando a necessidade deste trabalho (O2). Com essa análise, foi possível definir um modelo de avaliação, tendo em mente diferentes tipos de avaliação, a rubrica do curso, e a importância do feedback (O3). O próximo passo foi automatizar a avaliação desse modelo desenvolvido, usando, entre outras técnicas, deep learning (O4). Por fim, a avaliação automatizada para o curso Machine Learning para Todos! foi integrada no sistema CodeMaster (O5).

Considerando os resultados do levantamento do estado da arte, o possível impacto deste trabalho é grande. Um recurso de aprendizado de Machine Learning com foco para alunos no final do ensino fundamental com avaliação automatizada é raro, especialmente em português. Por isso, este é um passo importante na base da AI Literacy para os jovens estudantes brasileiros.

Como trabalhos futuros, alguns aspectos da avaliação podem ser melhorados, adições podem ser feitas, para ter-se um sistema mais completo e robusto. Pode-se implementar os dois critérios da rubrica que foram retirados, criando métodos para a detecção da limpeza, qualidade e relevância dos dados, e garantir que imagens impróprias não sejam incluídas no treinamento do modelo. Além disso, seria benéfico a adição de um método de avaliação em massa, pelo qual um professor envia um arquivo com múltiplas respostas de seus alunos, e o sistema retorna a nota de cada um.

## Referências

ALA-MUTKA, K. M. A Survey of Automated Assessment Approaches for Programming Assignments. *Computer Science Education*, v. 15, n. 2, p. 83-102, 2005.

AMERSHI, S. et al. Software Engineering for Machine Learning: A Case Study. *Proc. of the IEEE/ACM 41st International Conference on Software Engineering: Software Engineering in Practice*, Montreal, Quebec, Canada, 2019.

ANDRADE, D. F.; TAVARES, H. R.; VALLE, R. C. *Teoria da Resposta ao Item: Conceitos e Aplicações*. São Paulo: ABE — Associação Brasileira de Estatística, 4º SINAPE, 2000.

ANGULAR. The modern web developer's platform. 2021. Disponível em: <<https://angular.io>>. Acesso em: Dezembro de 2021

AI4K12. AI4K12 Big Idea #3 Progression Chart - Working Draft of Big Ideia 3. 19 de Novembro de 2020. Disponível em: <[https://drive.google.com/file/d/1QL6I\\_I5cdNTVnYBIZ3\\_Lxur2DgFjmG\\_d/view](https://drive.google.com/file/d/1QL6I_I5cdNTVnYBIZ3_Lxur2DgFjmG_d/view)>. Acesso em: Dezembro de 2020

ALVES, N. C. *CodeMaster: Um Modelo de Avaliação do Pensamento Computacional na Educação Básica através da Análise de Código de Linguagem de Programação Visual*. Dissertação (Programa de Pós-Graduação em Ciência da Computação (PGCC)) – Universidade Federal de Santa Catarina, 2019

BRASIL. Base Nacional Comum Curricular (BNCC). *Educação é a Base*. Brasília, MEC/CONSED/UNDIME, 2017. Disponível em: <[http://basenacionalcomum.mec.gov.br/images/BNCC\\_publicacao.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_publicacao.pdf)>. Acesso em março de 2021.

BLOOM, B.S. et al. *Taxonomy of educational objectives*. Vol. 1: Cognitive domain. New York: McKay, v. 20, p. 24, 1956.

BRANCH, R. M. *Instructional Design: The ADDIE Approach*. New York: Springer, 2009.

BNE. Vagas de Emprego na Área de Informática, 2020. Disponível em: <<https://www.bne.com.br/vagas-de-emprego-na-area-de-informatica?Area=Inform%C3%A1tica&Sort=0&Page=1>>. Acesso em: Outubro 2020.

BRASSCOM. Índice BRASSCOM de Convergência Digital, 2012. Disponível em: <[https://brasscom.org.br/wp-content/uploads/2017/08/ibcd\\_2012.pdf](https://brasscom.org.br/wp-content/uploads/2017/08/ibcd_2012.pdf)>. Acesso em: Setembro de 2020.



