

UNIVERSIDADE FEDERAL DE SANTA CATARINA  
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA

PEDRO PAULO PEREIRA JUNIOR

**DESENVOLVIMENTO DE UM MODELO DE AVALIAÇÃO DE APRENDIZAGEM  
DE *MACHINE LEARNING* NO ENSINO MÉDIO  
VOLTADO A DETECÇÃO DE OBJETOS**

FLORIANÓPOLIS

2022

PEDRO PAULO PEREIRA JUNIOR

**DESENVOLVIMENTO DE UM MODELO DE AVALIAÇÃO DE APRENDIZAGEM  
DE *MACHINE LEARNING* NO ENSINO MÉDIO VOLTADO A DETECÇÃO DE  
OBJETOS**

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

*Machine Learning* (ML) é uma tecnologia que está crescendo e se tornando cada vez mais popular dentre as tecnologias atuais. Por isso é muito importante que as pessoas que a utilizam entendam, mesmo que de forma superficial, como essas tecnologias funcionam. Como ML está integrado em vários aspectos da vida das pessoas, é importante que o ensino desses conceitos seja iniciado já na Educação Básica. Nesse contexto, diversas iniciativas estão sendo criadas, porém a maioria não fornece um modelo de avaliação de aprendizagem para ajudar na aplicação desses conceitos, atualmente sendo feita de maneira manual, tomando tempo, esforço e preparação por parte dos instrutores, dificultando a adoção do ensino de ML nas escolas. Por isso, o objetivo deste trabalho é desenvolver um modelo de avaliação de aprendizagem de ML no ensino médio, voltado à detecção de objetos, automatizando esse processo por meio de técnicas de *Deep Learning*. O modelo de avaliação desenvolvido é integrado em um Jupyter Notebook na plataforma Google Colab no contexto do curso desenvolvido por Gonçalves (2022). Com isto, este trabalho traz uma solução, que consiste em um modelo de avaliação que permite avaliar a aprendizagem de ML, facilitando todo o processo de ensino de ML, contribuindo dessa maneira ao ensino de ML nas escolas brasileiras.

**Palavras chave:** *Machine Learning*, Detecção de objetos, Educação Básica,, Avaliação, Deep Learning, YOLO.

## **ABSTRACT**

Machine Learning (ML) is a technology that is growing and becoming more and more popular among the current ones. That's why it's very important that people understand, even if superficially, how these technologies work. As ML is integrated into various aspects of people's lives, it is important that these concepts start already in Basic Education. In this context, several initiatives are being created, but most do not offer a learning assessment model to help in the application of these concepts, currently being done manually, taking time, effort and preparation on the part of the instructors, making it difficult to adopt teaching of ML in schools. Therefore, the objective of this work is to develop a model for evaluating ML learning in high school, aimed at object detection, automating this process through Deep Learning techniques. The evaluation model developed is integrated into a Jupyter Notebook on the Google Colab platform in the context of the course developed by Gonçalves (2022). With this, this work brings a solution, which consists of an evaluation model that allows evaluating ML learning, facilitating the entire ML teaching process, thus contributing to ML teaching in Brazilian schools.

**Palavras chave:** Machine Learning, Object Detection, Basic Education, Avaliação, Assessment, Deep Learning, YOLO.

## LISTA DE FIGURAS

Figura 1: IA, ML e DL (BERCHANE, 2018)	10
Figura 2: Rede Neural Simples e Rede Neural Profunda (IBMa, 2021).	12
Figura 3: Redes Neurais Profundas e Redes Neurais Convolucionais (STAMFORD, 2021).	12
Figura 4: Dimensões de uma entrada de imagem na rede convolucional. (ALVES, 2018).	13
Figura 5: Processo de maxpooling. (ALVES, 2018).	13
Figura 6: Modelos de redes convolucionais, classificadas por ano de publicação. (VON WANGENHEIM, 2021).	14
Figura 7: Processo de detecção de objetos em imagem utilizando YOLO (GANDHI, 2018)	15
Figura 8: tamanhos de modelos Yolov5 (YOLOv5, 2022).	16
Figura 9: Processos de planejamento e desenvolvimento de modelo de ML (GRESSE VON WANGENHEIM e VON WANGENHEIM).	16
Figura 10: Exemplo de campo de texto em um Jupyter Notebook no ambiente do Google Colab	18
Figura 11: Comparação dos resultados com objetivos do processo de avaliação de aprendizagem (GRESSE VON WANGENHEIM, 2021).	25
Figura 12: Níveis de aprendizagem da taxonomia de Bloom. Adaptado de (JOHNSON, 2006).	25
Figura 13: Arquitetura da automação	47
Figura 14: Exemplo de widget interativo da questão de interpretação de precisão média	48
Figura 15: Escala de “ninjas-robôs” conforme a pontuação	54
Figura 16: Exemplo de apresentação da avaliação para a nota	55

## LISTA DE TABELAS

Tabela 1: Descrição das fases e etapas de processos de ML.	20
Tabela 2: Resumo dos objetivos de aprendizagem da terceira grande ideia (SALVADOR et al., 2021a) a partir de (TOURETZKY et al., 2019)	24
Tabela 3: Competências necessárias para AI Literacy (SALVADOR et al., 2021a) a partir de (TOURETZKY et al., 2019).	26
Tabela 4: Níveis de aprendizagem e seus respectivos tipos de avaliação. Adaptado de (RAGUPATHI, 2020).	29
Tabela 5: Termos de busca.	32
Tabela 6: Strings de buscas utilizadas nas diferentes bases de dados (SALVADOR et al., 2021b).	32
Tabela 7: Número de artigos identificados por base de dados	33
Tabela 8: Informações extraídas. Adaptado de (SALVADOR et al., 2021b).	34
Tabela 9: Avaliações de aprendizagem de Machine Learning no Ensino Médio	35
Tabela 10: Características das avaliações de aprendizagem	36
Tabela 11: Conceitos de Machine Learning avaliados	37
Tabela 12: Uso da Internet em atividades escolares de alunos de escolas urbanas (CETIC, 2019a).	40
Tabela 13: Computadores existentes na cada dos estudantes. (CETIC, 2019b)	40
Tabela 14: Professores de escolas urbanas que participaram de curso de formação continuada sobre o uso de computador e internet em atividades de ensino (CETIC, 2019c).	41
Tabela 15: Objetos a ser detectados	42
Tabela 16: Objetivos de aprendizagem de Machine Learning para detecção de objetos no Ensino Médio adaptado a partir de (GONÇALVES, 2022).	43
Tabela 17: Rubrica de avaliação e proposta de automação da avaliação. Adaptado de (SALVADOR et al., 2021b).	44
Tabela 18: Definição do modelo de tarefa.	46
Tabela 19: Formato do Cartão de Modelo.	47
Tabela 20: Requisitos funcionais. Adaptado de (SALVADOR et al., 2021b).	48
Tabela 21: Requisitos não funcionais. Adaptado de (SALVADOR et al., 2021b).	49
Tabela 22: Cartão de Modelo do modelo.	59

## **LISTA DE ABREVIATURAS E SIGLAS**

IA Inteligência Artificial

ML *Machine Learning*

BNCC Base Nacional Comum Curricular

DL *Deep Learning*

TL *Transfer Learning*

ECD *Evidence-Centered Assessment Design*

YOLO *You Only Look Once*

DNN *Deep Neural Networks*

CNN *Convolutional Neural Networks*

## SUMÁRIO

<b>1. INTRODUÇÃO</b>	<b>9</b>
<b>1.1 CONTEXTUALIZAÇÃO</b>	<b>9</b>
<b>1.2 OBJETIVOS</b>	<b>10</b>
<b>1.3 METODOLOGIA DE PESQUISA</b>	<b>11</b>
<b>1.4 ESTRUTURA DO DOCUMENTO</b>	<b>12</b>
<b>2. FUNDAMENTAÇÃO TEÓRICA</b>	<b>13</b>
<b>2.1 MACHINE LEARNING</b>	<b>13</b>
2.1.1 Processo de desenvolvimento de um modelo de detecção de objetos	19
2.1.2 Jupyter Notebook e Google Colab	20
<b>2.2 ENSINO DE MACHINE LEARNING NO ENSINO MÉDIO</b>	<b>21</b>
<b>2.3 AVALIAÇÃO DA APRENDIZAGEM</b>	<b>27</b>
<b>3. ESTADO DA ARTE</b>	<b>31</b>
<b>3.1 DEFINIÇÃO DO PROTOCOLO DE BUSCA</b>	<b>31</b>
<b>3.2 EXECUÇÃO DA BUSCA</b>	<b>33</b>
<b>3.3 ANÁLISE DOS RESULTADOS</b>	<b>34</b>
<b>3.4 DISCUSSÃO</b>	<b>37</b>
<b>4. MODELO DE AVALIAÇÃO</b>	<b>39</b>
<b>4.1 ANÁLISE E MODELAGEM DO DOMÍNIO</b>	<b>39</b>
<b>4.2 DESENVOLVIMENTO DO FRAMEWORK CONCEITUAL</b>	<b>42</b>
4.2.1 Modelo de estudante	42
4.2.2 Modelo de Evidência	43
4.2.3 Modelo de Tarefa	45
4.2.4 Modelo de documentação	46
<b>5 DESENVOLVIMENTO DA AUTOMAÇÃO</b>	<b>48</b>
<b>5.1 ANÁLISE DE REQUISITOS</b>	<b>48</b>
<b>5.2 ARQUITETURA</b>	<b>49</b>
<b>5.3 IMPLEMENTAÇÃO</b>	<b>50</b>
5.3.1 Critérios	51
5.3.2 Uso da automação	55
5.3.3 Apresentação da avaliação	56
5.3.4 Geração do relatório do modelo treinado	57
5.3.5 Disponibilização da automação	61
5.3.6 Tutorial de utilização	62
<b>6 CONCLUSÃO</b>	<b>64</b>
<b>7 REFERÊNCIAS</b>	<b>65</b>



# 1. INTRODUÇÃO

## 1.1 CONTEXTUALIZAÇÃO

Com o passar dos anos, a utilização de sistemas computacionais no dia a dia das pessoas vem crescendo cada vez mais. Muitas atividades hoje são realizadas por meio de *softwares* que são desenvolvidos para ajudar, facilitando essas tarefas. Com todo esse impacto e influência da tecnologia é cada vez mais notável a necessidade de que todos tenham conhecimento sobre essas tecnologias, não apenas os profissionais dessas áreas. Inteligência Artificial (*Artificial Intelligence - IA*) e Aprendizado de Máquina (*Machine Learning - ML*) são áreas da computação que estão muito presentes na sociedade atual (IEEE, 2020). Aprendizado de máquina é uma área da Inteligência artificial que estuda tecnologias que permitem que máquinas possam aprender a partir de dados já processados, sem terem que ser programadas especificamente (JAKHAR; KAUR, 2020). ML é uma tecnologia muito usada em sistemas de buscas, em mecanismos de recomendações, identificação de spam em e-mails eletrônicos, entre outros usos que fazem parte da vida de muitas pessoas.

A área de ML está crescendo cada vez mais. Por exemplo, de acordo com a pesquisa feita pela IEEE, 32% dos entrevistados disseram que IA e ML serão as principais tecnologias no ano de 2021 (IEEE, 2020). É possível observar ainda que, apesar da crise econômica atual que o país vive, durante a pandemia da COVID-19, a quantidade de vagas referentes a tecnologia chega a ser de aproximadamente 30 mil (BNE, 2021). E especificamente, “Especialista em Inteligência Artificial” ficou entre as 15 profissões emergentes do Brasil do LinkedIn, crescendo 73% em relação ao ano anterior (LINKEDIN, 2020).

Porém, apesar de todas essas oportunidades e incentivo à tecnologia, o ensino desses temas, ou até mesmo de conceitos básicos de computação, nas escolas brasileiras são escassos. Dada essa oportunidade de mercado referente a IA/ML, e a falta de profissionais na área, por exemplo, o governo da China vem implementando o ensino dessas áreas no ensino médio, levando os estudantes a aprender uma compreensão básica de ML (LIO, 2020). Existem também outras iniciativas de outros países para a implementação do ensino de IA/ML na Educação Básica. Com esta intenção, o AI4K12 está criando diretrizes de currículo *K-12 Guidelines for Artificial Intelligence* (TOURETZKY, 2019a). Alinhado a estas diretrizes, diversos cursos de IA/ML estão sendo

criados para a Educação Básica, usando diversas técnicas de implementação de *Machine Learning* (MARQUES et al., 2020). Neste contexto, a iniciativa Computação na Escola visa introduzir a aprendizagem de conceitos de ML no contexto brasileiro, seguindo as diretrizes como *K-12 Guidelines for Artificial Intelligence* (TOURETZKY, 2019a). E, com este objetivo, são desenvolvidos cursos on-line para ensinar conceitos básicos de ML em relação a várias tarefas, como a detecção de objetos (GONÇALVES, 2022). Esses cursos abordam todo o processo de ML da análise de requisitos até a implantação e questões éticas e dos impactos do ML na sociedade.

Devido sua importância no processo de ensino para auxiliar na progressão da aprendizagem do aluno, precisa-se avaliar seus resultados de aprendizagem e fornecer um *feedback* informativo e construtivo ao estudante. Porém observa-se que atualmente ainda faltam pesquisas sobre como avaliar a aprendizagem de ML do aluno na Educação Básica. Além disso, levando em consideração as características e restrições típicas na educação básica, necessita-se a automatização deste tipo de avaliação para aumentar a consistência das avaliações além de minimizar o esforço e tempo das avaliações do ponto de vista do professor.

Portanto, notando a atual falta de modelos de avaliação para sistematicamente avaliar o aprendizado dos conceitos de ML, este projeto busca responder a seguinte pergunta de pesquisa: Como automatizar a avaliação da aprendizagem de ML na detecção de objetos no Ensino Médio?

## **1.2 OBJETIVOS**

### **Objetivo geral**

O objetivo geral deste trabalho é desenvolver um modelo de avaliação da aprendizagem de ML voltado à detecção de objetos no Ensino Médio. São adotadas técnicas de *deep learning* para automatizar a avaliação da aprendizagem. O modelo desenvolvido é integrado em um Jupyter Notebook sendo executado na plataforma Google Colab adicionando a avaliação automatizada da aprendizagem de ML. O modelo de aprendizagem é baseado nas diretrizes de currículo de ensino de IA na Educação Básica (TOURETZKY et al., 2019a).

### **Objetivos Específicos**

O1. Analisar a fundamentação teórica sobre ensino de ML no Ensino Médio, avaliação da aprendizagem e *machine learning*.

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

O3. Definir um modelo de avaliação (tipos de avaliação, rubrica e *feedback*).

O4. Desenvolver uma biblioteca para automatizar a avaliação utilizando *deep learning*.

### **1.3 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 *machine learning*

#### **Etapa 2 – Estado da arte**

É realizada uma síntese da literatura seguindo o processo proposto por Petersen et al. (2008) visando identificar e analisar modelos de avaliação da aprendizagem de ML no Ensino Médio. 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**

Nessa etapa é seguida a metodologia de design instrucional ADDIE (Branch, 2009) para realizar o desenvolvimento do curso como todo, utilizando como metodologia para o desenvolvimento da avaliação o *Evidence-Centered Assessment Design* (MISLEVY et al., 2003). Todo o processo de desenvolvimento é realizado em três etapas: a primeira é a análise do contexto do público alvo, em seguida é definido o desempenho que deseja ser

alcançado definindo um plano de avaliação, e por fim, é desenvolvido todos os materiais de avaliação (rubrica).

A3.1 – Analisar e modelar o domínio;

A3.2 – Desenvolver o framework conceitual;

A3.3 – Projetar a implementação da avaliação.

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

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

A4.1 - Implementar um modelo de detecção de objetos com Yolo5s?

A4.2 - Iteração da automação da rubrica, usando em alguns casos *deep learning*

A4.3 - Execução e apresentação dos resultados no jupyter

### **1.4 ESTRUTURA DO DOCUMENTO**

A presente monografia é organizada primeiramente com o capítulo de fundamentação teórica, abordando os temas relevantes, como *Machine Learning*, Ensino de ML e Avaliação da Aprendizagem. O terceiro capítulo tem como objetivo levantar o estado da arte, sendo buscado diversos projetos, a partir do protocolo de busca definido, que possam agregar no presente trabalho. No quarto capítulo é realizada a criação do modelo de avaliação e o desenvolvimento do modelo conceitual. No quinto capítulo é implementada a automação da avaliação da aprendizagem, são definidos também os requisitos funcionais e não funcionais, é apresentada a arquitetura de toda implementação realizada, bem como a implementação em um pseudo código. O último capítulo é referente às conclusões relacionadas a esta monografia e propostos trabalhos futuros.

## 2. FUNDAMENTAÇÃO TEÓRICA

### 2.1 MACHINE LEARNING

A inteligência era comumente estudada por áreas relacionadas a ciências humanas, muito ligado ao saber e aspectos comportamentais dos seres humanos. Porém com o passar dos anos, diversos pesquisadores se dedicaram a estudar esse tema em outras áreas do conhecimento, como engenharia, psicologia, neurologia, computação, entre outros. Apesar do estudo da inteligência estar muito atrelado a ciências humanas, hoje existem diversos outros campos de estudos sobre esse tema. No ramo da computação, a ciência que procura estudar e compreender como funciona essa teoria aplicada às máquinas, é conhecida como Inteligência Artificial (IA).

IA refere-se à área da ciência da computação orientada ao entendimento, construção e validação de sistemas inteligentes, isto é, que exibem, de alguma forma, características associadas ao que chamamos de inteligência (RICH e KNIGHT, 1994). Complementando essa ideia, a IA procura compreender a inteligência por meio de diversos campos de conhecimento, buscando simular a inteligência humana (LUGER, 2004). Como IA é um termo muito amplo, com o avanço da tecnologia, foram surgindo vários subcampos de estudos, como *Machine Learning* e *Deep Learning* (Figura 1), relacionados também à compreensão da inteligência humana por meio da computação (JEFFCOCK, 2018).

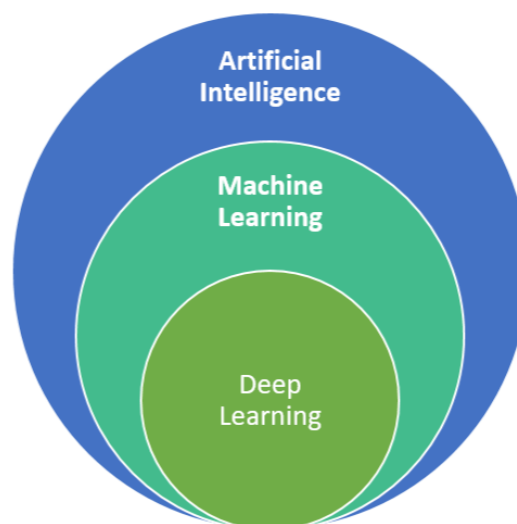


Figura 1: IA, ML e DL (BERCHANE, 2018).

*Machine learning* é um campo de estudo Inteligência Artificial que permite com que computadores tenham a habilidade de aprender sem ser explicitamente programados

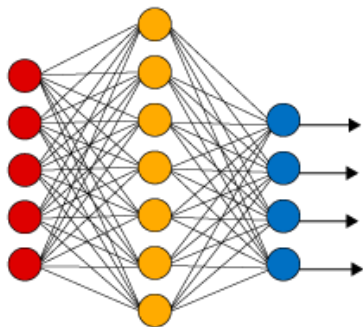
(SAMUEL, 2018). Significa que um único programa, uma vez criado, será capaz de aprender como fazer algumas atividades inteligentes sem necessidade de programação de programação. Isso contrasta com programas construídos com propósito, cujo comportamento é definido por heurísticas feitas à mão que explicitamente definem seu comportamento. Ou seja, em vez de escrevermos um código para um programa específico usando um conjunto de instruções para realizar uma tarefa, as máquinas poderão ser treinadas com um conjunto de dados e algoritmos que lhes dá a capacidade de aprender como executar uma tarefa (BERCHANE, 2018).

*Deep Learning* é um dos principais recursos para que o ML possa aprimorar a capacidade de reconhecer dados e gerar insights, principalmente ao levar em consideração uma grande base de dados (STEFANINI, 2019). Esses métodos melhoraram muito o estado da arte em detecção visual de objetos, reconhecimento de voz, sistemas de recomendações, etc (LECUN et al., 2015).

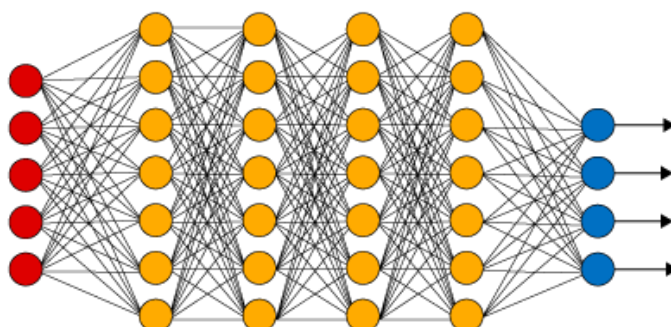
Uma Rede Neural (*Neural Network*) é um sistema que é projetado para modelar a forma como o cérebro realiza uma tarefa, onde são compostas por nós ou neurônios, por meio de ligações direcionadas, que propagam a ativação de nós anteriores, contendo em cada ligação seus respectivos pesos, que determina a força e sinal de conexão (RUSSEL; NORVIG, 2009). As redes neurais são geralmente divididas em camadas, no modelo mais básico de rede, temos uma camada de entrada de nós de fonte que se projeta sobre camadas ocultas intermediárias até uma camada de saída (HAYKIN, 2007).

Enquanto ML tem sua base desenvolvida em uma Rede Neural Simples, DL baseia-se em Rede Neurais Profundas (*Deep Neural Networks*), pois enquanto *NNs* trabalham com camadas mais lineares, *DNNs* conseguem trabalhar com centenas de camadas hierárquicas, possibilitando análises mais complexas com múltiplos níveis de abstração (MATLAB, 2017). Nessas redes neurais profundas, cada camada treina um conjunto de características que são geradas a partir das camadas antecessoras. E quanto mais características são processadas, mais os nós conseguem reconhecer características ainda mais complexas. Todo esse conjunto de camadas e nós que são interligados e processados em uma rede neural profunda é conhecido como hierarquia de características. Essa hierarquia de características, de complexidade e abstrações incrementais, que permite redes de *deep learning* sejam capazes de processar conjuntos de dados de alta dimensão com bilhões de parâmetros (PATHMIND, 2019).

### Simple Neural Network



### Deep Learning Neural Network



● Input Layer    ● Hidden Layer    ● Output Layer

Figura 2: Rede Neural Simples e Rede Neural Profunda (IBMa, 2021).

Entretanto, apesar dessas redes serem capazes de identificar estruturas em dados não-estruturados e não-categorizados, como fotos, mídia crua, texto, vídeo e gravações de áudios (PATHMIND, 2019), estudos indicam que para obtermos um bom treinamento nessas redes, precisa-se de aproximadamente 1000 imagens para cada classe a ser classificada (WARDEN, 2017). Porém, usando redes neurais convolucionais, juntamente com *Transfer Learning*, com modelos pré-treinados, conseguimos obter bons resultados utilizando poucos dados para treinamento (SHU, 2019).

Redes Neurais Convolucionais (*Convolutional Neural Networks*) são redes neurais utilizadas para classificação de imagens e detecção de objetos. Essa rede neural artificial recebe uma imagem e subsequentemente processa a mesma em camadas internas, nas quais consegue-se identificar e diferenciar objetos dentro da imagem (BORGES, 2021). Em uma camada convolucional, cada neurônio artificial é organizado espacialmente, onde cada entrada está sempre sujeita a uma operação de convolução, onde uma operação matemática de matrizes é aplicada. Essas operações de convolução podem ser entendidas como filtros no sinal de entrada, e com isso, ao invés das CNNs utilizarem filtros prontos, utilizam filtros que foram personalizados durante o processo de treinamento (BAULÉ et al., 2020).

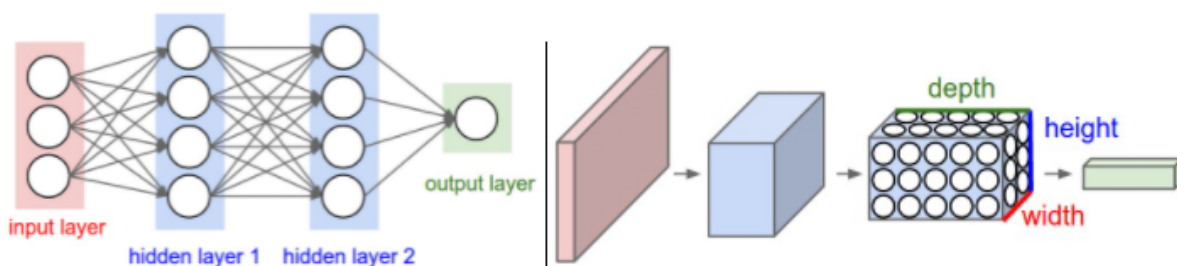


Figura 3: Redes Neurais Profundas e Redes Neurais Convolucionais (STAMFORD, 2021).

As CNNs tiram proveito do fato que as entradas consistem em imagens e restringem a arquitetura a uma forma mais sensível. A rede neural utiliza a largura e altura (de acordo com as dimensões da imagem) e profundidade, determinada pela quantidade de canais de cores. Em geral as imagens utilizam três canais, *RGB* (*Red-Green-Blue*), com os valores de cada pixel.

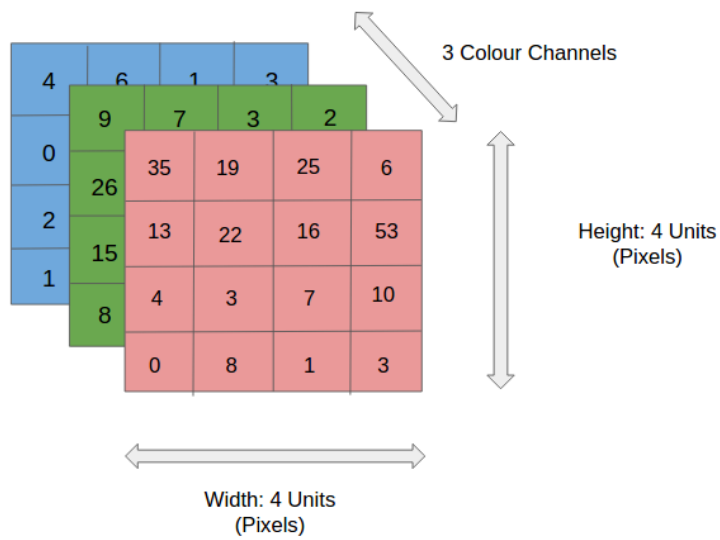


Figura 4: Dimensões de uma entrada de imagem na rede convolucional. (ALVES, 2018).

O filtro que é utilizado na operação de convolução é uma matriz que percorre a imagem, um pixel por processo, realizando a multiplicação e a soma dos seus valores com os valores do pixel da imagem. Além desse filtro, aplica-se uma camada de pooling, que serve para simplificar uma camada processada. A partir de uma área pré-definida, que percorre toda a camada anterior, é feito um *maxpooling*, ou seja, o maior número da unidade é passado para a saída. A figura 5 mostra um exemplo de um *maxpooling* sendo realizado.

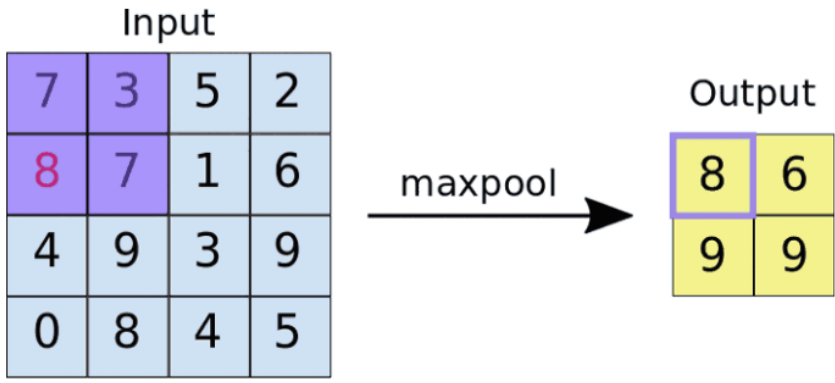


Figura 5: Processo de *maxpooling*. (ALVES, 2018).



Por meio de todo esse processo de convolução e *pooling*, consegue-se extrair as diferentes características contidas na imagem, tudo isso com um menor custo computacional. Essa saída da *CNNs*, e de qualquer outro algoritmo de ML, é nomeado modelo. Esses modelos gerados por algoritmos de ML tem como objetivo melhorar sua aprendizagem e acurácia durante o treinamento (BROWNLEE, 2020).

Com o passar dos anos, diversos modelos de redes neurais convolucionais foram surgindo. Normalmente a saída dessas redes neurais é um casco convexo retangular (*bounding box*), que indica a região onde possivelmente se encontra o objeto que se deseja detectar (VON WANGENHEIM, 2021). A Figura 6 mostra a evolução desses modelos, classificando-os por anos de publicação.

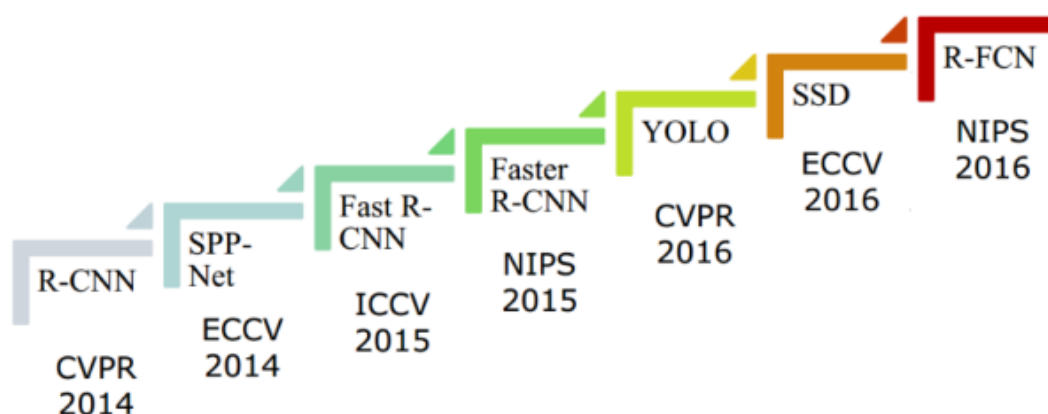


Figura 6: Modelos de redes convolucionais, classificadas por ano de publicação. (VON WANGENHEIM, 2021).

Entre esses vários modelos, existem os modelos que classificam regiões que estão associadas a possíveis extratores de características, e redes neurais convolucionais de disparo único para reconhecimento de objetos. Entre as que classificam regiões de possíveis extratores de características existe o modelo R-CNN (*Region Based Convolutional Neural Network*), que inicialmente identifica 2000 possíveis candidatos, usando técnicas simples de visão computacional, a partir disso cada candidato é reconstruído em uma imagem quadrada, com isso cada imagem é alimentada a uma rede neural convolucional que produz um vetor de recursos de 4096 dimensões como saída. Após isso, uma SVM (*Support Vector Machines*), faz a classificação dos vetores de características, produzindo duas saídas, a classificação, e um *offset*, que é utilizado para ajudar o *bounding box* (GANDHI, 2018).

Já entre os modelos que realizam um disparo único para reconhecimento de objetos, existe o YOLO (*You Only Look Once*), que é uma tecnologia que vem crescendo muito e

conseguindo ótimos resultados de performance (FARIA, 2021). YOLO é muito diferente dos outros algoritmos de detecção de objetos, pois ele precisa olhar apenas uma vez a imagem para enviar à rede neural. A entrada desse algoritmo é uma imagem, sendo dividida em um grid de  $S \times S$  células. Essas células são usadas para prever *bounding boxes*, na qual cada predição desses *bounding boxes* possui 5 valores:  $x$ ,  $y$ ,  $w$ ,  $h$  e *confiança*. As coordenadas que indicam o centro do *bounding box*, em relação à localização da célula, é representada pelos valores de  $x$  e  $y$ . Já as dimensões do *bounding box* são representadas pelos valores de  $w$  e  $h$  (VON WANGENHEIM, 2021).

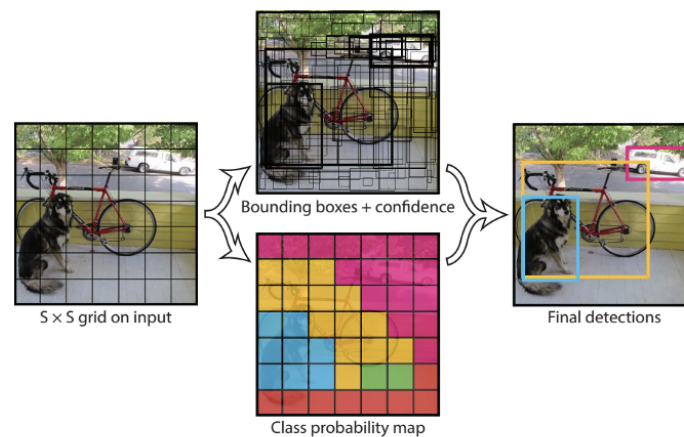


Figura 7: Processo de detecção de objetos em imagem utilizando YOLO (GANDHI, 2018)

YOLO possui diversas versões, a última e mais atual, e será utilizada no presente trabalho, é o Yolov5 (YOLOv5, 2022). Uma das vantagens do Yolov5 em relação às versões anteriores é a velocidade e precisão no momento do treinamento do modelo. Como o YOLOv5 é implementado inicialmente no PyTorch, ele se beneficia do ecossistema PyTorch estabelecido: o suporte é mais simples e a implantação é mais fácil. Além disso, como uma estrutura de pesquisa mais conhecida, iterar no YOLOv5 pode ser mais fácil para a comunidade de pesquisa mais ampla. Isso também simplifica a implantação em dispositivos móveis, pois o modelo pode ser compilado para ONNX e CoreML com facilidade. O lançamento do YOLOv5 inclui cinco tamanhos de modelos, a Figura 8 mostra as opções disponíveis.

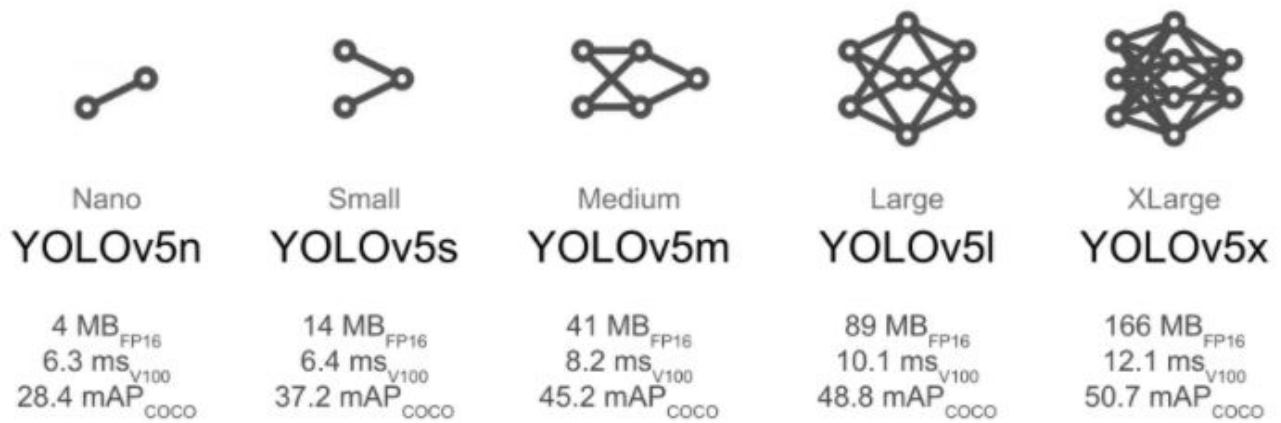


Figura 8: tamanhos de modelos Yolov5 (YOLOv5, 2022).

### 2.1.1 Processo de desenvolvimento de um modelo de detecção de objetos

Construir um aplicativo de ML é um processo iterativo que requer que os alunos executem uma sequência de etapas, como por exemplo apresentado na Figura 8 e Tabela 1, visando a tarefa de classificação de imagens (AMERSHI et al. 2019; FIEBRINK et al. 2018). A fim de visar o desenvolvimento de soluções inteligentes úteis, tal processo abrange não só o desenvolvimento do modelo de ML, mas também a identificação de necessidades, idealização, bem como a implantação do modelo de ML desenvolvido e seus testes como uma solução inteligente, por exemplo, como parte de um aplicativo móvel.

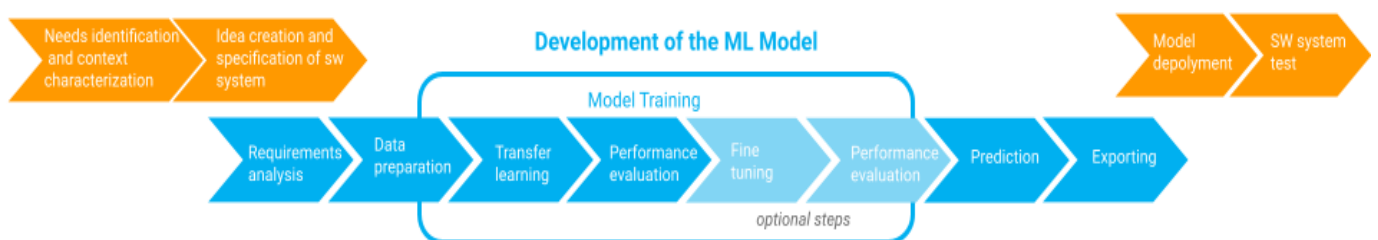


Figura 9. Processos de planejamento e desenvolvimento de modelo de ML (GRESSE VON WANGENHEIM e VON WANGENHEIM).

Fase	Etapa	Descrição	Saida(s)
Análise de requisitos do modelo de ML		Durante esta fase, o objetivo principal do modelo DL e seus recursos de destino são especificados. Isso também inclui a caracterização das entradas e saídas esperadas, especificando o problema.	Análise do modelo de <i>Deep Learning</i> Análise de implantação
Preparação dos dados	Coleta e limpeza dos dados	Nessa fase é realizada a coleta dos dados e definidas as técnicas de limpeza e revisão da qualidade dos dados. Os dados são limpos, o que envolve a remoção de registros imprecisos ou com ruído do conjunto de dados.	Dados limpos e padronizados, armazenados no HD ou google drive
	Rotulagem dos dados	Os dados precisam ser rotulados/categorizados, para que o algoritmo de ML consiga interpretar corretamente os dados. Para detecção de objetos, também são criados os <i>bounding boxes</i> .	Conjunto de dados rotulados salvo em diretórios separados (um para cada classe) no HD ou no Google Drive.
	Pré-processamento dos dados	Para aumentar o conjunto de dados, normalmente são realizadas novas inserções de dados, alterações e até adaptações. O tamanho do lote é definido. O conjunto de dados é normalmente dividido em um conjunto de treinamento para treinar o modelo, um conjunto de validação para selecionar o melhor candidato de todos os modelos e um conjunto de teste para realizar uma avaliação de desempenho imparcial do modelo escolhido em dados invisíveis. A verificação da qualidade dos dados é realizada a fim de verificar se os objetivos podem ser alcançados com a dada qualidade dos dados disponíveis (dados incorretos ou insuficientes, desequilibrados ou enviesados).	É feito o <i>upload</i> dos dados para a ferramenta de DL ( <i>jupyter</i> ).
Treinamento e evolução do modelo	Treinamento	Em seguida, um modelo é construído ou, mais tipicamente, escolhido a partir de modelos bem conhecidos que se mostraram eficazes em problemas ou domínios comparáveis. Definir rotinas de treinamento envolve definir a taxa de aprendizagem e o número de épocas, bem como métricas de desempenho.	Seleção da arquitetura do modelo e modelo pré-treinado; Especificação de parâmetros para o treinamento; Execução do treinamento.
	Avaliação de desempenho	O modelo treinado é avaliado com base nas informações sobre as métricas de desempenho específicas dependendo da tarefa específica de DL a ser alcançada, como precisão, erro médio absoluto, etc. A métrica mais usada para detecção de objetos é a Precisão Média (AP), que é definida como a precisão de detecção média em diferentes recuperações e geralmente é avaliada de uma maneira específica de classe de objeto. Para comparar as classes de objetos gerais de desempenho, o AP médio (mAP) calculado sobre todas as classes de objetos é comumente usado como métrica final de desempenho.	Análise de perda/taxa de erro Análise de métricas de desempenho específicas da tarefa
Predição		O modelo é testado com novos dados para obter uma aproximação de como o modelo se comporta no mundo real.	Teste com novos dados
Exportação do modelo		O modelo é exportado para permitir sua integração em um sistema de software	Modelo de DL exportado

Tabela 1. Descrição das fases e etapas de processos de ML.

## 2.1.2 Jupyter Notebook e Google Colab

Como aplicações de *Machine Learning* trabalham com grande volume e processamento de dados, é necessário que se tenha uma infraestrutura muito robusta e bem arquitetada para conseguir executar os treinamentos e algoritmos com um grande volume de dados (GOOGLEa, 2021). Neste contexto, o aplicativo web de código aberto Jupyter Notebook se faz muito útil, pois é uma ferramenta que permite que usuários criem equações, textos,

recursos de multimídia, algoritmos já documentados, e muito mais (ZOOM, 2021). Jupyter Notebook fornece um ambiente de desenvolvimento de algoritmos de ML muito interativo e fácil de usar que não funciona apenas como um ambiente de desenvolvimento, mas também como uma ferramenta de apresentação ou ferramenta educacional (ODSC, 2018).



Figura 10: Exemplo de campo de texto em um Jupyter Notebook no ambiente do Google Colab.

Um notebook Jupyter possui dois componentes, o *front-end* e o *back-end*. No *front-end* os usuários inserem o código de programação ou texto em “células” retangulares em uma página da web. O navegador então envia o código para um *kernel back-end* que executa o código e retorna os resultados. Muitos kernels Jupyter foram criados, suportando dezenas de linguagens de programação. Notebooks também podem ser executados na nuvem, por meio do projeto da empresa Google, na plataforma Google Colab. Além de todos esses serviços, é possível executar o Jupyter Notebook sem acesso à rede, executando diretamente em seu computador, possibilitando realizar trabalhos localmente (ODSC, 2018).

## 2.2 ENSINO DE *MACHINE LEARNING* NO ENSINO MÉDIO

O avanço das tecnologias impactou nos padrões de vida das pessoas, levando-as a desenvolverem novos hábitos. Atualmente a tecnologia é vista em diversos cenários, como no trabalho, em nosso dia a dia, na educação, entre outros. O uso da tecnologia como recurso de apoio para ensino de *Machine Learning* no Ensino Médio vem sendo muito discutido nos dias atuais (BENATTI, 2021).