- BRASSCOM. Relatório Setorial, 2019. Disponível em: <<https://brasscom.org.br/relatorio-setorial-de-tic-2019/>>. Acesso em: Setembro de 2020.
- CARVALHO, L. et al. Detecção precoce de evasão em cursos de graduação presencial em Computação: um estudo preliminar. Anais do Workshop sobre Educação em Computação, SBC, 2019
- CARPENTER, G. A.; GROSSBERG, S. The ART of adaptive pattern recognition by a self-organizing neural network. Computer, v. 21, n. 3, p. 77–88, 1988.
- CLICK4IT. General Guidelines for developing assessments, 2013. Disponível em: <[http://www.click4it.org/images/e/e1/Toolkit\\_Assessment.pdf](http://www.click4it.org/images/e/e1/Toolkit_Assessment.pdf)>. Acesso em: 01 de Dezembro de 2020
- CME. RESOLUÇÃO CME Nº02/2011. Conselho Municipal de Educação de Florianópolis, 2011.
- COMPUTAÇÃO NA ESCOLA, 2020. Disponível em: <<https://cursos.computacaonaescola.ufsc.br>>. Acesso em: Setembro 2020
- CORDEIRO, A. M. et al. Revisão sistemática: uma revisão narrativa. Revista do Colégio Brasileiro de Cirurgiões, v. 34, n. 6, p. 428-431, 2007.
- CORTINA, J. M. What is coefficient alpha? An examination of theory and applications. Journal of Applied Psychology. v. 78, p. 98-104, 1993.
- CRONBACH, L. J. Coefficient alpha and the internal structure of tests. Psychometrika, v. 16, n. 3. p. 297–334, 1951.
- CSTA. ACM. CSTA K–12 Computer Science Standards, Revised 2017, 2017. Disponível em: <<https://www.doe.k12.de.us/cms/lib/DE01922744/Centricity/Domain/176/CSTA%20Computer%20Science%20Standards%20Revised%202017.pdf>>. Acesso em: Setembro de 2020.
- DEMETRIO, M. F.. Desenvolvimento de um analisador e avaliador de código de App Inventor para ensino de computação. Trabalho de conclusão de curso apresentado como parte dos requisitos para obtenção do grau de Bacharel em Ciências Da Computação, UFSC, Florianópolis, 2017.
- ESTÚDIO NSC. Diário Catarinense. Mais de 30 mil estudantes de Florianópolis receberão chips com internet para auxílio nos estudos, 25 de Fevereiro de 2021. Disponível em: <<https://www.nscototal.com.br/noticias/mais-de-30-mil-estudantes-de-florianopolis-receberao-chips-com-internet-para-auxilio-nos>>. Acesso em: Abril 2021
- EVANGELISTA, I.; BLESIO, G.; BENATTI, E. Why Are We Not Teaching Machine Learning at High School? A Proposal. Proc. of the World Engineering Education Forum - Global Engineering Deans Council, Albuquerque, NM, USA, 2018.
- FERREIRA, M. N. F.. Um modelo para uma unidade de ensino sobre princípios do design visual de interfaces de app: um estudo de caso. Dissertação de mestrado, UFSC, Florianópolis, 2020
- FLASK. Web development, one drop at a time. 2021. Disponível em: <<https://flask.palletsprojects.com/en/2.0.x/>>. Acesso em: Dezembro de 2021
- GARCIA, A. B. Avaliação de aprendizagem de conceitos de machine learning nos anos finais do ensino fundamental. Dissertação do Programa de Pós-Graduação em Métodos e Gestão em Avaliação – PPGMGA, UFSC, Florianópolis, 2022.
- GLORFELD, L. W. An improvement on Horn's parallel analysis methodology for selecting the correct number of factors to retain. Educational and Psychological Measurement, v. 55, n. 3, p. 377-393, 1995.
- GONÇALVES, R. Q. Ensino de Gerenciamento de Projetos de Software Mediado por Ferramentas. 2017. Tese de Doutorado - Universidade Federal de Santa Catarina.
- GOODFELLOW, I. et al. Generative Adversarial Nets. Université de Montréal, 2014.
- GRESSE VON WANGENHEIM, C.; et al. CodeMaster – Automatic Assessment and Grading of App Inventor and Snap! Programs. Informatics in Education, 17(1), 2018, 117-150.