O Ensino Médio é o último três níveis da Educação Básica. Essa etapa dura três anos, e tem como objetivo aprofundar os conhecimentos adquiridos nos anos anteriores, mas também preparar o estudante para o mercado de trabalho e o Ensino Superior (MEC, 2021a). Nesse contexto, visando mitigar esses problemas de falta de incentivo do uso da

tecnologia, e despertar o interesse dos jovens cada vez mais na tecnologia, surgiram diversas iniciativas nesse sentido ao redor do mundo (MARQUES et al., 2020).

Atualmente no Brasil esta área de conhecimento está sendo citada na Base Nacional Comum Curricular (BNCC), que é um documento de caráter normativo que tem como objetivo definir o conjunto de aprendizagens essenciais que todos os alunos devem desenvolver ao longo das etapas e modalidades da Educação Básica (MEC, 2021b). Atualmente aborda somente o ensino de pensamento computacional dentro da área de matemática.

A área de Matemática, no Ensino Fundamental, centra-se na compreensão de conceitos e procedimentos em seus diferentes campos e no desenvolvimento do **pensamento computacional**, visando à resolução e formulação de problemas em contextos diversos. (MEC; BNCC, 2018, p. 471).

Além do pensamento computacional, a BNCC também cita outros temas importantes, como a Inteligência Artificial. Na normativa, a BNCC trata Inteligência Artificial como um possível tema que os estudantes podem escolher dentro da área de Matemática e suas Tecnologias (MEC, 2021b).

A tecnologia de Inteligência Artificial vem crescendo, e se tornando muito importante para a sociedade e indústria. Com isso alguns países estão procurando inserir esse tema já no Ensino Médio para despertar o interesse de seus estudantes. Na China, por exemplo, está incorporando em seus currículos o ensino da Inteligência Artificial nas escolas no Ensino Médio (YOO, 2020).

É nesse contexto mundial que algumas diretrizes curriculares surgiram, por exemplo o CSTA K-12 Computer Science Framework (CSTA, 2016), que busca padronizar e fornecer um guia para quais conteúdos ligados a computação os estudantes, do Ensino Fundamental e Médio, devem ser apresentados. Apesar disso, nesse *framework*, o estudo sobre Inteligência Artificial é citado em apenas dois objetivos de aprendizagem:

- Descreva como a inteligência artificial move muitos softwares e sistemas físicos.
- Implementar um algoritmo de Inteligência Artificial que jogue um jogo contra um oponente humano ou resolva um problema.

Visando auxiliar e complementar essas propostas de diretrizes curriculares, a AI4K-12 uma iniciativa da Associação para o Avanço da Inteligência Artificial (AAAI) em conjunto com a Associação de Professores de Ciência da Computação (CSTA), anunciou uma ação conjunta para desenvolver diretrizes para ensinar Inteligência Artificial para

estudantes do ensino fundamental e médio (AAAI, 2018). Com o intuito de estruturar essas diretrizes, são definidas 5 grandes ideias que todo aluno deve saber (TOURETZKY et al., 2019):

- #1. Percepção: os computadores percebem o mundo usando sensores;
- #2. Representação e Raciocínio: Os agentes mantêm modelos/representações do mundo e usa os mesmos para raciocinar;
- #3. Aprendizagem: Os computadores podem aprender com dados;
- #4. Interação natural: Fazer os agentes interagir naturalmente com os humanos é um grande desafio para os desenvolvedores de IA.
- #5. Impacto Social: As aplicações de Inteligência Artificial podem impactar tanto positivamente quanto negativamente a sociedade.

A terceira grande ideia, referente a Aprendizagem, está diretamente ligada ao ensino de *Machine Learning*, pois tem como objetivo principal fazer com que computadores possam aprender com dados. De acordo com a diretriz, os conceitos de ML devem ser abordados na educação K-12 devem incluir (TOURETZKY et al., 2019a):

- O que é aprender?
- Abordagens para Machine Learning (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 os dados de treinamento influenciam a aprendizagem;
- Limitações do *Machine Learning*.

Os objetivos relacionados à aprendizagem de ML são detalhados (AI4K12, 2020) na Tabela 2.

Tabela 2 - Resumo dos objetivos de aprendizagem da terceira grande ideia (SALVADOR et al., 2021a) a partir de (TOURETZKY et al., 2019)

Conceito		3-5 ano escolar (EUA)	6-8 ano escolar (EUA)	9-12 ano escolar (EUA)
		Anos iniciais do Ensino Fundamental	Anos finais do Ensino Fundamental	Ensino Médio
Natureza de Aprendizagem	Humano vs. máquinas	Diferenciar entre como pessoas e computadores aprendem	Contrastar as características únicas do aprendizado humano e das maneiras que máquinas operam	Definir aprendizagem supervisionada, não supervisionada e por reforço e dar exemplos de aprendizagem humana similares a cada algoritmo

	Achando padrões em dados	Modelar como aprendizagem supervisionada identifica padrões em dados rotulados	Modelar como aprendizagem supervisionada identifica padrões em dados rotulados	Modelar como ML constrói um raciocinador para detecção de objetos ou previsão ajustando os parâmetros do raciocinador (suas interpretações internas)
	Treinando um modelo	Treinar um modelo de detecção de objetos usando machine learning, e então examinar a precisão do mesmo com entradas novas	Treinar e avaliar um modelo de detecção de objetos ou previsão usando machine learning em um conjunto de dados tabular	Usar um algoritmo de aprendizagem supervisionada ou não supervisionada para treinar um modelo com dados reais e avaliar os resultados
	Construir vs. usar um raciocinador	Demonstrar como dados de treino são rotulados quando se usa uma ferramenta de ML	Explicar a diferença entre treinar e usar um modelo de raciocínio	Ilustrar o que acontece em cada etapa necessária ao usar ML para construir um detector de objetos
	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	Descrever como vários tipos de algoritmos de ML aprendem ao ajustar suas interpretações internas
	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	Selecionar o tipo de algoritmo de ML apropriado (aprendizado supervisionado, não supervisionado ou por reforço) para resolver um problema de raciocínio
Redes neurais	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	Descrever as arquiteturas e usos das redes neurais feedforward, redes convolucionais 2D, recorrentes e adversárias generativas
	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	Treinar uma rede neural multicamadas usando o algoritmo de aprendizagem de retropropagação e descrever como os pesos dos neurônios e as saídas das unidades ocultas mudam como resultado da aprendizagem
Conjuntos de dados	Conjuntos de feature	Criar um conjunto de dados rotulado com features explícitas de diferentes tipos e usa uma ferramenta de machine learning para treinar um detector de objetos nestes dados	Criar um conjunto de dados para treinar um detector de objetos domésticos de decisão e explorar o impacto que diferentes features têm na árvore de decisão	Comparar dois conjuntos de dados com dados reais em termos de features incluídas e como estas features estão codificadas



	Conjunto de dados 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	Avaliar um conjunto de dados usado para treinar um sistema de IA real considerando o tamanho do conjunto de dados, a forma com que os dados foram adquiridos e rotulados, o armazenamento necessário, e o tempo estimado para produzi-lo
	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 detector de objetos, e como esse viés pode ser introduzido se o conjunto de treino não for balanceado apropriadamente	Investigar desequilíbrios nos dados de treinamento em termos de gênero, idade, etnia e outras variáveis demográficas que podem resultar em um modelo enviesado, utilizando alguma ferramenta de visualização

Deste modo, é esperado que os estudantes do Ensino Médio estejam aptos a simular uma rede neural utilizando o *Tensor Flow Playground* (ao nível de compreender), uma ferramenta interativa que permite, em tempo real, uma simulação de pequenas redes neurais, possibilitando visualizar seus respectivos resultados (GOOGLE, 2021b). Já atingir o nível de aprendizagem de aplicação seguindo a taxonomia de Bloom, espera-se que, por meio de ferramentas como scikit-learn e Python, estejam aptos a desenvolver pequenas aplicações de *Machine Learning* (TOURETZKY, 2019b).

Buscando estruturar ainda mais esse ensino de Inteligência Artificial, LONG e MAGERKO (2020) desenvolveram uma definição para *AI Literacy* (Alfabetização de IA), sugerindo algumas competências chaves, que os estudantes devem aprender. Essas competências são divididas em 5 temas diferentes (LONG; MAGERKO, 2020):

- O que é IA?
- O que IA pode fazer?
- Como a IA funciona?
- Como a IA deve ser usada?
- Como as pessoas enxergam a IA?

Tabela 3 - Competências necessárias para AI Literacy (SALVADOR et al., 2021a) a partir de (TOURETZKY et al., 2019).

#	Competência	Descrição
1	Reconhecendo IA	Distinguir entre artefatos tecnológicos que usam ou não IA
2	Entendendo Inteligência	Analisar e discutir criticamente características que fazem uma entidade "inteligente", incluindo discutir diferenças entre humanos, animais e inteligência de máquina
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
4	Generalista vs. Específico	Diferenciar entre IA generalista e IA específica
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
6	Imaginar o IA do Futuro	Imaginar possíveis aplicações futuras de IA e considerar os efeitos de tal aplicação no mundo
7	Representações	Entender o que representação de conhecimento é e descrever alguns exemplos de representações de conhecimento
8	Tomada de Decisão	Reconhecer e descrever exemplos de como computadores raciocinam e fazem decisões
9	Passos de ML	Entender os passos envolvidos em Machine Learning e as práticas e desafios de cada passo
10	Papel Humano na IA	Reconhecer que humanos fazem um papel importante na programação, escolha de modelos e no refinamento de sistemas de IA
11	Data Literacy	Reconhecer que computadores geralmente aprendem a partir dos dados
12	Aprender pelos Dados	Entender que dados não podem ser tomados por si só, e requerem interpretação. Descrever como os exemplos de treino dados em um conjunto de dados inicial podem afetar o resultado de um algoritmo
13	Interpretar Criticamente os Dados	Entender que dados não podem ser tomados por si só, e requerem interpretação. Descrever como os exemplos de treino dados em um conjunto de dados inicial podem afetar o resultado de um algoritmo
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)
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.
16	Ética	Identificar e descrever diferentes perspectivas nos problemas éticos chave acerca de IA (privacidade, singularidade, tomadas de decisões éticas, etc.)
17	Programabilidade	Compreender que agentes são programáveis.

Com base nessas diretrizes, são desenvolvidas unidades instrucionais (UIs) para ensinar *Machine Learning*. Essas unidades instrucionais, podem ser cursos, oficinas, tutoriais, etc. Nesse sentido, observa-se que nos últimos anos diversas unidades instrucionais foram desenvolvidas com foco no ensino de Inteligência Artificial e *Machine Learning* no Ensino Médio (MARQUES et al., 2020). Apesar da maioria dessas UIs terem o foco direcionado apenas na parte teórica, fomentando o conhecimento e compreensão,

existem também algumas que incentivam o estudante a desenvolver modelos de Machine Learning (MARQUES et al., 2020).

Essas unidades instrucionais muitas das vezes utiliza a progressão *Use-Modify-Create* (UMC), que é baseada em elementos da Teoria da Aprendizagem Experiencial (*Experiential Learning Theory*), que define que o conhecimento é criado por meio da transformação da experiência (MARTIN et al., 2020). Nesse sentido, a UMC tem sido usada para envolver os alunos, de forma progressiva em ambientes computacionais. Dentro da etapa de Uso (*Use*), os alunos consomem materiais de ML prontos, seja modelos, artefatos, dados processados, etc. Posteriormente, na etapa de Modificar (*Modify*) os alunos alteram o processo computacional dos artefatos, podendo inserir novos parâmetros, conjuntos dados, etc. Já na etapa de Criar (*Create*), os alunos já se encontram confiantes para desenvolver novos artefatos computacionais com seu próprio *design*.

Muitas dessas unidades instrucionais tem um tempo curto de duração, sendo oferecidas como aulas expositivas, oficinas, cursos, tanto no formato online, quanto no presencial. Nesses eventos, os conteúdos são tipicamente voltados a iniciantes, tendo como conteúdo a classificação de objetos, jogos, visão computacional, etc (MARQUES, 2020). Embora muitas dessas unidades instrucionais sejam em língua inglesa, ainda assim existem iniciativas em língua portuguesa, como por exemplo *Machine Learning Para Todos* (GRESSE VON WANGENHEIM et al., 2020).

## **2.3 AVALIAÇÃO DA APRENDIZAGEM**

A avaliação da aprendizagem é uma das etapas mais importantes no processo de aprendizagem dos alunos. A avaliação do processo de ensino e aprendizagem é realizada de forma contínua, cumulativa e sistemática, é um instrumento utilizado para avaliar a evolução dos alunos ao longo do processo de ensino-aprendizagem. Esse processo não deve somente priorizar o resultado ou o processo de ensino, mas deve buscar identificar os conhecimentos construídos e as dificuldades de uma forma dialógica (NÚÑEZ, 2019).

Nesse contexto, a avaliação de ensino e aprendizagem traz diversos benefícios, não só para os alunos, mas também para os educadores. Pois, além de permitir que o aluno saiba como está o andamento do seu aprendizado, permite que os professores possam buscar novos métodos para impulsionar o ensino e a aprendizagem (ANDRADE, 2020).

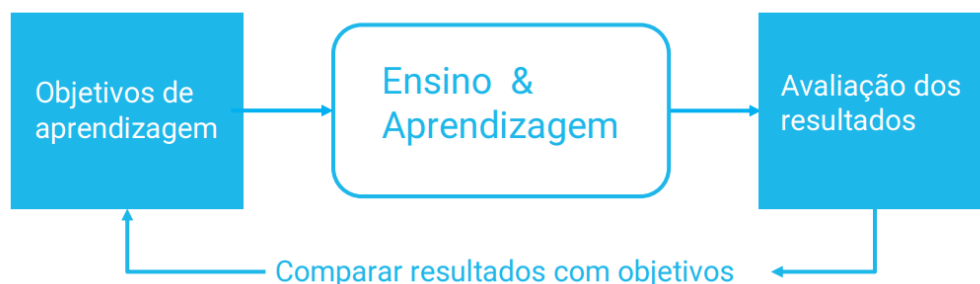


Figura 11: Comparação dos resultados com objetivos do processo de avaliação de aprendizagem (GRESSE VON WANGENHEIM, 2021).

Os objetivos de aprendizagem podem se direcionar a diferentes níveis de aprendizagem. A taxonomia de Bloom (BLOOM et al., 1956) é um sistema de classificação que estrutura os objetivos educacionais de forma hierárquica, classificando os objetivos para o aprendizado, separando-os em níveis de especificidade e complexidade, além de analisar os níveis de aquisição e conhecimento (DUTRA, 2020). Segundo essa classificação, o aprendizado pode ser classificado em três domínios: (1) **domínio cognitivo**, que está ligado a aprendizagem intelectual, (2) domínio afetivo, que lida com nossos sentimentos e emoções, e (3) domínio psicomotor, que está ligado a execução de tarefas que dependem do nosso sistema motor. O domínio cognitivo tem sido o foco principal da educação tradicional, sendo usado para estruturar objetivos, avaliações e atividades do currículo (DUTRA, 2020). A Tabela 4 categoriza os objetivos educacionais que são levantados no modelo de domínio cognitivo.

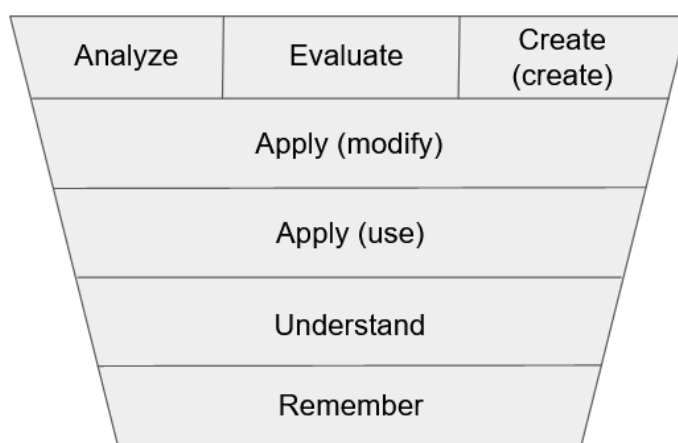


Figura 12: Níveis de aprendizagem da taxonomia de Bloom. Adaptado de (JOHNSON, 2006).

De acordo com o nível de aprendizagem são definidos os métodos de avaliação. Existem diversos tipos de avaliações, utilizando a Taxonomia de Bloom, a Tabela 4 apresenta os três primeiros níveis de aprendizagem e seus respectivos tipos de avaliação:

Tabela 4: Níveis de aprendizagem e seus respectivos tipos de avaliação. Adaptado de (RAGUPATHI, 2020).

Nível	Descrição	Tipos de avaliação
Conhecimento	Reconhecer e lembrar fatos, termos e conceitos básicos	Múltipla escolha, verdadeiro/falso, ligar colunas
Compreensão	Entender fatos e ideias, organizar, interpretar, traduzir e escrever ideias principais.	Múltipla escolha, verdadeiro/falso, redação
Aplicação	Aplicar conhecimentos adquiridos para resoluções de problemas	Resposta curtas discursivas, redação

Para auxiliar nesse processo de avaliação e aprendizagem, existem diferentes tipos de avaliação de aprendizagem que podem ser aplicadas, tendo a possibilidade de ser utilizadas de forma alternada. De acordo com o design instrucional, uma avaliação pode ser classificada em: **diagnóstica**, **formativa** e **somativa**. A avaliação diagnóstica é realizada antes de qualquer interferência instrucional, pois tem como objetivo identificar os conteúdos e conhecimentos prévios do aluno sobre o tema em questão. Nesse cenário, a avaliação serve como uma ferramenta para planejar melhor o ensino e a aprendizagem (ANDRADE, 2020). Ela pode ser operacionalizada por meio de entrevistas, questionários, produções textuais, simulados e etc. O objetivo da avaliação formativa é identificar se as propostas aplicadas estão sendo alcançadas durante o processo de ensino e aprendizagem. Nesse processo são coletados dados sobre a aprendizagem do aluno durante o ensino, para que seja feita a adequação do ensino e o repasse do feedback para o aluno (NÚÑEZ, 2019). Esse tipo de avaliação pode ser feita diariamente, ocasionalmente ou de forma periódica, por meio de observação de desempenho, aplicação de provas, desenvolvimento de projetos, etc. Já a avaliação somativa, a mais conhecida no dia a dia, é o tipo de avaliação que quantifica, atribuindo uma nota aos estudantes, geralmente sendo aplicada bimestralmente ou semestralmente. Esse tipo de avaliação abrange todos os conteúdos que os estudantes aprenderam durante aquele período, com isso, não consegue avaliar nenhum conceito com muita profundidade, pois abrange uma diversidade muito grande de conteúdos (ANDRADE, 2020).

Em muitos casos, os alunos não sabem como estão sendo avaliados, quais são as habilidades e conhecimentos que se espera que eles tenham desenvolvido durante o processo de aprendizagem. Deste modo, uma alternativa para lidar com esses tipos de

avaliações de habilidades mais complexas é associar o uso de rubricas de avaliação. A rúbrica é um instrumento de avaliação que indica, em forma de escala, quais critérios específicos de uma determinada tarefa se deseja avaliar (LENHARDT, 2020).

Principalmente voltado à avaliação de aprendizagem nos níveis mais altos (aplicação etc.) se usa avaliações baseadas no desempenho. Além de ficar explícito para o estudante quais aspectos estão sendo avaliados, a rubrica facilita muito no fornecimento de *feedback* por parte dos avaliadores. As rubricas de avaliações são compostas basicamente por quatro componentes (MATHEUS, 2018):

- Descrição detalhada da tarefa.
- As dimensões da tarefa, que se referem aos aspectos que serão avaliados.
- Uma escala que descreve diferentes níveis de desempenho.
- Descrição dos diferentes níveis de desempenho em cada uma das dimensões da tarefa.

É necessário que as rubricas sejam confiáveis e validadas. A confiabilidade refere-se a quão consistente ou preciso é um instrumento de avaliação, tendo a capacidade de reproduzir um resultado consistente no tempo e no espaço, ou a partir de avaliadores diferentes (SOUZA et al., 2017). E a validade tem como objetivo medir o quão preciso é aquele instrumento de avaliação e se o mesmo mede o que se propõe a medir (GRESSE VON WANGENHEIM, 2021).

### 3. ESTADO DA ARTE

Neste capítulo é apresentado o estado da arte visando a abordagens de avaliação de aprendizagem de *Machine Learning* no Ensino Médio. Para realizar o mapeamento sistemático da literatura é utilizado o processo proposto por Petersen et al (2008). Considerando a revisão da literatura realizada no início de 2021 com a mesma pergunta de pesquisa (SALVADOR et al., 2021), visa-se neste trabalho levantar avaliações em cursos publicados mais recentemente, complementando a revisão publicada.

#### 3.1 DEFINIÇÃO DO PROTOCOLO DE BUSCA

Com base no tema deste trabalho, visa-se realizar um mapeamento sistemático, que tem como objetivo identificar classificar e interpretar pesquisas disponíveis, utilizando critérios de qualificação claros e reproduzíveis (PETERSEN et al., 2008) atualizando a revisão literária feita por Salvador et al. (2021b). A pergunta de pesquisa que este mapeamento procura responder é: **quais modelos existem para avaliação de aprendizagem de Machine Learning no Ensino Médio?** Esta questão é refinada nas seguintes perguntas de análise:

- **PA1.** Quais unidades instrucionais voltadas ao ensino de ML no Ensino Médio apresentam avaliações de aprendizagem?
- **PA2.** Quais são as características destas avaliações em termos de nível de aprendizagem, conteúdo e tipo?
- **PA3.** Que feedback instrucional é apresentado?

**Bases de dados:** Para a realizar a busca de artigos, são utilizado como fonte de pesquisa os artigos indexados pelas ferramentas digitais da área: Scopus, IEEE Xplore, ACM, SpringerLink, ScienceDirect, arXiv, SocArXiv e Google Scholar.

**Crítérios de inclusão e exclusão:** São considerados artigos na língua inglesa cujo foco seja apresentar um modelo de avaliação de aprendizagem de Machine Learning no contexto do Ensino Médio. Complementando a revisão realizada por Salvador et al. (2021b) no início de 2021, são considerados nesta pesquisa artigos publicados em 2021, dado a tendência recente de ensino de conceitos de ML. Foram excluídos artigos com foco no ensino de ML no Ensino Fundamental e/ou Ensino Superior ou que abordam

Inteligência Artificial sem abordar conceitos de ML. Publicações como blogs, vídeos ou outras ferramentas que não compõem uma unidade instrucional para estudantes de Ensino Médio e artigos que não apresentam informações substanciais foram descartados. Também foram excluídos artigos que somente apresentam uma avaliação do curso em si, mas não apresentam propostas para a avaliação da aprendizagem do aluno.

**Critérios de qualidade:** São considerados apenas artigos que apresentem informações substanciais para se extrair referentes às perguntas de análise. São excluídos artigos que apresentam, por exemplo, somente um resumo de uma proposta e para os quais não são encontradas mais informações detalhadas. Também são excluídas unidades instrucionais que possuem custo para acesso do material completo.

**Termos de busca:** Com base na pergunta de pesquisa foram definidos inicialmente os termos de busca e seus sinônimos indicados na Tabela 5.

Tabela 5- Termos de busca.

Termo	Sinônimos
<i>machine learning</i>	<i>data science, artificial intelligence, deep learning</i>
<i>object detection</i>	-
<i>education</i>	<i>teach*, course, mooc, learn*</i>
<i>school*</i>	<i>k-12, kids, children, teen*</i>

Uma string de busca genérica foi definida a partir dos termos de busca para aplicar nas bases de dados.

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

A string de busca genérica foi adaptada para realizar a busca em cada uma das bases de dados consideradas (Tabela 6).

Tabela 6 - Strings de buscas utilizadas nas diferentes bases de dados (SALVADOR et al., 2021b).

Base de dados	String de busca
Scopus	(TITLE-ABS-KEY(teach* OR education OR course OR mooc OR learn*) AND TITLE-ABS-KEY("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning" OR "object detection") AND TITLE-ABS-KEY("k-12" OR school* OR kids OR children OR teen*)) AND PUBYEAR > 2020 (LIMIT-TO(SUBJAREA, "COMP"))



IEEE Xplore	("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" OR "object detection") AND ("Abstract":"k-12" OR "Abstract":school* OR "Abstract":kids OR "Abstract":children OR "Abstract":teen*)
ACM	[[Abstract: teach*] OR [Abstract: education] OR [Abstract: course] OR [Abstract: mool] OR [Abstract: learn*]] AND [[Abstract: "machine learning"] OR [Abstract: "data science"] OR [Abstract: "artificial intelligence"] OR [Abstract: "deep learning"] OR [Abstract: "object detection"]] AND [[Abstract: "k-12"] OR [Abstract: school*] OR [Abstract: kids] OR [Abstract: children] OR [Abstract: teen*]] AND [Publication Date: (01/01/2021 TO *)]
SpringerLink	(teach* OR education OR course OR MOOC OR learn*) ("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning" OR "object detection") ("k-12" OR school* OR kids OR children OR teen*) Discipline: Computer Science 2021-2021
ScienceDirect	Year: 2020-2021 Title, abstract, keywords: (teach OR education OR course OR learn) ("machine learning" OR "deep learning" OR "object detection") ("k-12" OR school OR teen)
arXiv	order: -announced_date_first; size: 200; date_range: from 2021-01-01 to 2021-12-31; classification: Computer Science (cs); include_cross_list: True; terms: AND abstract=teach* OR education OR course OR MOOC OR learn*; AND abstract="machine learning" OR "data science" OR "artificial intelligence" OR "deep learning" OR "object detection"; AND abstract="k-12" OR school* OR kids OR children OR teen*
SocArXiv	(teach* OR education OR course OR MOOC OR learn*) ("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning" OR "object detection") ("k-12" OR school* OR kids OR children OR teen*) AND date:[2021-01-01 TO 2021-12-31]
Google Scholar	(teach* OR education OR course OR MOOC OR learn*) ("machine learning" OR "data science" OR "artificial intelligence" OR "deep learning" OR "object detection") ("k-12" OR school* OR kids OR children OR teen*)

### 3.2 EXECUÇÃO DA BUSCA

A busca dos artigos foi realizada em julho de 2021 pelo autor do presente trabalho e revisada pela orientadora. A busca inicial resultou em 24.037 artigos. A quantidade de artigos em cada etapa do processo de seleção é apresentada na Tabela 7.

Tabela 7 - Número de artigos identificados por base de dados

Base de dados	Número de artigos identificados	Número de artigos analisados	Primeira etapa analisando título e resumo de curso de ML no Ensino Médio	Segunda etapa analisando o texto na íntegra de curso de ML no Ensino Médio	Terceira etapa analisando o texto na íntegra de avaliação como parte do curso de ML no Ensino Médio
Scopus	537	300	2	0	-
IEEE Xplore	128	128	0	-	-
SpringerLink	6.412	300	1	1	1

ACM	57	57	2	1	1
ScienceDirect	17	17	0	-	-
arXiv	37	37	0	-	-
SocArXiv	249	249	0	-	-
Google Scholar	16.600	300	1	1	1
<b>Total</b>					<b>3 (sem duplicatas)</b>

Complementando o mapeamento realizado por Salvador et al. (2021b), consideramos apenas os resultados que continham atividades relacionadas a algum tipo de avaliação no Ensino Médio. Na primeira etapa, títulos e resumos dos artigos foram analisados, selecionado os artigos que possuíam alguma referência a curso de ML no Ensino Médio. Na segunda etapa, os artigos selecionados foram analisados por completo, considerando apenas aqueles respeitavam os critérios levantados anteriormente. E na última etapa, foi analisado se esses artigos selecionados possuíam a avaliação como parte do ensino de ML no Ensino Médio.

### 3.3 ANÁLISE DOS RESULTADOS

Foi realizada a extração dos dados dos artigos selecionados, tendo como objetivo responder às questões definidas no protocolo de busca. As informações relevantes às perguntas de análise foram extraídas conforme especificado na Tabela 8.

Tabela 8 - Informações extraídas. Adaptado de (SALVADOR et al., 2021b).

<b>Pergunta de análise</b>	<b>Dados a extrair</b>
PA1. Quais unidades instrucionais voltadas ao ensino de ML no Ensino Médio apresentam avaliações de aprendizagem?	Nome
	Descrição
	Fonte
PA2. Quais são as características destas avaliações em termos de nível de aprendizagem, conteúdo e tipo?	Método de avaliação
	Níveis de aprendizado do Bloom
	Fase do "Use-Modify-Create"

	Conceitos de ML
PA3. Que feedback instrucional é apresentado?	Tipo de feedback instrucional
	Tipo de automação

## PA1. QUAIS UNIDADES INSTRUCIONAIS VOLTADAS AO ENSINO DE ML NO ENSINO MÉDIO APRESENTAM AVALIAÇÕES DE APRENDIZAGEM?

Na primeira etapa foi extraído das unidades instrucionais dados como referência, nome, descrição e fonte, para que futuramente possamos utilizar como fonte de consulta. Esses dados estão contidos na tabela 9.

Tabela 9 - Avaliações de aprendizagem de Machine Learning no Ensino Médio

Referência	Nome	Descrição
(GARCÍA et al., 2021)	LearningML	Oficina e Infraestrutura técnica voltadas a ensinar conceitos básicos de ML por meio de um sistema web e uma extensão para Scratch permitindo a implantação do modelo desenvolvido.
(VOULGARI et al., 2021)	ArtBot	Oficina aplicando um Jogo gratuito que permite que os jogadores interajam com o personagem principal por meio de práticas de IA/ML.
(BENDECHACHE et al., 2021)	AI in My Life	Oficina voltada a ensinar conceitos básicos de IA/ML, tendo atividades práticas de ML.

Foram encontrados somente 3 artigos que possuem algum tipo de avaliação de aprendizagem de ML voltado ao Ensino Médio publicado no início do ano de 2021. Mesmo observando a existência de mais alguns nos últimos 10 anos (Salvador et al., 2021), é nítido a lacuna de modelos de avaliação no contexto de cursos *Machine Learning* para o Ensino Médio.

## PA2. QUAIS SÃO AS CARACTERÍSTICAS DESTAS AVALIAÇÕES EM TERMOS DE NÍVEL DE APRENDIZAGEM, CONTEÚDO E TIPO?

Em uma segunda etapa foram extraídos os métodos de avaliação utilizados, bem como os quais os níveis de aprendizagem foram atingidos utilizando a taxonomia de Bloom e quais as fases do "Use-Modify-Create" foram aplicadas (Tabela 10).

Tabela 10 - Características das avaliações de aprendizagem

Nome	Conceitos de ML avaliados	Níveis de aprendizagem do Bloom	Fases do "Use-Modify-Create"	Método de avaliação
(GARCÍA et al., 2021)	Conceitos básicos de ML	Conhecimento, Compreensão, Aplicação	<i>Use, Modify, Create</i>	Questionário com 14 questões, questões abertas.
(VOULGARI et al., 2021)	Conceitos básicos de ML, impacto de ML	Conhecimento, Compreensão, Aplicação	<i>Use, Modify, Create</i>	Questionário com 19 questões abertas
(BENDECHACHE et al., 2021)	Conceitos básicos de ML, impacto de ML	Conhecimento	<i>Use</i>	Questionário, questões abertas.

De todas as avaliações selecionadas, duas referem-se a todos os níveis de aprendizagem da taxonomia de Bloom, possibilitando o estudante a chegar ao nível de aplicação. Como todas as avaliações têm como objetivo introduzir o tema para os estudantes, os conceitos básicos de ML são o foco da avaliação de aprendizagem. Além disso, uma dessas avaliações aborda o tema de segurança, ética e privacidade.

Todas as avaliações utilizam de questionários para avaliar as respostas dos estudantes. Apesar de uma dessas avaliações ficar apenas na fase de *Use* do ciclo "Use-Modify-Create" (BENDECHACHE et al., 2021), as outras duas (GARCÍA et al., 2021) e (VOULGARI et al., 2021), permitem que os estudantes criem e modifiquem modelos personalizados de ML.

### PA3. QUE FEEDBACK INSTRUCIONAL É APRESENTADO?

O *feedback* é uma das etapas mais importantes da avaliação, e pode ser apresentado de diversas maneiras em uma avaliação. Para que haja uma avaliação em grande escala, consistente e confiável, é interessante que a mesma seja automatizada. Nesse sentido, foi analisado quais *feedbacks* instrucionais eram oferecidos aos estudantes, bem como, quais os tipos de automação da avaliação, caso existisse (Tabela 11).

Tabela 11 - Conceitos de Machine Learning avaliados

Nome	Tipo de <i>feedback</i>	Tipo de automação
(GARCÍA et al., 2021)	Autocorreção durante o teste do modelo, permitindo a alteração dos parâmetros utilizados.	Correção automatizada, tendo a precisão do modelo criado.
(VOULGARI et al., 2021)	Autocorreção durante o jogo, permitindo a alteração dos parâmetros utilizados.	Correção automatizada, tendo <i>feedbacks</i> durante as etapas do jogo.
(BENDECHACHE et al., 2021)	Correção manual pelos instrutores do workshop.	Não possui

Analisando os dados coletados, é possível observar que Bendeche et al., (2021), não possui uma forma de automação para apresentar o *feedback*, limitando-se a apenas realizar a correção manual dos questionários respondidos pelos estudantes, informando apenas se as questões foram respondidas corretamente. García et al. (2021), apresenta uma avaliação informando ao estudante qual o nível de precisão e desempenho do seu modelo criado. Dessa maneira, fornece ao estudante um *feedback* incremental que permite a ele modificar seu modelo para melhorar a precisão e desempenho. Já Voulgari et al. (2021), retorna *feedbacks* em fases do jogo, mostrando penalidades (por meio de ícones vermelhos) e recompensas (por meio de ícones verdes), dessa maneira, guiando o estudante para que vá melhorando o aprendizado do seu personagem. Com esse *feedback*, o estudante consegue melhorar o desempenho do seu personagem e concluir as fases do jogo.

### 3.4 DISCUSSÃO

É possível observar a partir dessa revisão que continuam existindo poucas alternativas de avaliação no contexto de ensino de Machine Learning no Ensino Médio, complementando também a pesquisa realizada por Salvador et al. (2021b).

A maioria dessas unidades instrucionais tem como foco cursos introdutórios (Marques et al. 2020), avaliando conceitos básicos de ML, atingindo níveis de aprendizagem mais baixos segundo a taxonomia de Bloom, tendo assim poucas unidades instrucionais que chegam nos níveis mais altos de aplicação de ML.

Observa-se também que a maioria destas unidades instrucionais focam em ensinar a tarefa de classificação de imagens, sons etc. sendo esta uma das tarefas consideradas mais fáceis. Desta forma não encontrando também ainda modelos de avaliação no contexto de ensino de outras tarefas, como por exemplo a detecção de objetos.

O *feedback* apresentado aos estudantes é simples, indicando a corretude da tarefa, não focando tanto na avaliação de desempenho com base nos artefatos criados pelos estudantes como resultado da aprendizagem.

Nesse sentido, levando em consideração os dados obtidos, é possível observar a necessidade de melhoria nos modelos de avaliações referente ao ensino de Machine Learning no Ensino Médio, aprofundando os conceitos de Machine Learning e aprimorando as formas de avaliação de aprendizagem, tornando-as cada vez mais automatizadas para que, futuramente, possam ser utilizadas em larga escala nas escolas brasileiras.

## AMEAÇAS À VALIDADE DA REVISÃO DA LITERATURA

Como em qualquer mapeamento sistemático, existem algumas ameaças à validade dos resultados. As ameaças potenciais foram identificadas e estratégias de mitigação para minimizar os impactos foram aplicadas.

- **Viés de publicação:** Mapeamentos sistemáticos podem sofrer do viés comum de que os resultados positivos têm maior probabilidade de serem publicados do que os negativos. No entanto, foi considerado que os resultados dos artigos, sejam positivos ou negativos, têm apenas uma pequena influência sobre esse mapeamento sistemático, uma vez que foi buscado caracterizar os modelos de avaliação de aprendizagem de *Machine Learning* no Ensino Médio.
- **Identificação de estudos:** Outro risco é a omissão de estudos relevantes. A fim de mitigar esse risco, a *string* de busca foi construída para ser o mais abrangente possível, considerando não apenas os principais conceitos, mas também sinônimos. Isso gera um possível problema em ocultar artigos em bases de dados que foram recortadas em 300 resultados pelo grande volume de artigos encontrados, embora a possibilidade de ocorrência de tal fato seja menor do que a de omitir resultados sem usar sinônimos.
- **Seleção e extração de dados de estudos:** Ameaças para estudar seleção e extração de dados foram mitigadas por meio do fornecimento de uma definição detalhada dos critérios de inclusão, exclusão e de qualidade. Foi definido e documentado um protocolo para a seleção do estudo e a execução do mapeamento foi validada pela orientadora do presente trabalho.

## 4. MODELO DE AVALIAÇÃO

Neste capítulo é apresentado o modelo de avaliação de aprendizagem de *Machine Learning* voltado para o público do Ensino Médio, bem como a análise do público alvo e o contexto escolar. Foi desenvolvido também o plano de avaliação, a rubrica e *feedback* instrucional. Para construir a avaliação é utilizada a *Evidence-Centered Design* (ECD), uma metodologia sistemática para projetar avaliações visando assegurar que as evidências coletadas e interpretadas sejam consistentes com o conhecimento que a avaliação busca avaliar (MISLEVY et al., 2003).

### 4.1 ANÁLISE E MODELAGEM DO DOMÍNIO

Com o objetivo de desenvolver uma avaliação de aprendizagem coerente com a realidade, busca-se analisar o domínio do cenário em que é aplicada a avaliação, e, desta forma, adequar o modelo de avaliação ao contexto de aplicação.

**Público-alvo.** Este projeto visa desenvolver uma avaliação de aprendizagem voltada para os estudantes do Ensino Médio das escolas brasileiras. O Ensino Médio é a etapa final da Educação Básica, tendo como duração 3 anos, abrangendo estudantes entre 15 a 18 anos de idade. Espera-se nessa etapa final da educação básica que os alunos consolidem seus conhecimentos adquiridos no Ensino Fundamental, tenham um preparo para o mercado de trabalho e cidadania, aprimorem sua formação ética, adquirindo sua autonomia intelectual e compreendam os fundamentos científico-tecnológicos dos processos produtivos, relacionando a teoria com a prática, no ensino de cada disciplina (MINISTÉRIO DA EDUCAÇÃO, 2017).

Dado o grande avanço nas tecnologias de informação e comunicação, e o crescimento do uso dessas tecnologias pelos jovens, espera-se que nessa etapa da educação, os estudantes utilizem cada vez mais tecnologias digitais para a realização de uma série de atividades relacionadas a todas as áreas do conhecimento (MINISTÉRIO DA EDUCAÇÃO, 2017). O uso da internet e suas tecnologias estão sendo cada vez mais utilizadas para realizar tarefas escolares, conforme a Tabela 12. Sendo que 66% desses

alunos possuem ao menos um dispositivo com acesso a internet em suas casas, de acordo com a Tabela 13.

Tabela 12 - Uso da Internet em atividades escolares de alunos de escolas urbanas (CETIC, 2019a).

Atividades	5º ano do Ensino Fundamental	9º ano do Ensino Fundamental	Ensino Médio	Total
Participa de cursos	9%	17%	23%	<b>16%</b>
Realiza provas online	13%	28%	32%	<b>24%</b>
Divulga o trabalho realizado em grupo	23%	22%	30%	<b>25%</b>
Conversa com o professor	28%	23%	34%	<b>28%</b>
Utiliza a internet para realizar apresentações	31%	48%	60%	<b>46%</b>
Usa a Internet para aprender um idioma	50%	60%	64%	<b>58%</b>
Joga jogos educacionais	78%	55%	46%	<b>60%</b>
Faz atividade escolares a distância	42%	73%	80%	<b>65%</b>
Faz atividades e exercícios	56%	71%	79%	<b>69%</b>
Usa a internet para estudar para provas	55%	78%	88%	<b>74%</b>
Faz pesquisa do que é falado pelo professor na sala de aula	68%	77%	82%	<b>76%</b>
Faz pesquisa para trabalhos da escola	76%	88%	90%	<b>85%</b>

Tabela 13 - Computadores existentes na cada dos estudantes. (CETIC, 2019b)

Dispositivo	Escolas Públicas		Escolas Privadas		Total	
Desktop	61%	31%	91%	54%	66%	29%
Tablet		26%		44%		35%
Notebook		35%		71%		41%
Nenhum	39%		9%		34%	

Assim, atualmente observa-se que os alunos tipicamente aprendem o uso básico da tecnologia nas escolas. Porém ainda poucos aprendem a desenvolver soluções mais avançadas, apresentando um número baixo de alunos que no nível de Ensino Médio já sabem programar, menos ainda sabem atualmente aplicar técnicas de ML criando soluções inteligentes de tecnologia.

**Contexto escolar.** De acordo com Inep (2018), 60% das escolas brasileiras possuem recursos tecnológicos, como laboratório de informática, internet e banda larga. Aponta



também que essa disponibilidade de recursos se encontram mais presentes nas escolas de Ensino Médio quando se comparada a escolas de Ensino Fundamental.

Esses laboratórios geralmente são administrados por um profissional de tecnologia. Esses profissionais podem ser graduados ou estar graduando algum curso de licenciatura voltado à área de tecnologia. Apesar disso, poucos professores possuem qualificação para atuar nas aulas de computação. A Tabela 14 mostra que a maioria dos professores de escolas urbanas nunca receberam capacitação para uso de computador e internet em atividades de ensino-aprendizagem.

Tabela 14 - Professores de escolas urbanas que participaram de curso de formação continuada sobre o uso de computador e internet em atividades de ensino (CETIC, 2019c).

Capacitação	5º ano do Ensino Fundamental	9º ano do Ensino Fundamental	Ensino Médio	Total
Sim	23%	38%	40%	<b>33%</b>
Não	77%	62%	60%	<b>67%</b>

**Ensino de *Machine Learning* com Jupyter.** O modelo de avaliação de aprendizagem desenvolvido no presente trabalho é aplicado em um curso de *Machine Learning* voltado a detecção de objetos, no cenário doméstico, com o objetivo de ajudar pessoas cegas e/ou com deficiência visual (GONÇALVES, 2022). O curso é online e interativo, podendo ser realizado individualmente ou dentro de um contexto escolar. Em conjunto com Gonçalves (2022), são definidos dez objetos que serão detectados em um ambiente doméstico (Tabela 15).