GRESSE VON WANGENHEIM, C.; MARQUES, L. S.; HAUCK, J. C. R. Machine Learning for All – Introducing Machine Learning in K-12. SocArXiv, 2020.

HINTON, G., SEJNOWSKI, T. J. Unsupervised Learning: Foundations of Neural Computation. Cambridge: The MIT Press, 1999.

INEP. Sinopses Estatísticas da Educação Superior: graduação. Graduação. 2019. Disponível em: <http://inep.gov.br/sinopses-estatisticas-da-educacao-superior>. Acesso em: março de 2021.

JAKHAR, D.; KAUR, I. Artificial intelligence, machine learning and deep learning: definitions and differences. Clinical and Experimental Dermatology, v. 45, p. 131–132, jan. 2020.

JING, M. South China Morning Post. China looks to school kids to win the global AI race, 3 de Maio de 2018. Disponível em: <https://www.scmp.com/tech/china-tech/article/2144396/china-looks-school-kids-win-global-ai-race>. Acesso em: Outubro 2020

KAY, D. et al. . Automated grading assistance for student programs. Proc. of the 25th SIGCSE technical symposium on Computer science education, Phoenix, Arizona, EUA, p. 381 – 382, 1994.

KULHAVY, R., & STOCK, W. Feedback in written instruction: The place of response certitude. Educational Psychology Review, v. 1, n. 4, pp. 279–308, 1989.

LEE, I. et al. Computational thinking for youth in practice. ACM Inroads, v. 2, n. 1, p. 32–37, 2011.

LIMESURVEY. Turn questions into answers. Disponível em: <https://www.limesurvey.org>. Acesso em: Janeiro de 2022

LINKEDIN. Profissões Emergentes 2020, 2020. Disponível em: [https://business.linkedin.com/content/dam/me/business/en-us/talent-solutions/emerging-jobs-report/Emerging\\_Jobs\\_Report\\_Brazil.pdf](https://business.linkedin.com/content/dam/me/business/en-us/talent-solutions/emerging-jobs-report/Emerging_Jobs_Report_Brazil.pdf). Acesso em: Outubro 2020.

LONG, D.; MAGERKO, B. What is AI Literacy? Competencies and Design Considerations. Proc. of the CHI Conference on Human Factors in Computing Systems., Honolulu, HI, USA, 2020.

SALVADOR, G., GRESSE VON WANGENHEIM, C., RAUBER, M., GARCIA, A., BORGATTO, A. F. Avaliação de Aprendizagem de Machine Learning na Educação Básica: Um Mapeamento da Literatura. Anais do 29º WEI – Workshop sobre Educação em Computação, online, 2021

SCHMIDHUBER, J. Deep learning in neural networks: An overview. Neural Networks, v. 61, p. 85–117, 2015.

MARQUES, L. S. Ensino de Machine Learning na Educação Básica: um Mapeamento Sistemático do Estado da Arte. SBIE, Porto Alegre, RS, Brasil, p. 278-287 , 2021.

MARQUES, L. S.; GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R. Teaching Machine Learning in School: A Systematic Mapping of the State of the Art. Informatics in Education, v. 19, n. 2, p. 283–321, 2020.

MARQUES, L. S. GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R. Introdução à Machine Learning- Desenvolvimento de uma unidade instrucional para ensino de ML na educação básica. Relatório técnico ISSN 2236-5281 v. 1 n. 1, GQS/INCoD/INF/UFSC, 2020.

MEC, Base Nacional Comum Curricular, 2018. Disponível em: [http://basenacionalcomum.mec.gov.br/images/BNCC\\_EI\\_EF\\_110518\\_versaofinal\\_site.pdf](http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf). Acesso em: Setembro de 2020.