Tabela 15 - Objetos a ser detectados

<b>Objetos</b>
Armário
Cachorro
Cadeira
Cama
Criado Mudo
Mesa

Mochila
Porta
Sapato
Sofá

Inicialmente o curso irá apresentar conceitos básicos e exemplos de aplicação de *Machine Learning*. Posteriormente os estudantes poderão desenvolver seus próprios modelos de ML, tendo que realizar toda a preparação do conjunto dos dados, incluindo etapas de *upload* dos dados, e separação do conjunto para teste, treinamento e validação do modelo. Em uma etapa final, também são apresentadas algumas questões éticas e de impactos de ML na sociedade.

O curso referente ao treinamento do modelo ainda está em fase de desenvolvimento (GONÇALVES, 2022), deste modo, como a parte de implantação ainda não está desenvolvida, os critérios de avaliação referentes à implantação serão definidos assim que seu desenvolvimento for finalizado.

## **4.2 DESENVOLVIMENTO DO *FRAMEWORK* CONCEITUAL**

### **4.2.1 Modelo de estudante**

De acordo com Mislevy et al. (2003), o modelo de estudante define uma ou mais variáveis relacionadas aos conhecimentos e habilidades que estão sendo avaliadas, sendo utilizadas posteriormente como definição para o que se deseja avaliar. No presente trabalho, essas variáveis são os objetivos de aprendizagem esperados que os alunos atinjam durante o curso (Tabela 16). Os objetivos foram desenvolvidos seguindo as diretrizes de AI4K12 (2020), especificamente à grande ideia 3, seguindo também a alfabetização em Inteligência Artificial de Long e Magerko (2020), tópicos gerais de computação propostos (CSTA, 2017), e o processo de ML (GRESSE VON WANGENHEIM e VON WANGENHEIM, 2021).

Tabela 16 - Objetivos de aprendizagem de Machine Learning para detecção de objetos no Ensino Médio adaptado a partir de (GONÇALVES, 2022).

Identificador	Objetivo de aprendizagem	Taxonomia de Bloom	Fonte
OA1 - Conceitos básicos de ML	Conceitos básicos de ML, saber identificar exemplo de ML	Lembrar	3-A-i (AI4K12, 2020); 3B-AP-08 (CSTA, 2017); 1, 2, 5 (LONG e MAGERKO, 2020)
OA2 - Redes Neurais	Conceitos básicos sobre redes neurais, tipos de arquitetura e seu comportamento	Compreender	3-B-i 6-8, 3-B-i 9-12, 3-B-ii 3-5 (AI4K12, 2020)
OA3 - Conceitos básicos de detecção de objetos	Compreender os conceitos de <i>bounding-boxes</i> , redes neurais convolucionais e algoritmos de detecção de objetos.	Compreender	3-C-ii 6-8, 3-B i 9-12, (AI4K12, 2020)
OA4 - Preparação de dados	Preparar um conjunto de dados, criando as <i>bounding boxes</i> e rotulando os dados	Aplicar	3-C-ii 9-12 (AI4K12, 2020); 3B-AP-09 (CSTA, 2017); 12 (LONG e MAGERKO, 2020)
OA5 - Treinamento de um modelo de detecção de objetos	Treinar um modelo de detecção de objetos a partir de um conjunto de dados	Aplicar	3B-AP-09 (CSTA, 2017); 3-A-iii 9-12 (AI4K12, 2020);
OA6 - Avaliar desempenho do modelo	Avaliar a performance do modelo de ML utilizando um conjunto de dados de teste.	Aplicar	3-A-iii 9-12 (AI4K12, 2020);
OA7 - Implantar o modelo em um aplicativo <i>mobile</i>	Exportar o modelo de detecção de objetos em um aplicativo móvel.	Aplicar	(AMERSHI et al., 2019)
OA8 - Discussão ética	Identificar e descrever problemas éticos chave acerca do ML (privacidade, tomada de decisões éticas, dados enviesados e correlações inesperadas, etc)	Compreender	3-C-iii 6-8 (AI4K12); 3A-AP-24 (CSTA, 2017); 13,16 (LONG e MAGERKO, 2020)

#### 4.2.2 Modelo de Evidência

O modelo de evidência fornece instruções de como os resultados das tarefas realizadas pelo aluno devem ser avaliadas, também como essas variáveis deverão ser utilizadas para *feedback* no nível de tarefa (MISLEVY et al., 2003). A rubrica está sendo adaptada a partir da rubrica definida por Salvador et al. (2021b) voltada à aprendizagem de desenvolvimento de classificação de imagens. Assim foram feitas adaptações para adaptar o modelo de detecção de objetos (marcado em negrito na Tabela 17).

Tabela 17 - Rubrica de avaliação e proposta de automação da avaliação. Adaptado de (SALVADOR et al., 2021b).

ID	Critério	Níveis de desempenho			Técnica de automação
		Baixo - 0 pt.	Aceitável - 1 pt.	Bom - 2 pt.	
<b>Preparação de dados (OA3)</b>					
C1	Quantidade de imagens rotuladas	Menos de 5 imagens por categoria	6 de 10 imagens por categoria	Mais de 10 imagens por categoria	Listar os diretórios e somar a quantidade de arquivos de imagem a partir dos dados enviados ao Jupyter Notebook
C2	Distribuição do conjunto de dados	Quantidade de imagens por categoria varia muito	Quantidade de imagens por categoria varia pouco	Todas as categorias possuem a mesma quantidade de imagens	Listar os diretórios, somar a quantidade de arquivos de imagem e comparar variação por categoria a partir dos dados enviados ao Jupyter Notebook
<b>Preparação de dados (OA4)</b>					
<b>Treinamento de modelo de ML/Transfer Learning (OA5)</b>					
C3	Treinamento	O modelo não foi treinado	O modelo foi treinado com os parâmetros padrão	O modelo foi treinado com parâmetros ajustados (arquitetura, épocas e tamanhos de lotes)	Verificar se o comando de treinamento foi executado, retirar qual foi a arquitetura do comando de definição do learner, retirar qual o número de épocas e tamanhos de lotes do comando de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML (Transfer Learning) (OA6)</b>					
C4	Análise do desempenho	Detecções incorretas não identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação correta em relação ao modelo	Verificar se o comando de display de categorias com baixa acurácia foi executado e se respostas no widget de interpretação usando ipywidgets estão corretas
C5	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	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 (OA6)</b>					
C6	Testes com novos objetos	Nenhum novo objeto testado	1-2 novos objetos testados	Mais de dois novos objetos testados	Verificar quantidade de vezes que o comando de treinamento do modelo foi executado com novos objetos
C7	Interpretação dos testes	Interpretação errada	---	Interpretação correta	Verificar se respostas no widget de interpretação usando ipywidgets estão corretas

Para verificar as respostas das questões de interpretação e coletar dados referentes ao treinamento dentro do Jupyter Notebook serão usados widgets desenvolvidos com a ferramenta ipywidgets (IPYWIDGETS, 2017). Critérios em cinza não foram desenvolvidos pois faltam recursos que atualmente não são disponibilizados pelo yolov5.

**Modelo de medição.** Para o modelo de medição, é proposto um cálculo para chegar a uma nota final para as atividades desenvolvidas pelo estudante. Esse cálculo é realizado pela soma das pontuações dos quizzes e pontos da rubrica de desempenho.

Nesse sentido, a nota final do desempenho é dada por uma nota de 0 a 10, onde 10 representa o desempenho máximo do estudante. O cálculo é feito da seguinte forma:

$$\text{nota final} = (\text{soma da pontuação dos critérios da rubrica}) / 2$$

### 4.2.3 Modelo de Tarefa

O modelo de tarefa descreve o material que será produzido pelo estudante, definindo onde o modelo de avaliação será aplicado. O modelo de tarefa de avaliação pelo desempenho refere-se ao desenvolvimento de um modelo de ML para a detecção de 10 objetos em ambientes domésticos. Nesta tarefa, o estudante deve desenvolver este modelo seguindo o tutorial apresentado no curso de ML (GONÇALVES, 2022) em Jupyter Notebook executado no Google Colab. É disponibilizado também para o estudante um conjunto de dados de 500 imagens de objetos domésticos já rotulados. A definição da tarefa é apresentada de forma detalhada na Tabela 28

Tabela 18 - Definição do modelo de tarefa.

Objetivo do modelo de DL	
<b>Tarefa</b>	Detectar objetos domésticos em uma imagem (tipicamente em ambiente doméstico como sala, quarto, escritório, etc.).
<b>Tipo da tarefa</b>	<i>Multi-label</i> detecção de objetos
<b>Categorias</b>	10 objetos domésticos Armário, Cachorro, Cadeira, Cama, Criado Mudo, Mesa, Mochila, Porta, Sapato e Sofá
<b>Experiência</b>	Conjunto de imagens de objetos domésticos
Conjunto de dados	
<b>Fonte de dados</b>	Conjunto de dados de objetos domésticos disponibilizado pela CnE

<b>Quantidades de dados</b>	No total de 500 imagens	
<b>Padronização das imagens</b>	Formato: .jpeg	Tamanho: 480x480 pixels
<b>Desempenho</b>		
<b>Medidas</b>	Precisão média, com o nível de confiança de 50%, de no mínimo 90% total e por categoria. No mínimo 90%	

#### 4.2.4 Modelo de documentação

O modelo de documentação tem como objetivo automatizar a documentação das principais características do modelo de ML criado pelo estudante, sendo gerado automaticamente a partir das informações do Jupyter Notebook. O Cartão de Modelo serve como um modelo padrão de documentação, ele centraliza informações sobre um modelo de ML, tendo características de como o modelo foi construído, quais algoritmos foram utilizados, características do conjunto de dados, quais interpretações foram levantadas durante o processo de desenvolvimento, etc (MITCHELL et al., 2019). O Cartão de Modelo a ser montado terá o template conforme tabela 19.

Tabela 19 - Formato do Cartão de Modelo.

<b>Modelo de ML</b>	
Nome do modelo	
Data	
Versão	
<b>Objetivo do modelo de ML</b>	
Tarefa	
Contexto de uso	
Público alvo	
Riscos	
Tipo da tarefa	
Categorias	
<b>Conjunto de dados</b>	
Descrição dos dados	
Origem dos dados	

Distribuição dos dados por categoria	
Tamanho de imagens	
<b>Treinamento</b>	
Tipo de modelo	
Tamanho de lotes	
Quantidade de épocas	
<b>Avaliação de desempenho</b>	
Precisão média total	
Precisão média por categoria	
Matriz de confusão	
<b>Predição</b>	
Quantidade de testes realizados	
<b>Referências</b>	
Autores e afiliação	

## 5 DESENVOLVIMENTO DA AUTOMAÇÃO

A partir da análise e modelagem do domínio e do desenvolvimento do framework conceitual, foram realizadas várias iterações para automatizar a rubrica de avaliação, utilizando um processo iterativo/incremental de desenvolvimento de software (LARMAN; BASILI, 2003).

### 5.1 ANÁLISE DE REQUISITOS

A Tabela 20 apresenta os requisitos funcionais da automação e a Tabela 21 apresenta os requisitos não funcionais, de acordo com as funcionalidades implementadas no Jupyter Notebook no ambiente do Google Colab.

Tabela 20 - Requisitos funcionais. Adaptado de (SALVADOR et al., 2021b).

Requisito	Descrição	Artefato Entrada	Artefato Saída
Automatizar a aplicação da rubrica de avaliação	A ferramenta deve automatizar a aplicação da rubrica de avaliação de aprendizagem de ML	Objeto com as informações do treinamento no Jupyter Notebook com YOLO5s	Nota de 0 a 10 representando a aprendizagem de ML e pontuação por critério avaliado
Gerar um JSON com informações sobre o treinamento no Notebook	A ferramenta deve gerar um arquivo no formato JSON contendo as informações do Jupyter Notebook	Objeto com as informações do treinamento no Jupyter Notebook	Arquivo JSON com as informações do Jupyter Notebook
Gerar um Modelo de Documentação	A ferramenta deve gerar um Modelo de Documentação no formato PDF contendo as informações do Jupyter Notebook	Objeto com as informações do treinamento no Jupyter Notebook	Arquivo PDF no formato Modelo de Documentação com as informações do Jupyter Notebook
Apresentar um retorno visual dos resultados do modelo	A ferramenta deve retornar uma imagem de um "ninja-robô" retornando visualmente os resultados da avaliação de forma lúdica	Nota da avaliação	Imagem do "ninja-robô" com uma faixa conforme a nota da avaliação

Tabela 21 - Requisitos não funcionais. Adaptado de (SALVADOR et al., 2021b).

Requisito	Descrição
Linguagem de programação Python	Como a aplicação será executada em um Jupyter Notebook no Google Colab utilizando Python (recomenda-se a versão 3), a automação da rubrica deverá ser criada utilizando a mesma linguagem para compatibilidade de bibliotecas.



Repositório PyPI	A aplicação deve ser disponibilizada no repositório oficial de software de terceiros para Python, o PyPI (PYPI, 2022). É fonte padrão para pacotes e dependências do pip, disponível no Google Colab
Biblioteca YOLOv5	A aplicação irá utilizar a biblioteca do YOLOv5 para realizar o treinamento do modelo, e para realizar as detecções dos objetos.
Servidor HTTP	A aplicação deve ser disponibilizada como um servidor HTTP, expondo uma rota para realizar a avaliação com base no JSON gerado com as informações do treinamento
Repositório Git	O código-fonte da aplicação deve estar disponibilizado em um repositório Git institucional

## 5.2 ARQUITETURA

O desenvolvimento da automação da avaliação teve como resultado um pacote Python, cujo nome é `jupidetect` (Jupyter Notebook Object Detection). Todas as informações do modelo, como conjunto de dados, modelo escolhido, treinamento do Transfer Learning e respostas das questões de interpretação devem ser fornecidas em um objeto Python da classe `ObjectDetection`. Esse objeto deve estar disponível e com seus atributos preenchidos no código Python do Jupyter Notebook em que será aplicada a automação da rubrica. O `jupidetect` possui quatro módulos: *grader*, *json*, *ninja\_robot* e *pdf* e duas interfaces principais de uso: via pacote disponibilizado no PyPI ou servidor HTTP em Flask (2022), disponível para execução como container Docker (2021). Os módulos do `jupidetect` são apresentados a seguir.

- *grader*: responsável por automatizar a avaliação das competências, retornando o resultado da avaliação em um dicionário para o Jupyter Notebook;
- *json*: retorna um JSON com as informações coletadas do objeto `ObjectDetection`. Esse JSON pode ser utilizado como corpo da requisição para a API;
- *ninja\_robot*: salva no diretório uma imagem com um “ninja-robô” com uma faixa representando a nota atingida na avaliação;
- *pdf*: monta um Modelo de Documentação (conforme formato da seção 4) com as informações do modelo em formato PDF.

A arquitetura completa da automação é apresentada na Figura 13.

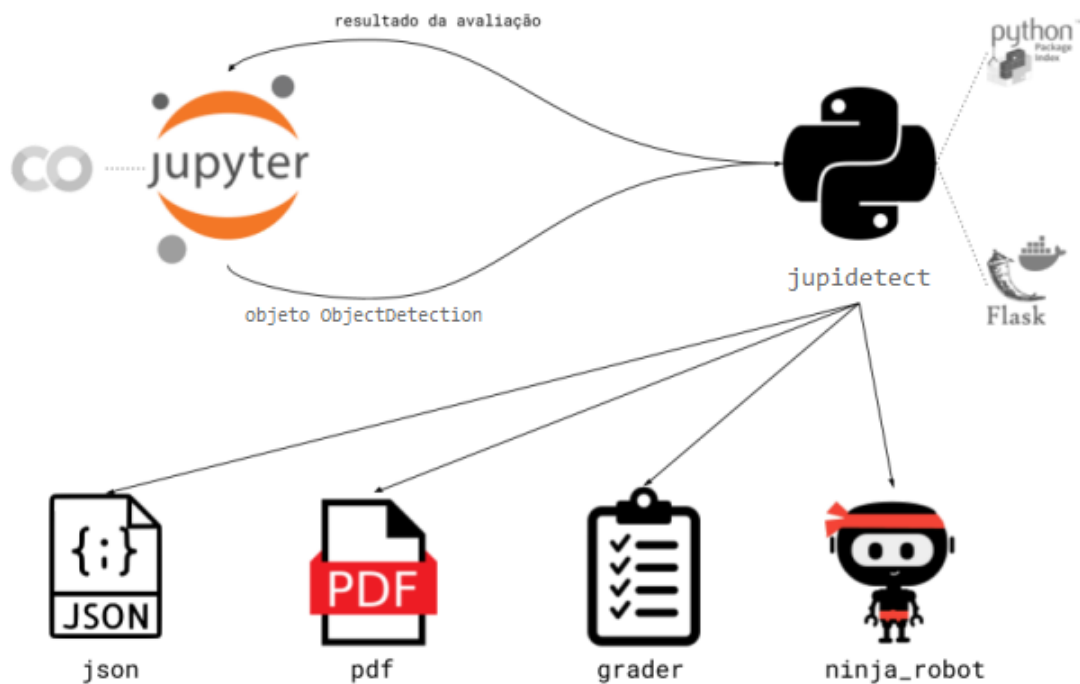


Figura 13: Arquitetura da automação

### 5.3 IMPLEMENTAÇÃO

A rubrica de avaliação foi aplicada e testada em um Jupyter Notebook do curso para a detecção de objetos utilizando Yolov5. Foram utilizados também alguns *widgets* interativos para coletar algumas informações necessárias do treinamento, como nas questões de interpretação de precisão média da Figura 13. O arquivo .ipynb, extensão do Jupyter Notebook, com exemplos de coleta das informações está disponibilizado com a biblioteca, podendo servir de base para adaptação e posterior utilização da ferramenta em outra tarefa de detecção de objetos em um Jupyter Notebook.

1. Qual a precisão média do modelo como todo com uma confiança de 50%?

0,742

2. Analisando esse valor, você pode concluir que o modelo detecta os objetos de forma geral com uma precisão média acima de 90%?

- Verdadeiro  
 Falso

3. Qual a precisão média (com 50% de confiança) por categoria?

Armario	0,17
Cachorro	0,77
Cadeira	0,877
Cama	0,995
Criado mudo	0,995
Mesa	0,855
Mochila	0,695
Porta	0,431
Sapato	0,642
Sofa	0,995

Figura 13 - Exemplo de *widget* interativo da questão de interpretação de precisão média

### 5.3.1 Critérios

Os critérios da rubrica de avaliação são apresentados a seguir, assim como um pseudocódigo representando o código Python que automatiza a sua correção correspondente.

#### Critério 1

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
Preparação de dados (OA3)				
C1	Quantidade de imagens rotuladas	Menos de 5 imagens por categoria	6 de 10 imagens por categoria	Mais de 10 imagens por categoria

```
avalia_c1():
```

para cada categoria do modelo  
 soma quantidade de imagens em uma pontuação  
 se imagens < 6 -> 0  
 se imagens < 10 e imagens > 5 -> 1  
 se imagens > 9 -> 2

retorna divisão da pontuação pela quantidade de categorias

**Critério 2**

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
<b>Preparação de dados (OA3)</b>				
C1	Distribuição do conjunto de dados	Quantidade de imagens por categoria varia muito	Quantidade de imagens por categoria varia pouco	Todas as categorias possuem a mesma quantidade de imagens

**avalia\_c2():**

para cada categoria do modelo  
 soma número positivo da diferença entre a quantidade de imagens da categoria atual e a próxima em uma pontuação

retorna  
 se pontuação > 20 -> 0  
 se pontuação > 0 e pontuação < 20 -> 1  
 se pontuação == 0 -> 2

**Critério 3**

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
<b>Treinamento de modelo de ML/Transfer Learning (OA5)</b>				
C3	Treinamento	O modelo não foi treinado	O modelo foi treinado com os parâmetros padrão	O modelo foi treinado com parâmetros ajustados (arquitetura, épocas e tamanhos de lotes)

**avalia\_c3():**

**retorna**

**se a variável que indica se modelo foi treinado for falsa -> 0**

**senão**

**se um dos arrays de modelos, épocas e tamanho de lotes  
tiver tamanho 1 -> 1**

**se os arrays de modelos, épocas e tamanhos de lotes  
tiverem tamanho maior que 1 -> 2**

#### **Critério 4**

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
<b>Avaliação e interpretação do desempenho de um modelo de ML (Transfer Learning) (OA6)</b>				
C4	Análise do desempenho	Detecções incorretas não identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação correta em relação ao modelo

**avalia\_c4():**

**pergunta qual a precisão média para todas as categorias**

**se estudante acerta  
soma 2 a uma pontuação**

**pergunta se o modelo possui uma precisão média acima de 90%**

**se estudante afirma e precisão média geral for acima de 90%  
soma 2 a uma pontuação**

**se estudante nega e precisão média geral por abaixo de 90%  
soma 1 a uma pontuação**

**pergunta a precisão média de todas as categorias**

**se estudante diz que todas as categorias possuem precisão média acima de 90%**

**se todas as categorias possuem precisão média acima de 90%  
soma 2 a uma pontuação**

**senão  
soma 0 a uma pontuação**

**retorna**

soma da pontuação / 3

### Critério 5

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
Avaliação e interpretação do desempenho de um modelo de ML/Testar e aperfeiçoar programas (OA6)				
C5	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

**avalia\_c5():**

**pede para estudante dizer se houve tentativas de melhorias**

**retorna**

**se houve tentativa de melhoria**

**há comentário sobre o que foi alterado -> 2**

**senão ->**

**senão -> 0**

### Critério 6

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
Avaliação e interpretação do desempenho de um modelo de ML/Testar e aperfeiçoar programas (OA6)				
C6	Testes com novos objetos	Nenhum novo objeto testado	1-2 novos objetos testados	Mais de dois novos objetos testados

**avalia\_c6():**

**pede novos objetos testados ao estudante**

**retorna**

**se objetos testados > 2 -> 2**

**se objetos testados < 3 e objetos testados > 0 -> 2**

se objetos testados == 0 -> 0

## Critério 7

ID	Critério	Níveis de desempenho		
		Baixo - 0	Aceitável - 2	Bom - 2
Avaliação e interpretação do desempenho de um modelo de ML/Testar e aperfeiçoar programas (OA6)				
C7	Interpretação dos testes	Interpretação errada	-	Interpretação correta

**avalia\_c7():**

**pede novos objetos testados e a predição do modelo ao estudante**  
**pede quantidade de acertos do modelo ao estudante**  
**pede se o modelo está funcionando corretamente ao estudante**

**itera nos arrays de objetos preditos e objetos testados, comparando os itens por posição. se iguais, soma em uma pontuação global**

**retorna**

**se pontuação == quantidade de acertos**

**se pontuação == quantidade de objetos e estudante diz que o modelo está funcionando -> 2**

**senão**

**se estudante diz que o modelo não está funcionando -> 2**

### 5.3.2 Uso da automação

O processo de instalação da biblioteca em um Jupyter Notebook e um exemplo resumido de uso são apresentados a seguir.

```
1. # Instala o pacote
2. pip install jupidetector
3.
4. # Importa as funções
5. from jupidetector import ObjectDetection, evaluate, write_json,
get_ninja_robot, PDFWriter
6.
7. # Cria um novo objeto ObjectDetection
8. jupidetector = ObjectDetection()
9.
10. # Defina os atributos do ObjectDetection...
11.
```

```
12. # Avalia o treinamento da detecção de objetos
13. score = evaluate(jupidetect)
14.
15. # Salva JSON com as informações do treinamento
16. write_json(jupidetect)
17.
18. # Salva imagem do ninja robô em './ninja_robot.png'
19. get_ninja_robot(score['total_score'])
20.
21. # Salva PDF com as informações do treinamento e questões de
interpretação
22. # recebendo como parâmetros o caminho da imagem da logo, distribuição
de
23. # dados e matriz de confusão
24. writer = PDFWriter('logo.png', 'dataset_distribution.png',
25. 'tl_confusion_matrix.png')
26.
27. writer.write(jupidetect)
```

### 5.3.3 Apresentação da avaliação

São utilizados os ipywidgets para apresentar os resultados da avaliação no Jupyter Notebook. Como resultado é apresentada a pontuação obtida para cada um dos critérios da rubrica (Tabela 20). É apresentada também a nota final calculada a partir das pontuações conforme definido no Modelo de Medição.

De forma lúdica, junto com a pontuação do treinamento do modelo é apresentado um mascote “ninja-robô” (ALVES et al., 2019), incentivando o engajamento dos estudantes para melhorar seu resultado e obter novas faixas. Os “ninja-robôs” possíveis de alcançar são apresentados na Figura 14.

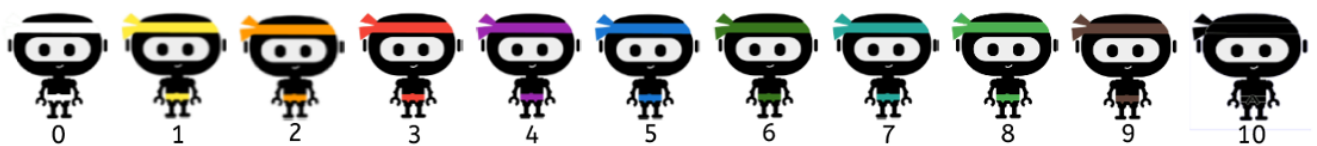


Figura 14: Escala de “ninja-robôs” conforme a pontuação



Ok! A nota do seu treinamento do modelo de detecção de objetos foi 5.



Figura 15: Exemplo de apresentação da avaliação para a nota 5

### 5.3.4 Geração do relatório do modelo treinado

É disponibilizado também no pacote do jupidetect, um módulo para efetuar a construção de um Modelo de Documentação, contendo as informações do modelo em formato PDF. Para utilizar o módulo, basta inserir algumas informações extras no objeto do ObjectDetection. O relatório do modelo treinado é exportado conforme a Tabela 22.


```
1. jupidetect.model_name = 'Modelo Teste'
2. jupidetect.model_date = '24/02/2022'
3. jupidetect.model_version = 'v1.0.0'
4. jupidetect.model_task = 'Detectar objetos domésticos em imagens, focado em 10 objetos domésticos específicos'
5. jupidetect.model_use_context = 'O modelo é utilizado como exemplo no contexto de ensino de na Educação Básica. Este modelo não foi treinado para ser utilizado em pesquisa na área de botânica.'
6. jupidetect.model_target_audience = 'Cidadãos (8+ anos), Foco em alunos do Ensino Médio'
7. jupidetect.model_risks = 'Detectar objetos erroneamente'
8. jupidetect.dataset_augmentation_size = '480x480 pixels'
9. jupidetect.dataset_description = 'Conjunto de imagens de objetos domésticos (tipicamente em ambiente doméstico (sala, quarto, escritório, etc)).'
10. jupidetect.dataset_origin = 'Conjunto de dados de objetos domésticos disponibilizado pela CnE'
```

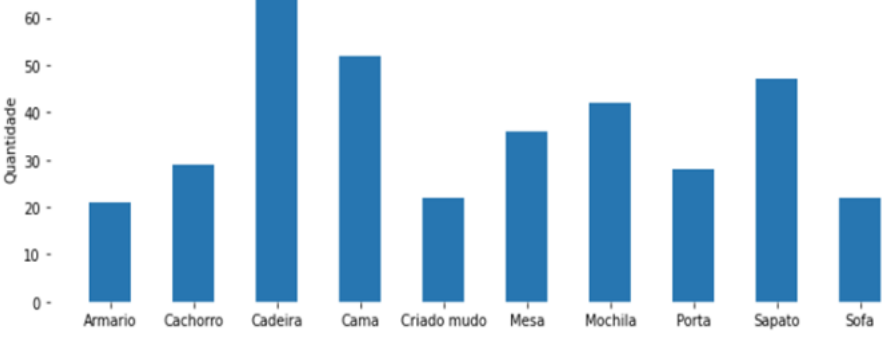
```

11. jupidetector.dataset_labeler_name = 'Por estudantes (conjunto de dados CnE)'
12. jupidetector.tl_batch_size = '128'
13. jupidetector.ft_epoch = '300'
14. jupidetector.ethics_limitations = 'Esse modelo é limitado a somente 10 objetos de ambiente doméstico com um desempenho aceitável. Os resultados da detecção de objeto devem ser utilizados com cuidado sempre revisado por humanos.'
15. jupidetector.ethics_considerations = 'N/A'
16. jupidetector.author = 'Nome do estudante'

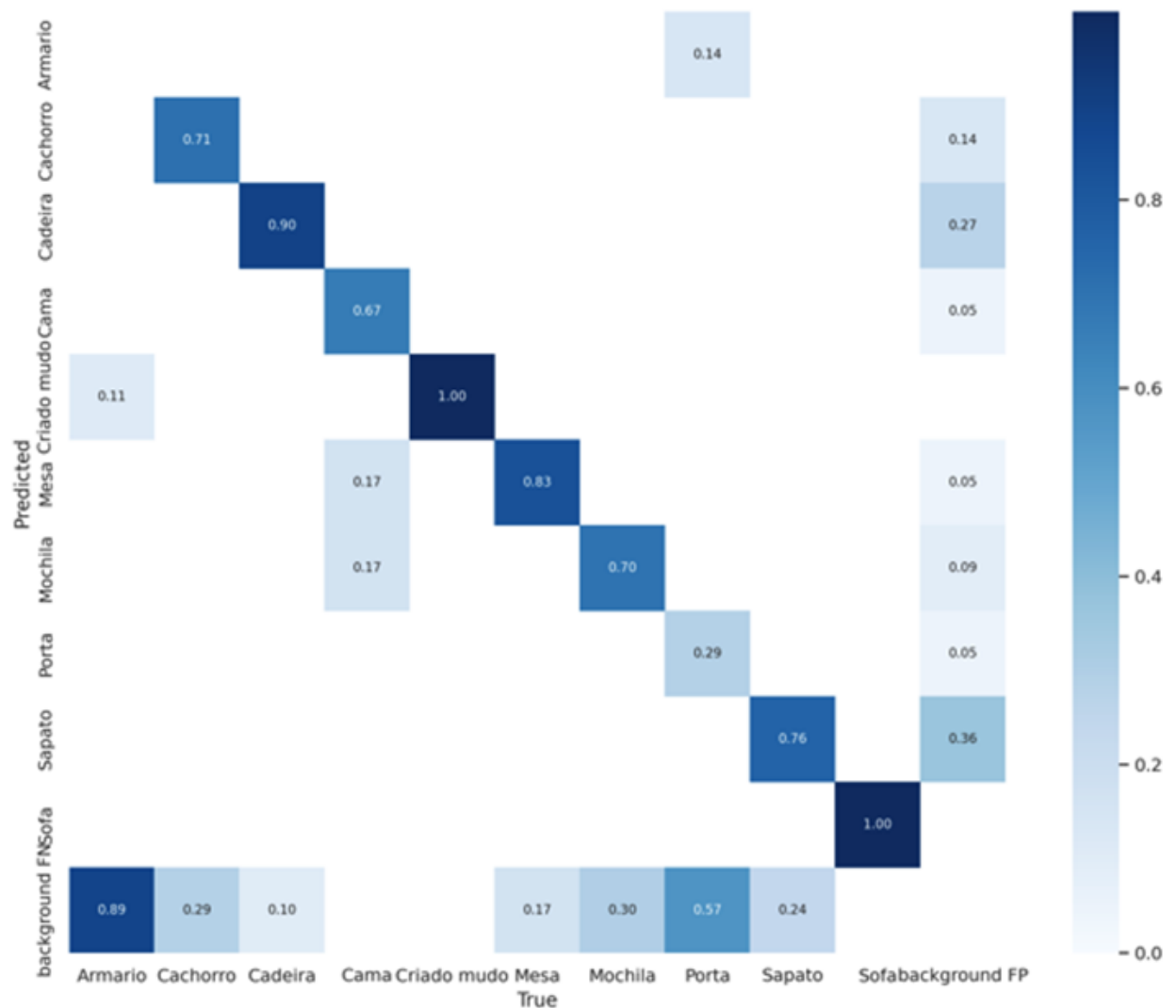
```

Tabela 22: Cartão de Modelo do modelo.

<div style="text-align: right;">  </div> <div style="text-align: left;"> <h2>Cartão de Modelo</h2> </div>	
<b>Nome do modelo</b>	Modelo Teste
<b>Data</b>	24/02/2022
<b>Versão</b>	v1.0.0
<b>Objetivo do modelo de ML</b>	
<b>Tarefa</b>	Detectar objetos domésticos em imagens, focado em 10 objetos domésticos específicos
<b>Contexto de uso</b>	O modelo é utilizado como exemplo no contexto de ensino de na Educação Básica. Este modelo não foi treinado para ser utilizado em pesquisa na área de botânica.
<b>Público alvo</b>	Cidadãos (8+ anos), Foco em alunos do Ensino Médio
<b>Riscos</b>	Detectar objetos erroneamente.
<b>Tipo da tarefa</b>	<i>Multi-label</i> detecção de objetos
<b>Categorias</b>	Armário, Cachorro, Cadeira, Cama, Criado mudo, Mesa, Mochila, Porta, Sapato e Sofá
<b>Conjunto de dados</b>	
<b>Descrição dos dados</b>	Conjunto de imagens de objetos domésticos (tipicamente em ambiente doméstico (sala, quarto, escritório, etc.)).
<b>Origem dos dados</b>	Conjunto de dados de objetos domésticos disponibilizado pela CnE
<b>Quantidade total de dados</b>	Total de 500 imagens
<b>Tamanho de imagens</b>	480x480 pixels
<b>Divisão do conjunto de dados</b>	80% para treinamento (400 imagens), 20% para validação (100 imagens)

<b>Distribuição dos dados por categoria</b>	<p style="text-align: center;">Quantidade por categoria</p>  <table border="1" data-bbox="614 235 1508 571"> <thead> <tr> <th>Categoria</th> <th>Quantidade</th> </tr> </thead> <tbody> <tr><td>Armario</td><td>22</td></tr> <tr><td>Cachorro</td><td>30</td></tr> <tr><td>Cadeira</td><td>65</td></tr> <tr><td>Cama</td><td>52</td></tr> <tr><td>Criado mudo</td><td>22</td></tr> <tr><td>Mesa</td><td>36</td></tr> <tr><td>Mochila</td><td>42</td></tr> <tr><td>Porta</td><td>28</td></tr> <tr><td>Sapato</td><td>48</td></tr> <tr><td>Sofa</td><td>22</td></tr> </tbody> </table>	Categoria	Quantidade	Armario	22	Cachorro	30	Cadeira	65	Cama	52	Criado mudo	22	Mesa	36	Mochila	42	Porta	28	Sapato	48	Sofa	22
Categoria	Quantidade																						
Armario	22																						
Cachorro	30																						
Cadeira	65																						
Cama	52																						
Criado mudo	22																						
Mesa	36																						
Mochila	42																						
Porta	28																						
Sapato	48																						
Sofa	22																						
<b>Labeling</b>	Por estudantes (conjunto de dados CnE)																						
<b>Treinamento</b>																							
<b>Tipo de modelo</b>	Yolov5m																						
<b>Tamanho de lotes</b>	128																						
<b>Quantidade de épocas</b>	300																						
<b>Avaliação do desempenho</b>																							
<b>Precisão média total</b>	All: 0.74																						
<b>Precisão média por categoria</b>	Armário: 0.18, Cachorro: 0.74 , Cadeira: 0.88, Cama: 0.94, Criado mudo: 0.99, Mesa: 0.87, Mochila: 0.71, Porta: 0.43, Sapato: 0.64 e Sofá: 0.99.																						

## Matriz de confusão



### Predição

Quantidade de testes realizados

2

### Limitações e considerações éticas

Limitações

Esse modelo é limitado a somente 10 objetos de ambiente doméstico com um desempenho aceitável. Os resultados da detecção de objeto devem ser utilizados com cuidado sempre revisado por humanos.

Considerações éticas

N/A

### Referências

Autores e afiliação

Nome do estudante

### 5.3.5 Disponibilização da automação

A automação será disponibilizada via pacote no PyPI ou também é disponibilizado um Dockerfile que quando utilizado disponibiliza o uso da avaliação por meio de um servidor HTTP, utilizando os arquivos a seguir.

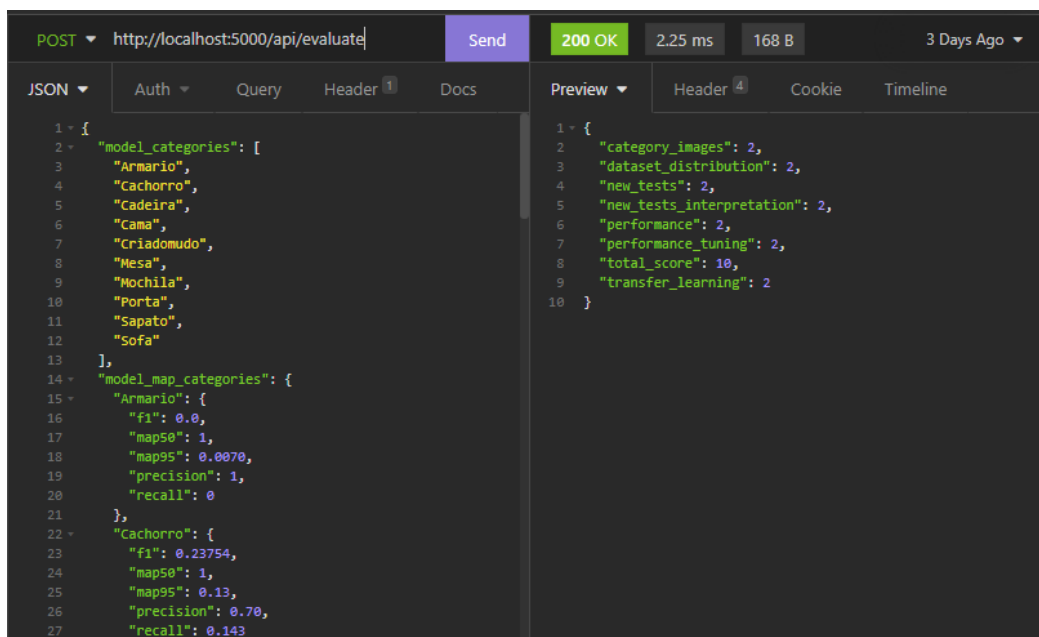
#### Dockerfile

```
1. FROM python:3.8-slim-buster
2.
3. WORKDIR /app
4.
5. COPY requirements.txt requirements.txt
6. RUN pip3 install -r requirements.txt
7.
8. COPY . .
9.
10. CMD ["python3", "-m", "flask", "run", "--host=0.0.0.0"]
```

#### Requirements.txt

```
1. Flask == 2.0.1
2. fpdf = "*"
3. PyPDF2 = "*"
```

Ao executar o container docker, o serviço irá subir um servidor HTTP, disponibilizando uma rota **/api/evaluate**, com isso é possível realizar uma chamada HTTP, com o método **POST**, enviando um objeto **JSON** com os dados do treinamento por meio do **body** da requisição, e a API irá retornar a nota final e as notas por categoria avaliada.



```
POST http://localhost:5000/api/evaluate 200 OK 2.25 ms 168 B 3 Days Ago
JSON Auth Query Header 1 Docs Preview Header 4 Cookie Timeline
1 {
2   "model_categories": [
3     "Armario",
4     "Cachorro",
5     "Cadeira",
6     "Cama",
7     "Criadomudo",
8     "Mesa",
9     "Mochila",
10    "Porta",
11    "Sapato",
12    "Sofa"
13  ],
14  "model_map_categories": {
15    "Armario": {
16      "f1": 0.0,
17      "map50": 1,
18      "map95": 0.0070,
19      "precision": 1,
20      "recall": 0
21    },
22    "Cachorro": {
23      "f1": 0.23754,
24      "map50": 1,
25      "map95": 0.13,
26      "precision": 0.70,
27      "recall": 0.143
```

O pacote possui o nome de `jupidetect` e está disponível conforme apresentado em <http://computacaonaescola.paginas.ufsc.br/jupidetect/>. Para a publicação foi utilizada a biblioteca Poetry (2022), simplificando o processo de geração de versões e upload do pacote no PyPI, utilizando os comandos o arquivo `pyproject.toml` apresentado a seguir.

```
1. # Gera versão, podendo ser 'patch', 'minor' ou 'major'
2. poetry version patch
3.
4. # Publica pacote no PyPI
5. poetry publish --build
6. [build-system]
7. requires = [
8.     "setuptools>=42",
9.     "wheel"
10. ]
11. build-backend = "setuptools.build_meta"
12.
13. [tool.poetry]
14. name = "jupidetect"
15. version = "1.0.0"
16. description = "Jupyter Notebook Object Detection"
17. license = "BSD-3-Clause"
18. authors = [
19.     "pedro.pereira <pedro.p@grad.ufsc.br>"
20. ]
21. readme = 'README.md'
22. repository = "https://codigos.ufsc.br/gqs/jupidetect"
23. homepage = "https://codigos.ufsc.br/gqs/jupidetect"
24.
25. [tool.poetry.dependencies]
26. python = "~2.7 || ^3.2"
27. fpdf = "*"
28. PyPDF2 = "*"
```

### 5.3.6 Tutorial de utilização

Para utilizar o `jupidetect` em um Jupyter Notebook, deve se seguir as seguintes etapas:

1. Instalar a biblioteca do PyPI, importar a classe `ObjectDetection` e as funções desejadas:

```
1. pip install jupidetect
2.
3. from jupidetect import ObjectDetection, evaluate, write_json,
   get_ninja_robot, PDFWriter
```

2. Criar um objeto da classe `ObjectDetection` e preencher seus atributos seguindo o exemplo da seção 5.3.2. Alguns atributos devem ser preenchidos apenas para a geração do PDF, como na seção 5.3.4.
3. Utilizar as funções disponíveis, conforme seção 5.3.3. Os arquivos (PDF e JSON) são salvos no mesmo diretório em que a ferramenta é utilizada.
4. O código fonte será disponibilizado no repositório do PyPI e também nos códigos UFSC.

## 6 CONCLUSÃO

O presente trabalho apresenta o desenvolvimento de um modelo de avaliação de aprendizagem de *Machine Learning* voltado à tarefa de detecção de objetos em imagens, num contexto do Ensino Médio. Durante o desenvolvimento do TCC, foi analisada a fundamentação teórica em relação ao conceito ML, de ensino de ML no Ensino Médio, ML com Jupyter e Python, avaliação de aprendizagem (OE1). Foi levantado também o estado da arte de modelos de avaliação da aprendizagem de ML no contexto do Ensino Médio por meio de um mapeamento sistemático (OE2).

Foi desenvolvido um modelo conceitual para avaliar a aprendizagem de ML no Ensino Médio, criando uma rubrica conforme o estado da arte levantado (OE3). A partir do modelo conceitual, foi criada uma biblioteca automatizando a avaliação com base na rubrica, dentro do Jupyter Notebook, e gerando um resumo do modelo treinado para fins de documentação (OE4). Com isso, espera-se contribuir para o ensino de ML nas escolas Brasileiras, buscando facilitar a avaliação da aprendizagem do aluno e assim contribuir no seu progresso na aprendizagem como um todo.