MYSQL, 2021. Disponível em: <https://www.mysql.com/>. Acesso em: dezembro de 2021.

NODEJS, Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. 2021. Disponível em: <https://nodejs.org/en/>. Acesso em, Dezembro de 2021

NOTAI.TECH, NudeNet: Neural Nets for Nudity Classification, Detection and selective censoring, 2019. Disponível em: <https://github.com/notAI-tech/NudeNet>. Acesso em: Setembro de 2021.

O'BRIEN C., The Mercury News, Get ready for the decade of gamification, 21 de Outubro de 2010. Disponível em: <<https://www.mercurynews.com/2010/10/21/obrien-get-ready-for-the-decade-of-gamification/>>. Acesso em: Janeiro de 2021

PINTO, N. et al. Quando a Aluna se Torna a Mestre: Um Relato da Experiência de Alunas de Graduação Aplicando Dinâmicas de Ensino de Computação para Alunas de Ensino Médio. Anais do Workshop sobre Educação em Computação, SBC, 2020.

POWERS, D. Evaluation: From Precision, Recall and F-Factor to ROC, Informedness, Markedness & Correlation. p. 25, 2008

RADHAKRISHNAN, P. What are Hyperparameters ? and How to tune the Hyperparameters in a Deep Neural Network?. Disponível em: <<https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a>>. Acesso em: Dezembro de 2020.

RODRIGUEZ, A. Restful web services: The basics. IBM developerWorks, 2008.

RUSSELL, S. J., NORVIG, P. Artificial Intelligence: A modern Approach. 3ª edição. New Jersey: Prentice Hall, 2010

SBC, Diretrizes para ensino de Computação na Educação Básica, 2019. Disponível em: <<https://www.sbc.org.br/documentos-da-sbc/send/203-educacao-basica/1220-bncc-em-itinerario-informativo-computacao-2>>. Acesso em: Dezembro de 2020.

SOFTEX. Relatório Anual, 2014. Disponível em: <<https://softex.br/booksoftex/>>. Acesso em: Setembro de 2020.

SOUZA, A. C. DE et al. Propriedades psicométricas na avaliação de instrumentos: avaliação da confiabilidade e da validade. Epidemiologia e Serviços de Saúde, v. 26, n. 3, p. 649–659, 2017.

SPRING, Spring makes Java simple. Modern. Productive. Reactive. Cloud-ready. 2021. Disponível em: <<https://spring.io/projects/spring-boot>>. Acesso em: Dezembro de 2021

STEGEMAN, M.; BARENSEN, E.; SMETSERS, S. Designing a rubric for feedback on code quality in programming courses. Proc. of the 16th Koli Calling International Conference on Computing Education Research, p. 160-164, 2016.

TACKE, O. SNORDIAN's H5PxAPIkatchu, 2017. Disponível em: <<https://wordpress.org/plugins/h5pxapikatchu/>>. Acesso em: Dezembro 2020

TEACHABLE MACHINE. Train a computer to recognize your own images, sounds, & poses. 2021. Disponível em: <<https://teachablemachine.withgoogle.com>>. Acesso em: Janeiro de 2022

TISSENBAUM, M. et al. From computational thinking to computational action. Communications of the ACM, v. 62, n. 3, p. 34–36, 2019.

TOURETZKY, D. S. et al. K-12 Guidelines for Artificial Intelligence: What Students Should Know. Proc. of the ISTE Conference, Philadelphia, PA, USA. 2019

VALUEVA, M. V. et al. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. Mathematics and Computers in Simulation, v. 177, p. 232–243, 2020.

WOOLFOLK, A. Educational psychology. 10ª edição. Boston: Pearson/Allyn and Bacon, 2007.

ZHU, X.; GOLDBERG, A. B. Introduction to Semi-Supervised Learning. Synthesis Lectures on Artificial Intelligence and Machine Learning, v. 3, n. 1, p. 1–130, 2009.