Para trabalhos futuros pode-se citar a integração da ferramenta com uma interface visual do Jupyter Notebook, a integração da ferramenta em dispositivos móveis e uma melhora validação da rubrica e da fórmula do cálculo de nota aplicado, além da realização de estudos de avaliação mais amplos, explorando um pouco mais da matriz de confusão, explorando mais algumas análises estatísticas.



## 7 REFERÊNCIAS

AAAI. **AAAI Launches “AI for K-12” Initiative in Collaboration with the Computer Science Teachers Association (CSTA) and AI4All, 2018.** Disponível em: <<https://aaai.org/Pressroom/Releases/release-18-0515.php>>. Acesso em: Junho 2021.

AI4K12. **Draft of Big Idea #3 Progression Chart (Learning), 2020.** Disponível em: <[https://drive.google.com/file/d/1QL6l\\_I5cdNTVnYBIZ3\\_Lxur2DgFjmG\\_d/view](https://drive.google.com/file/d/1QL6l_I5cdNTVnYBIZ3_Lxur2DgFjmG_d/view)> Acesso em:

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.

ALVES, G. **Entendendo Redes Convolucionais (CNNs), 2018.** Disponível em: <<https://medium.com/neuronio-br/entendendo-redes-convolucionais-cnns-d10359f21184>>. Acesso em:

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

ANDRADE, S. **Saiba os benefícios da avaliação da aprendizagem escolar e 5 tipos para você aplicar em sua escola.** 2020. Disponível em: <<https://educacao.imagine.com.br/avaliacao-da-aprendizagem-escolar/>>. Acesso em: Julho 2021.

BAULÉ, D. d. S.; GRESSE VON WANGENHEIM, C.; VON WANGENHEIM, A.; HAUCK, J. C. R. **Recent Progress in Automated Code Generation from GUI Images using Machine Learning Techniques.** Journal of Universal Computer Science, 26(9), 2020.

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

BENDECHACHE, M.; TAL, I.; WALL, P.; GREHAN, L.; CLARKE, E.; ODRISCOLL, A.; HAEGEN, L. V. D.; LEONG, B.; KEARNS, A.; BRENNAN, R. **AI in My Life: AI, Ethics & Privacy Workshops for 15-16-Year-Olds.** 2021. Proc. of 13th ACM Web Science Conference 2021. Association for Computing Machinery, New York, NY, USA, 34–39.

BERCHANE, N. **Artificial Intelligence, Machine Learning, and Deep Learning: Same context, Different concepts.** 2018. Disponível em: <<https://master-iesc-angers.com/artificial-intelligence-machine-learning-and-deep-learning-same-context-different-concepts/>>. Acesso em: Julho 2021.

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: Maio 2021.

BLOOM, B. S.; ENGELHART, M. D.; FURST, E. J.; HILL, W. H.; KRATHWOHL, D. R. **Taxonomy of educational objectives: The classification of educational goals.** Handbook I: Cognitive domain. New York: David McKay Company.

BORGES, C. D. B. **Aprendizado de atravessabilidade em imagens aéreas usando redes totalmente convolucionais.** 2021. 84f. Dissertação (Mestrado em Engenharia Elétrica e Computação) – Programa de Pós-Graduação em Engenharia Elétrica e de Computação - Campus de Sobral, Universidade Federal do Ceará, Sobral, 2021.

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: Maio 2020.

BRASSCOM. **Relatório Setorial,** 2019. Disponível em: <<https://brasscom.org.br/relatorio-setorial-de-tic-2019/>>. Acesso em: Maio 2021.

BROWNLEE, J. **Difference Between Algorithm and Model in Machine Learning,** 2020. Disponível em: <<https://machinelearningmastery.com/difference-between-algorithm-and-model-in-machine-learning/>>. Acesso em: Junho 2021.

CARVALHO, L.; SANTOS A.; NAKAMURA F.; OLIVEIRA E. **Detecção precoce de evasão em cursos de graduação presencial em Computação: um estudo preliminar.** In: Anais do Workshop sobre Educação em Computação, Porto Alegre, Brasil, 2019

CARDOSO, M. C. S. A.; SAMPAIO, A. S. F. **Dificuldades para o uso da informática no ensino: percepção dos professores de matemática após 40 anos da inserção digital no contexto educacional brasileiro.** Educação Matemática Pesquisa, 21(2), pp. 044-084, 2019

CETIC. **Escolar urbanas, por número de computadores disponíveis para uso pedagógico,** 2018. Disponível em: <<https://cetic.br/pt/tics/educacao/2018/escolas-urbanas/D33/>>. Acesso em: Março 2021.

CETIC. **Alunos de escolas urbanas, por uso da internet em atividades escolares.** 2019a. Disponível em: <<https://www.cetic.br/pt/tics/educacao/2019/escolas-urbanas-alunos/E1A/>>. Acesso em: Setembro 2021.

CETIC. **Alunos de escolas urbanas, por tipo de computador existente no domicílio.** 2019b. Disponível em: <<https://www.cetic.br/pt/tics/educacao/2019/escolas-urbanas-alunos/B6/>>. Acesso em: Setembro 2021.

CETIC. **Professores de escolas urbanas que participaram de curso de formação continuada sobre o uso de computador e internet em atividades de ensino.** 2019c. Disponível em: <<https://www.cetic.br/pt/tics/educacao/2019/escolas-urbanas-professores/D6B/>>. Acesso em: Setembro 2021.

CHARNIAK, E.; MCDERMOTT, D. **Introduction to Artificial Intelligence.** Addison-Wesley, 1985.

CIEŚLIK, J. **How to Do Data Exploration for Image Segmentation and Object Detection (Things I Had to Learn the Hard Way),** 2021. Disponível em: <<https://neptune.ai/blog/data-exploration-for-image-segmentation-and-object-detection>>. Acesso em: Junho 2021.

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: Abril 2021.

CORDEIRO, A. M.; OLIVEIRA, G. M.; RENTERÍA, J. M.; GUIMARÃES, C. A. **Revisão sistemática: uma revisão narrativa.** Revista do Colégio Brasileiro de Cirurgiões, 34(6), 428-431, 2007.

DOCKER. **Docker.** Disponível em: <https://www.docker.com/>. Acesso em: 10 fev. 2022.

DUTRA, R. **Taxonomia de Bloom: Entenda e aplique em 2021.** 2020. Disponível em: <<https://tutormundi.com/blog/taxonomia-de-bloom/>>. Acesso em: Julho de 2021

FARIA, A. **Conheçam o YOLO v4, o estado da arte em Visão Computacional,** 2021. Disponível em: <<https://www.i2ai.org/content/blog/2021/1/conhecamos-yolo-v4-o-estado-da-arte-em-visao-compu/>>. Acesso em: Junho 2021.

FAYYAD, U. PIATESTSKY-SHAPIO, G.; SMYTH., P. The KDD process for extracting useful knowledge from volumes of data. Communications of the ACM, 39(11), pp. 27-34, 1996.

FIEBRINK, R; GILLIES, M. **Introduction to the Special Issue on Human-Centered Machine Learning.** ACM Trans. Interact. Intell. Syst. 8(2) 2018.

FLASK. **Flask Documentation.** Disponível em: <https://flask.palletsprojects.com/en/2.0.x/>. Acesso em: 10 fev. 2022.

FRANÇA, L. **TECNOLOGIA NA EDUCAÇÃO: COMO GARANTIR MAIS MOTIVAÇÃO EM SALA DE AULA?,** 2021. Disponível em: <<https://www.somospar.com.br/tecnologia-na-educacao-e-motivacao-em-sala/>>. Acesso em: Junho de 2021.

GANDHI, R. **R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms**, 2018. Disponível em: <<https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>>. Acesso em: Junho de 2021.

GARCÍA, J. D. R.; LEÓN, J. M.; GONZÁLEZ, M. R.; ROBLES, G. **Evaluation of an Online Intervention to Teach Artificial Intelligence with LearningML to 10-16-Year-Old Students**. Proc. of the 52nd ACM Technical Symposium on Computer Science Education. Association for Computing Machinery, New York, NY, USA, 177–183, 2021.

GONÇALVES, L., M. **Desenvolvimento de um Curso de Machine Learning com foco em Detecção de Objetos e Design Thinking para Alunos do Ensino Médio**. TCC (Graduação) - Curso de Sistemas de Informação, INE, Universidade Federal de Santa Catarina, Florianópolis, 2022.

GOOGLE. **Arquitetura de um modelo de aprendizado de máquina sem servidor**, 2021a. Disponível em: <<https://cloud.google.com/architecture/architecture-of-a-serverless-ml-model?hl=pt-br>>. Acesso em: Junho 2021.

GOOGLE. **An interactive in-browser visualization of neural networks**, 2021b. Disponível em: <<https://opensource.google/projects/tensorflow-playground>>. Acesso em: Junho 2021.

GRESSE VON WANGENHEIM, C; VON WANGENHEIM, A. **Overview on a human-centric interactive ML process for teaching ML in K-12**. Grupo de Qualidade de Software/INCOD/UFSC, INE, Universidade Federal de Santa Catarina, Florianópolis, 2021.

GRESSE VON WANGENHEIM, C. **Como avaliar a aprendizagem no ensino de computação na Educação Básica?** Palestrante convidado, EduComp - Simpósio Brasileiro de Educação em Computação, Brasil, 2021.

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

GRESSE VON WANGENHEIM, C.; HAUCK, J. C. R.; DEMETRIO, M. F.; PELLE, R. ALVES, N. d. C.; BARBOSA, H.; AZEVEDO, L. F. **CodeMaster – Automatic Assessment and Grading of App Inventor and Snap! Programs**. Informatics in Education, vol. 17, no. 1, 117-150, 2018.

H5P.ORG. **CREATE, SHARE AND REUSE INTERACTIVE HTML5 CONTENT IN YOUR BROWSER**. 2021. Disponível em: <<https://h5p.org/>>. Acesso em: Setembro, 2021.

HAYKIN, S. **Redes Neurais: Princípios e Prática**, 2ª edição. Porto Alegre, RS, Brasil, Bookman Editora, 2007.

HORA, H. R. M.; MONTEIRO, G. T. R; ARICA, J. **Confiabilidade em Questionários para Qualidade: Um Estudo com o Coeficiente Alfa de Cronbach**. Produto & Produção, 11(2), p. 85 - 103, 2010,

IBM. **Capítulo 3 – O Que São Redes Neurais Artificiais Profundas ou Deep Learning?**. Deep Learning Book, 2021a. Disponível em: <https://www.deeplearningbook.com.br/o-que-sao-redes-neurais-artificiais-profundas/>. Acesso em: 28/05/2021

IBM. **Supervised vs. Unsupervised Learning: What's the Difference?**, 2021b. Disponível em: <<https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning>>. Acesso em: Junho 2021.

INEP. 2018. **Notas estatísticas - Censo Escolar 2018**. Disponível em: [https://download.inep.gov.br/educacao\\_basica/censo\\_escolar/notas\\_estatisticas/2018/8/notas\\_estatisticas\\_censo\\_escolar\\_2018.pdf](https://download.inep.gov.br/educacao_basica/censo_escolar/notas_estatisticas/2018/8/notas_estatisticas_censo_escolar_2018.pdf). Acesso em: Setembro 2021.

JAKHAR, D.; KAUR, I. **Artificial intelligence, machine learning and deep learning: definitions and differences**. Clinical and Experimental Dermatology, 45(1), 131–132, 2020.

JEFFCOCK, P. **What's the Difference Between AI, Machine Learning, and Deep Learning?**. 2018. Disponível em: <https://blogs.oracle.com/bigdata/post/whatx27s-the-difference-between-ai-machine-learning-and-deep-learning>. Acesso em: 15/05/2021

JOHNSON, C. FULLER, U. **Is Bloom's taxonomy appropriate for computer science?**. ACM International Conference Proceeding Series. 276. 120-123, 2006.

LARMAN, C.; BASILI, V. R. **Iterative and Incremental Development: A Brief History**. Computer, 36(6), pp. 47-56, 2003.

LENHARDT, T. **Avaliação por rubrica: como esse instrumento pode ajudar na avaliação durante o período de Educação Remota?** 2020. Disponível em: <<https://www.scaffoldeeducation.com.br/avaliacao-por-rubrica-como-esse-instrumento-pode-ajudar-na-avaliacao-durante-o-periodo-de-educacao-remota/>>. Acesso em: Julho 2021.

LECUN, Y.; BENGIO, Y.; HINTON, G. **Deep Learning**. Nature, 521, 436-44, 2015.

LINKEDIN. **Profissões Emergentes**, 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.

LIU, Dawn. . **China ramps up tech education in bid to become artificial intelligence leader**, 2020.National Broadcasting Company Disponível em: <<https://www.nbcnews.com/news/world/china-ramps-tech-education-bid-become-artificial-intelligence-leader-n1107806>> . Acesso em: Março 2021

LÖBLER M. L.; BULEGON L. M. L.; MAYUMI N. J. **Os Laboratórios de Informática em Escolas Públicas e sua Relação com o Desempenho Escolar**. Revista Novas Tecnologias na Educação, vol. 10, no. 3, 2012.

LONG, D.; MAGERKO, B.. **What is AI Literacy?** Competencies and Design Considerations. Proc. of the Conference on Human Factors in Computing Systems, Honolulu, HI, USA, 1-16, 2020.

LUGER, G. F. **Inteligência Artificial: Estruturas e Estratégias para a Solução de Problemas Complexos**. Bookman. 4a edição, Porto Alegre, RS, 2004.

MCCARTHY, J; HAYES, P. J. **Some Philosophical Problems from the Standpoint of Artificial Intelligence**. Machine Intelligence, 4, American Elsevier, New York, NY, USA, 1969.

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, 283–321, 2020.

MARTIN, F.; LEE, I.; SENTANCE, S.; LYTLER, N.; LAO, N. **Extending and Evaluating the Use-Modify-Create Progression for Engaging Youth in Computational Thinking**. Portland, OR, USA. ACM, New York, NY, USA, 2020.

MATHEUS, dos R., A. **Rubrica como ferramenta para a avaliação de habilidades**. 2018. Disponível em: <<http://site.primeiraescolha.com.br/blog-educacao/rubrica-como-ferramenta-para-a-avaliacao-de-habilidades>>. Acesso em: Julho 2021.

MATLAB, I. **Introducing Deep Learning with MATLAB**. The MathWorks Inc. New York, USA, 2017.

MECa. **Uma concepção para o ensino médio**, 2021. Disponível em: <<http://portal.mec.gov.br/pec-g/195-secretarias-112877938/seb-educacao-basica-2007048997/13561-uma-concepcao-para-o-ensino-medio>>. Acesso em: Junho 2021.

MECb. **Base Nacional Comum Curricular**, 2021. Disponível em: <<http://basenacionalcomum.mec.gov.br/>>. Acesso em: Junho 2021.

MISLEVY, R. J.; ALMOND, R. G.; LUKAS, J. F. **A Brief Introduction to Evidence-centered Design**. Educational Testing Service, Research & Development Division, Princeton, NJ, USA. 2003.

MITCHELL, M. et al. **Model Cards for Model Reporting**. arXiv:1810.03993, 2019.

NISHIMOTO, E. B. **Principais conceitos por trás do Machine Learning**, 2018. Disponível em: <<https://medium.com/neuronio-br/principais-conceitos-por-tras-do-machine-learning-a4b942d5d309>>. Acesso em: Junho 2021.

NÚÑEZ, B. N.; ROSEMBERGUE, A. P. M. **A importância da avaliação escolar**. 2019. Disponível em: <<https://jus.com.br/artigos/74544/a-importancia-da-avaliacao-escolar>>. Acesso em: Julho 2021.

PATHMIND, A. **Beginner's Guide to Convolutional Neural Networks (CNNs)**. Disponível em: <<https://wiki.pathmind.com/convolutional-network>> . Acesso em: junho 2020.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. **Systematic Mapping Studies in Software Engineering**. Proc. of the 12th Int. Conference on Evaluation and Assessment in Software Engineering, Bari, Italy, pp. 68-77, 2008

RAGUPATHI, K. **Designing Effective Online Assessments**. Resource Guide - National University of Singapore, Singapore. 2020.

RUSSEL, S.; NORVIG, P. **Artificial Intelligence: A Modern Approach, 3ª edição**. New Jersey, NY, USA, Prentice Hall, 2009.

RICH, E; KNIGHT, K. **Inteligência Artificial**. Makron Books. 2a edição, São Paulo, SP, BR, 1994.

SALVADOR, G. C., GRESSE VON WANGENHEIM, C., LAYDNER, M., RAUBER, M. F. **Resumo de Objetivos de Aprendizagem de Machine Learning na Educação Básica**, Relatório técnico INCOD-GQS.04.2021P, INCOD/INE/UFSC, Florianópolis, Brasil, 2021a.

SALVADOR, G. C. **Desenvolvimento de um Modelo de Avaliação de Aprendizagem de Machine Learning voltado a Classificação de Imagens no Ensino Médio**. TCC (Graduação) - Curso de Sistemas de Informação, INE, Universidade Federal de Santa Catarina, Florianópolis, 2021.

SAMUEL, A. L. **Some Studies in Machine Learning Using the Game of Checkers**. IBM Journal of Research and Development, 44(1-2), pp. 210–229, 2000.

SHU, M. **Deep learning for image classification on very small datasets using transfer learning**, 2019. Disponível em: <<https://lib.dr.iastate.edu/creativecomponents/345>>. Acesso em: Junho 2021.

RIBEIRO, A. da S.; OLIVEIRA, B.; GONÇALVES, L. **O QUE É TEORIA DE RESPOSTA AO ITEM (TRI)?** 2019. Disponível em: <<https://operdata.com.br/blog/o-que-e-teoria-de-resposta-ao-item/>>. Acesso em: Julho 2021.

SOUZA, A. C. de; ALEXANDRE, N. M. C; GUIRARDELLO, E. de B. **Propriedades psicométricas na avaliação de instrumentos: avaliação da confiabilidade e da validade**. *Epidemiologia e Serviços de Saúde*. 2017, v. 26, n. 3 [Acessado 13 Julho 2021] , pp. 649-659. ISSN 2237-9622.

STAMFORD. **Convolutional Neural Networks for Visual Recognition**, 2021. Disponível em: <<https://cs231n.github.io/convolutional-networks/>>. Acesso em: Junho 2021.

STEFANINI. Stefanini Group. **Machine Learning x Deep Learning: entenda a diferença**, 2019. Disponível em: <<https://stefanini.com/pt-br/trends/artigos/machine-learning-vs-deep-learning>> . Acesso em: Maio 2021.

TOURETZKY, D. S. et al. **Envisioning AI for K-12: What Should Every Child Know about AI?** Proc. of the 33rd Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence, Honolulu, HI, USA, 2019a.

TOURETZKY, D. S.; GARDNER-MCCUNE, C.; MARTIN, F.; SEEHORN, D. **K-12 Guidelines for Artificial Intelligence: What Students Should Know**. Proc. of the Int. Society for Technology in Education Conference, Philadelphia, PA, USA, 2019b.

VON WANGENHEIM, A. **Detecção de Objetos em Imagens: Deep Learning**, 2018. Disponível em: <<http://www.lapix.ufsc.br/ensino/visao/visao-computacionaldeep-learning/>>. Acesso em: novembro 2019

VOULGARI, I.; ZAMMIT, M.; STOURAITIS, E.; LIAPIS, A.; YANNAKAKIS, G. **Learn to Machine Learn: Designing a Game Based Approach for Teaching Machine Learning to Primary and Secondary Education Students**. Interaction Design and Children. Association for Computing Machinery, New York, NY, USA, 593–598, 2021.

WARDEN, P. **How many images do you need to train a neural network?** 2017. Disponível em: <<https://petewarden.com/2017/12/14/how-many-images-do-you-need-to-train-a-neural-network/>>. Acesso em: Maio 2021.

WIRTH, R.; HIPPI, J. CRISP-DM: **Towards a standard process model for data mining**. Proc. of the Int. Conference on Practical Applications of Knowledge Discovery and Data mining, Manchester, UK, 2000.

YOO, A. **AI in Education in China. Will it turn students into mindless drones?**, 2020. Disponível em: <<https://www.mathgenie.com/blog/ai-in-education-in-china.-will-it-turn-students-into-mindless-drones>>. Acesso em: Junho 2021.

YOLOv5. **Yolov5**. Disponível em: <<https://github.com/ultralytics/yolov5>>. Acesso em: Fevereiro 2022.

ZOOM. **Jupyter Notebook: o que é e como usar**. 2021. Disponível em: <<https://www.zoom.com.br/pc-computador/deumzoom/jupyter-notebook-o-que-e-como-usar>>. Acesso em: Setembro 2021.



# Desenvolvimento de um Modelo de Avaliação de Aprendizagem de *Machine Learning* no Ensino Médio voltado a Detecção de Objetos

Pedro Paulo Pereira Junior

Departamento de Informática e Estatística  
Universidade Federal de Santa Catarina (UFSC) – Florianópolis – SC – Brasil

pedro.p@grad.ufsc.br

**Resumo.** Machine Learning (ML) é uma tecnologia que está crescendo e se tornando cada vez mais popular dentre as tecnologias atuais. Por isso é muito importante que as pessoas que a utilizam entendam, mesmo que de forma superficial, como essas tecnologias funcionam. Como ML está integrado em vários aspectos da vida das pessoas, é importante que o ensino desses conceitos seja iniciado já na Educação Básica. Nesse contexto, diversas iniciativas estão sendo criadas, porém a maioria não fornece um modelo de avaliação de aprendizagem para ajudar na aplicação desses conceitos, atualmente sendo feita de maneira manual, tomando tempo, esforço e preparação por parte dos instrutores, dificultando a adoção do ensino de ML nas escolas. Por isso, o objetivo deste trabalho é desenvolver um modelo de avaliação de aprendizagem de ML no ensino médio, voltado à detecção de objetos, automatizando esse processo por meio de técnicas de Deep Learning. O modelo de avaliação desenvolvido é integrado em um Jupyter Notebook na plataforma Google Colab no contexto do curso desenvolvido por Gonçalves (2022). Com isto, este trabalho traz uma solução, que consiste em um modelo de avaliação que permite avaliar a aprendizagem de ML, facilitando todo o processo de ensino de ML, contribuindo dessa maneira ao ensino de ML nas escolas brasileiras.

## 1. Introdução

Com o passar dos anos, a utilização de sistemas computacionais no dia a dia das pessoas vem crescendo cada vez mais. Muitas atividades hoje são realizadas por meio de softwares que são desenvolvidos para ajudar, facilitando essas tarefas. Com todo esse impacto e influência da tecnologia é cada vez mais notável a necessidade de que todos tenham conhecimento sobre essas tecnologias, não apenas os profissionais dessas áreas. Inteligência Artificial (Artificial Intelligence - IA) e Aprendizado de Máquina (Machine Learning - ML) são áreas da computação que estão muito presentes na sociedade atual.

Dada essa oportunidade, e a falta de profissionais na área, por exemplo, o governo da China vem implementando o ensino dessas áreas no ensino médio, levando os estudantes a aprender uma compreensão básica de ML. Neste contexto, a iniciativa Computação na Escola visa introduzir a aprendizagem de conceitos de ML no contexto brasileiro, seguindo as diretrizes como K-12 Guidelines for Artificial Intelligence (TOURETZKY, 2019a). E, com este objetivo, são desenvolvidos cursos on-line para ensinar conceitos básicos de ML em relação a várias tarefas, como a detecção de objetos (GONÇALVES, 2022). Esses cursos abordam todo o processo de ML da análise de requisitos até a implantação e questões éticas e dos impactos do ML na sociedade.

Devido sua importância no processo de ensino para auxiliar na progressão da aprendizagem do aluno, precisa-se avaliar seus resultados de aprendizagem e fornecer um feedback informativo e construtivo ao estudante. Porém observa-se que atualmente ainda faltam pesquisas sobre como avaliar a aprendizagem de ML do aluno na Educação Básica. Além disso, levando em consideração as características e restrições típicas na educação básica, necessita-se a automatização deste tipo de avaliação para aumentar a consistência das avaliações além de minimizar o esforço e tempo das avaliações do ponto de vista do professor.

## **2. Metodologia**

A presente monografia é organizada primeiramente com o capítulo de fundamentação teórica, abordando os temas relevantes, como Machine Learning, Ensino de ML e Avaliação da Aprendizagem. O terceiro capítulo tem como objetivo levantar o estado da arte, sendo buscado diversos projetos, a partir do protocolo de busca definido, que possam agregar no presente trabalho. No quarto capítulo é realizada a criação do modelo de avaliação e o desenvolvimento do modelo conceitual. No quinto capítulo é implementada a automação da avaliação da aprendizagem, são definidos também os requisitos funcionais e não funcionais, é apresentada a arquitetura de toda implementação realizada, bem como a implementação em um pseudo código. O último capítulo é referente às conclusões relacionadas a esta monografia e propostos trabalhos futuros.

## **3. Estado da Arte**

Com base no tema deste trabalho, foi realizado um mapeamento sistemático, que tem como objetivo identificar classificar e interpretar pesquisas disponíveis, utilizando critérios de qualificação claros e reproduzíveis (PETERSEN et al., 2008) atualizando a revisão literária feita por Salvador et al. (2021b). A pergunta de pesquisa que este mapeamento procura responder é: quais modelos existem para avaliação de aprendizagem de Machine Learning no Ensino Médio?

Para a realizar a busca de artigos, são utilizado como fonte de pesquisa os artigos indexados pelas ferramentas digitais da área: Scopus, IEEE Xplore, ACM, SpringerLink, ScienceDirect, arXiv, SocArXiv e Google Scholar.

É possível observar a partir dessa revisão que continuam existindo poucas alternativas de avaliação no contexto de ensino de Machine Learning no Ensino Médio, complementando também a pesquisa realizada por Salvador et al. (2021b).

A maioria dessas unidades instrucionais tem como foco cursos introdutórios (Marques et al. 2020), avaliando conceitos básicos de ML, atingindo níveis de aprendizagem mais baixos segundo a taxonomia de Bloom, tendo assim poucas unidades instrucionais que chegam nos níveis mais altos de aplicação de ML. Observa-se também que a maioria destas unidades instrucionais focam em ensinar a tarefa de classificação de imagens, sons etc. sendo esta uma das tarefas consideradas mais fáceis. Desta forma não encontrando também ainda modelos de avaliação no contexto de ensino de outras tarefas, como por exemplo a detecção de objetos. O feedback apresentado aos estudantes é simples, indicando a corretude da tarefa, não focando tanto na avaliação de desempenho com base nos artefatos criados pelos estudantes como resultado da aprendizagem.

Nesse sentido, levando em consideração os dados obtidos, é possível observar a necessidade de melhoria nos modelos de avaliações referente ao ensino de Machine Learning no Ensino Médio, aprofundando os conceitos de Machine Learning e aprimorando as formas de avaliação de

aprendizagem, tornando-as cada vez mais automatizadas para que, futuramente, possam ser utilizadas em larga escala nas escolas brasileiras.

## 4. Modelo de avaliação de aprendizagem

Buscando amenizar os pontos encontrados pelo mapeamento sistemático do Estado da Arte, foi desenvolvido um modelo de avaliação de aprendizagem. Foi desenvolvido também o plano de avaliação, a rubrica e feedback instrucional. Para construir a avaliação é utilizada a Evidence-Centered Design (ECD), uma metodologia sistemática para projetar avaliações visando assegurar que as evidências coletadas e interpretadas sejam consistentes com o conhecimento que a avaliação busca avaliar (MISLEVY et al., 2003).

### 4.1 Modelo de Estudante

De acordo com Mislevy et al. (2003), o modelo de estudante define uma ou mais variáveis relacionadas aos conhecimentos e habilidades que estão sendo avaliadas, sendo utilizadas posteriormente como definição para o que se deseja avaliar. No presente trabalho, essas variáveis são os objetivos de aprendizagem esperados que os alunos atinjam durante o curso (Tabela 1).

Tabela 1 - Objetivos de aprendizagem de Machine Learning para detecção de objetos no Ensino Médio adaptado a partir de (GONÇALVES, 2022).

Identificador	Objetivo de aprendizagem	Taxonomia de Bloom	Fonte
OA1 - Conceitos básicos de ML	Conceitos básicos de ML, saber identificar exemplo de ML	Lembrar	3-A-i (AI4K12, 2020); 3B-AP-08 (CSTA, 2017); 1, 2, 5 (LONG e MAGERKO, 2020)
OA2 - Redes Neurais	Conceitos básicos sobre redes neurais, tipos de arquitetura e seu comportamento	Compreender	3-B-i 6-8, 3-B-i 9-12, 3-B-ii 3-5 (AI4K12, 2020)
OA3 - Conceitos básicos de detecção de objetos	Compreender os conceitos de <i>bounding-boxes</i> , redes neurais convolucionais e algoritmos de detecção de objetos.	Compreender	3-C-ii 6-8, 3-B i 9-12, (AI4K12, 2020)
OA4 - Preparação de dados	Preparar um conjunto de dados, criando as <i>bounding boxes</i> e rotulando os dados	Aplicar	3-C-ii 9-12 (AI4K12, 2020); 3B-AP-09 (CSTA, 2017); 12 (LONG e MAGERKO, 2020)
OA5 - Treinamento de um modelo de detecção de objetos	Treinar um modelo de detecção de objetos a partir de um conjunto de dados	Aplicar	3B-AP-09 (CSTA, 2017); 3-A-iii 9-12 (AI4K12, 2020);
OA6 - Avaliar desempenho do modelo	Avaliar a performance do modelo de ML utilizando um conjunto de dados de teste.	Aplicar	3-A-iii 9-12 (AI4K12, 2020);
OA7 - Implantar o modelo em um aplicativo <i>mobile</i>	Exportar o modelo de detecção de objetos em um aplicativo móvel.	Aplicar	(AMERSHI et al., 2019)
OA8 - Discussão ética	Identificar e descrever problemas éticos chave acerca do ML (privacidade, tomada de decisões éticas, dados enviesados e correlações inesperadas, etc)	Compreender	3-C-iii 6-8 (AI4K12); 3A-AP-24 (CSTA, 2017); 13,16 (LONG e MAGERKO, 2020)

## 4.2 Modelo de evidência

O modelo de evidência fornece instruções de como os resultados das tarefas realizadas pelo aluno devem ser avaliadas, também como essas variáveis deverão ser utilizadas para feedback no nível de tarefa (MISLEVY et al., 2003). A rubrica está sendo adaptada a partir da rubrica definida por Salvador et al. (2021b) voltada à aprendizagem de desenvolvimento de classificação de imagens. Assim foram feitas adaptações para adaptar o modelo de detecção de objetos (marcado em negrito na Tabela 2).

Tabela 2 - Rubrica de avaliação e proposta de automação da avaliação. Adaptado de (SALVADOR et al., 2021b).

ID	Critério	Níveis de desempenho			Técnica de automação
		Baixo - 0 pt.	Aceitável - 1 pt.	Bom - 2 pt.	
<b>Preparação de dados (OA3)</b>					
C1	Quantidade de imagens rotuladas	Menos de 5 imagens por categoria	6 de 10 imagens por categoria	Mais de 10 imagens por categoria	Listar os diretórios e somar a quantidade de arquivos de imagem a partir dos dados enviados ao Jupyter Notebook
C2	Distribuição do conjunto de dados	Quantidade de imagens por categoria varia muito	Quantidade de imagens por categoria varia pouco	Todas as categorias possuem a mesma quantidade de imagens	Listar os diretórios, somar a quantidade de arquivos de imagem e comparar variação por categoria a partir dos dados enviados ao Jupyter Notebook
<b>Preparação de dados (OA4)</b>					
<b>Treinamento de modelo de ML/Transfer Learning (OA5)</b>					
C3	Treinamento	O modelo não foi treinado	O modelo foi treinado com os parâmetros padrão	O modelo foi treinado com parâmetros ajustados (arquitetura, épocas e tamanhos de lotes)	Verificar se o comando de treinamento foi executado, retirar qual foi a arquitetura do comando de definição do learner, retirar qual o número de épocas e tamanhos de lotes do comando de treinamento
<b>Avaliação e interpretação do desempenho de um modelo de ML (Transfer Learning) (OA6)</b>					
C4	Análise do desempenho	Detecções incorretas não identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação incorreta em relação ao modelo	Detecções incorretas identificadas e interpretação correta em relação ao modelo	Verificar se o comando de display de categorias com baixa acurácia foi executado e se respostas no widget de interpretação usando ipywidgets estão corretas
C5	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	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 (OA6)</b>					
C6	Testes com novos objetos	Nenhum novo objeto testado	1-2 novos objetos testados	Mais de dois novos objetos testados	Verificar quantidade de vezes que o comando de treinamento do modelo foi executado com novos objetos

C7	Interpretação dos testes	Interpretação errada	---	Interpretação correta	Verificar se respostas no widget de interpretação usando ipywidgets estão corretas
----	--------------------------	----------------------	-----	-----------------------	--

### 4.3 Modelo de tarefa

O modelo de tarefa descreve o material que será produzido pelo estudante, definindo onde o modelo de avaliação será aplicado. O modelo de tarefa de avaliação pelo desempenho refere-se ao desenvolvimento de um modelo de ML para a detecção de 10 objetos em ambientes domésticos. A definição da tarefa é apresentada de forma detalhada na Tabela 3.

Tabela 3 - Definição do modelo de tarefa.

<b>Objetivo do modelo de DL</b>	
<b>Tarefa</b>	Detectar objetos domésticos em uma imagem (tipicamente em ambiente doméstico como sala, quarto, escritório, etc.).
<b>Tipo da tarefa</b>	<i>Multi-label</i> detecção de objetos
<b>Categorias</b>	10 objetos domésticos Armário, Cachorro, Cadeira, Cama, Criado Mudo, Mesa, Mochila, Porta, Sapato e Sofá
<b>Experiência</b>	Conjunto de imagens de objetos domésticos
<b>Conjunto de dados</b>	
<b>Fonte de dados</b>	Conjunto de dados de objetos domésticos disponibilizado pela CnE
<b>Quantidades de dados</b>	No total de 500 imagens
<b>Padronização das imagens</b>	Formato: .jpeg Tamanho: 480x480 pixels
<b>Desempenho</b>	
<b>Medidas</b>	Precisão média, com o nível de confiança de 50%, de no mínimo 90% total e por categoria. No mínimo 90%

### 4.4 Modelo de documentação

O modelo de documentação tem como objetivo automatizar a documentação das principais características do modelo de ML criado pelo estudante, sendo gerado automaticamente a partir das informações do Jupyter Notebook. O Cartão de Modelo serve como um modelo padrão de documentação, ele centraliza informações sobre um modelo de ML, tendo características de como o modelo foi construído, quais algoritmos foram utilizados, características do conjunto de dados, quais interpretações foram levantadas durante o processo de desenvolvimento, etc (MITCHELL et al., 2019). O Cartão de Modelo a ser montado terá o template conforme figura 1.

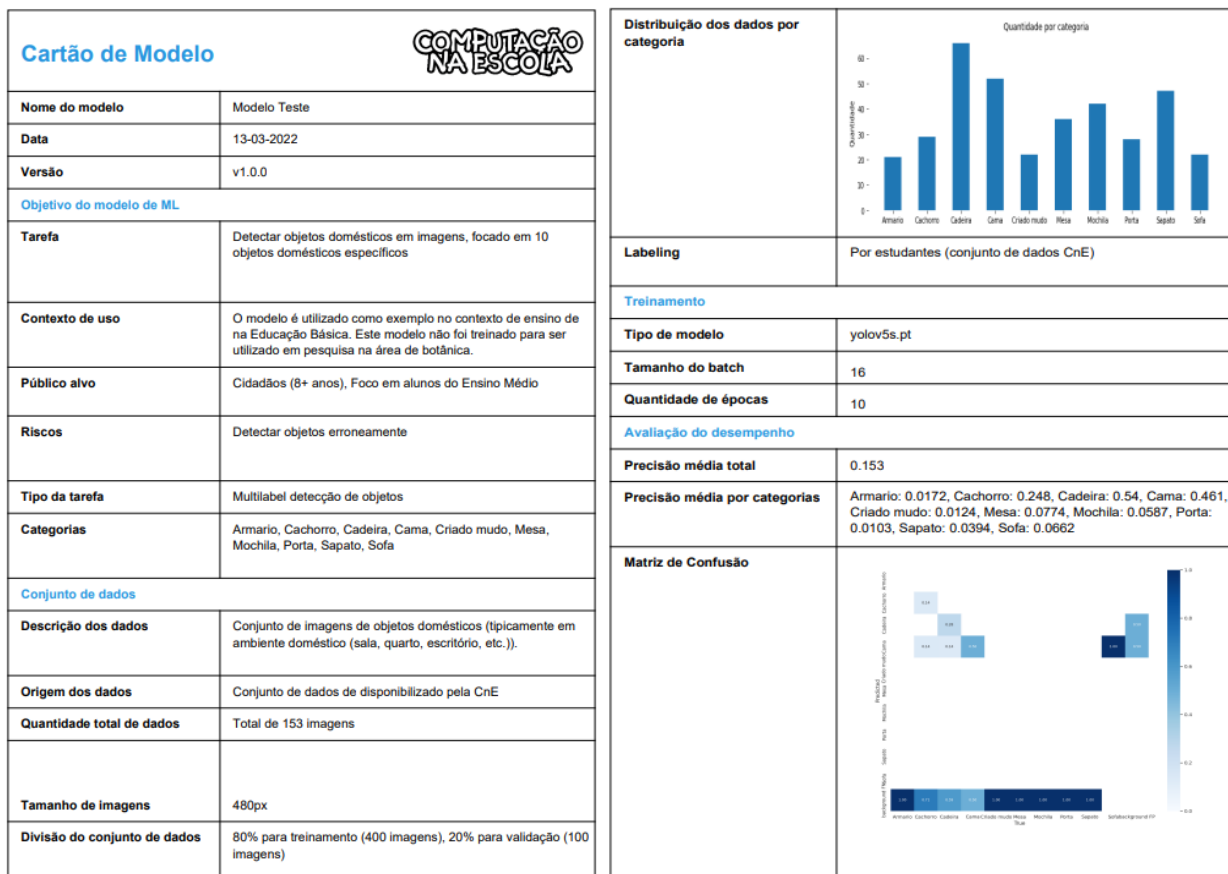


Figura 1. Exemplo de modelo de documentação

## 5. Automação da avaliação de aprendizagem

A partir da análise e modelagem do domínio e do desenvolvimento do framework conceitual, foram realizadas várias iterações para automatizar a rubrica de avaliação, utilizando um processo iterativo/incremental de desenvolvimento de software (LARMAN; BASILI, 2003). A automação da avaliação implementada neste trabalho foi na forma de um pacote Python, chamado jupidetect.

As informações sobre o conjunto de dados, modelo, treinamento do modelo e respostas das questões de interpretação devem ser fornecidas em um objeto Python da classe ObjectDetection. Esse objeto deve estar disponível e com seus atributos preenchidos no código Python do Jupyter Notebook em que será aplicada a automação da rubrica. A partir desse objeto, é possível realizar a avaliação do treinamento seguindo a rubrica proposta no modelo de evidência, gerar um PDF seguindo o modelo de documentação e apresentar visualmente os resultados da avaliação com um mascote “ninja-robô” (Figura 2). O pacote está disponível no repositório PyPI, conforme apresentado em <http://computacaonaescola.paginas.ufsc.br/jupidetect/>. A rubrica de avaliação foi aplicada e testada em um Jupyter Notebook do curso para a detecção de objetos.

Ok! A nota do seu treinamento do modelo de detecção de objetos foi 5.



Figura 2: Exemplo de apresentação da avaliação para a nota 5

## 7. Conclusão

O presente trabalho apresenta o desenvolvimento de um modelo de avaliação de aprendizagem de Machine Learning voltado à tarefa de detecção de objetos em imagens, num contexto do Ensino Médio. Durante o desenvolvimento do TCC, foi analisada a fundamentação teórica em relação ao conceito ML, de ensino de ML no Ensino Médio, ML com Jupyter e Python, avaliação de aprendizagem (OE1). Foi levantado também o estado da arte de modelos de avaliação da aprendizagem de ML no contexto do Ensino Médio por meio de um mapeamento sistemático (OE2).

Foi desenvolvido um modelo conceitual para avaliar a aprendizagem de ML no Ensino Médio, criando uma rubrica conforme o estado da arte levantada (OE3). A partir do modelo conceitual, foi criada uma biblioteca automatizando a avaliação com base na rubrica, dentro do Jupyter Notebook, e gerando um resumo do modelo treinado para fins de documentação (OE4). Com isso, espera-se contribuir para o ensino de ML nas escolas Brasileiras, buscando facilitar a avaliação da aprendizagem do aluno e assim contribuir no seu progresso na aprendizagem como um todo.

Para trabalhos futuros pode-se citar a integração da ferramenta com uma interface visual do Jupyter Notebook, a integração da ferramenta em dispositivos móveis e uma melhora validação da rubrica e da fórmula do cálculo de nota aplicado, além da realização de estudos de avaliação mais amplos, explorando um pouco mais da matriz de confusão, explorando mais algumas análises estatísticas.

## 8. Referências

GONÇALVES, L., M. **Desenvolvimento de um Curso de Machine Learning com foco em Detecção de Objetos e Design Thinking para Alunos do Ensino Médio**. TCC (Graduação) - Curso de Sistemas de Informação, INE, Universidade Federal de Santa Catarina, Florianópolis, 2022.

LARMAN, C.; BASILI, V. R. **Iterative and Incremental Development: A Brief History**. Computer, 36(6), pp. 47-56, 2003.

MISLEVY, R. J.; ALMOND, R. G.; LUKAS, J. F. **A Brief Introduction to Evidence-centered Design**. Educational Testing Service, Research & Development Division, Princeton, NJ, USA. 2003.

MITCHELL, M. et al. **Model Cards for Model Reporting**. arXiv:1810.03993, 2019.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. **Systematic Mapping Studies in Software Engineering**. Proc. of the 12th Int. Conference on Evaluation and Assessment in Software Engineering, Bari, Italy, pp. 68-77, 2008

SALVADOR, G. C., GRESSE VON WANGENHEIM, C., LAYDNER, M., RAUBER, M. F. **Resumo de Objetivos de Aprendizagem de Machine Learning na Educação Básica**, Relatório técnico INCOD-GQS.04.2021P, INCOD/INE/UFSC, Florianópolis, Brasil, 2021a.

SALVADOR, G. C. **Desenvolvimento de um Modelo de Avaliação de Aprendizagem de Machine Learning voltado a Classificação de Imagens no Ensino Médio**. TCC (Graduação) - Curso de Sistemas de Informação, INE, Universidade Federal de Santa Catarina, Florianópolis, 2021.

TOURETZKY, D. S. et al. **Envisioning AI for K-12: What Should Every Child Know about AI?** Proc. of the 33rd Association for the Advancement of Artificial Intelligence Conference on Artificial Intelligence, Honolulu, HI, USA, 2019a.