

**UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ**

**CHRIS HERBERT BERENGUER PEREIRA**

**A COMPUTAÇÃO AFETIVA COMO RECURSO PARA ESTUDOS DE  
USABILIDADE EM PRÁTICAS DE PRODUCT DISCOVERY: UMA ABORDAGEM  
UTILIZANDO MACHINE LEARNING**

**DOIS VIZINHOS**

**2023**

**CHRIS HERBERT BERENGUER PEREIRA**

**A COMPUTAÇÃO AFETIVA COMO RECURSO PARA ESTUDOS DE  
USABILIDADE EM PRÁTICAS DE PRODUCT DISCOVERY: UMA ABORDAGEM  
UTILIZANDO MACHINE LEARNING**

**Affective computing as a resource for usability studies in product discovery practices: an  
approach using Machine Learning**

Trabalho de conclusão de curso de Especialização (TCCE) apresentado como requisito para obtenção do título de Especialista em Tecnologia Python para Negócios da Universidade Tecnológica Federal do Paraná (UTFPR).  
Orientador(a): Marisângela Pacheco Brittes.

**DOIS VIZINHOS**

**2023**



[4.0 Internacional](https://creativecommons.org/licenses/by-nc/4.0/)

Esta licença permite compartilhamento do trabalho, mesmo para fins comerciais, sem a possibilidade de alterá-lo, desde que sejam atribuídos créditos ao(s) autor(es). Conteúdos elaborados por terceiros, citados e referenciados nesta obra não são cobertos pela licença.

**CHRIS HERBERT BERENGUER PEREIRA**

**A COMPUTAÇÃO AFETIVA COMO RECURSO PARA ESTUDOS DE  
USABILIDADE EM PRÁTICAS DE PRODUCT DISCOVERY: UMA ABORDAGEM  
UTILIZANDO MACHINE LEARNING**

Trabalho de Conclusão de Curso/ Especialização  
apresentado como requisito para obtenção do título de  
Especialista em Linguagem Python para Negócios da  
Universidade Tecnológica Federal do Paraná (UTFPR).

Data de aprovação: Vinte e oito de março de 2023

---

Marisângela Pacheco Brittes  
Doutorado  
Universidade Tecnológica Federal do Paraná

---

Pedro Henrique Alencar Machado  
Mestrado  
Universidade Tecnológica Federal do Paraná

---

Simone de Sousa Borges  
Doutorado  
Universidade Tecnológica Federal do Paraná

**DOIS VIZINHOS**

**2023**

## **AGRADECIMENTOS**

Agradeço a todos os professores das disciplinas cursadas por mim ao longo desta trajetória de Pós-Graduação Lato Sensu. Agradeço a todo o corpo docente do curso, aos meus colegas e pessoas que me incentivaram ao longo desta trajetória, em especial, ao Prof. Yuri Kaszubowski Lopes pela paciência e didática em suas aulas; ao Prof. Pedro Henrique Alencar Machado pelas valiosas interações e feedbacks profissionais; e a minha orientadora Dr<sup>a</sup> Marisângela Pacheco Brittes pelo entusiasmo de suas ótimas orientações e pelo apoio em um momento profissional difícil pelo qual passei.

Agradecimentos especiais a Alessandra Elbakyan por tornar o conhecimento acadêmico-científico universalmente acessível. Sem iniciativas como as do Sci-Hub, o meu acesso à literatura internacional teria sido consideravelmente limitado e este trabalho não teria alcançado os resultados que alcançou, qualquer que tenha sido considerada sua relevância no âmbito da pesquisa e das organizações.

## RESUMO

O desenvolvimento de novos produtos de software tem levado tanto startups quanto empresas maduras a buscar por alternativas metodológicas para redução de riscos, orientando etapas do desenvolvimento tecnológico com base em avaliações sistemáticas por parte de seus usuários. Essa maneira de gerenciar a inovação guiada pela verificação empírica de hipóteses de produto tem recebido o nome de *product discovery*, cujas práticas consistem no emprego de técnicas de pesquisa de usabilidade em suas iniciativas. Dentre tais técnicas, há destaque para os testes de usabilidade mediados face a face; porém, esses testes carecem de instrumentos de mensuração e podem reproduzir vieses inerentes à entrevista qualitativa. Para preencher essa lacuna, este estudo apresenta uma solução de classificação de emoções desenvolvida com recursos de Machine Learning para testes de usabilidade mediados, propondo uma aproximação conceitual entre os estudos de usabilidade e o campo da computação afetiva.

**Palavras-chave:** Machine Learning, Computação Afetiva, Product Discovery, Teste de Usabilidade, Pesquisa de Design

## ABSTRACT

The new software product development led both startups and big companies to seek methodological alternatives to reduce risks, guiding stages of technological development based on systematic user evaluations. Product discovery is an innovation managing method guided by empirical verification of product hypotheses and employs usability research techniques in its initiatives. Face-to-face mediated usability tests are one such technique; however, these tests lack measurement instruments and can reproduce inherent biases in qualitative interviews. To fill this void, this study aims to propose an emotion classification solution developed in usability tests, Machine Learning resources were used. Furthermore we propose a conceptual approach between usability studies and the affective computing field.

**Key Words:** Machine Learning, Affective computing, Product Discovery, Usability Test, Design Research

## LISTA DE ILUSTRAÇÕES

<b>FIGURA 1 - Métodos de pesquisa em estudos de usabilidade e suas questões de pesquisa.....</b>	<b>18</b>
<b>FIGURA 2 - Dimensões da pesquisa de usabilidade.....</b>	<b>21</b>
<b>FIGURA 3 - Desdobramentos da Inteligência Artificial.....</b>	<b>27</b>
<b>FIGURA 4 - Exemplos de arquiteturas de uma rede neural profunda.....</b>	<b>28</b>
<b>FIGURA 5 - Estrutura típica de uma CNN.....</b>	<b>29</b>
<b>FIGURA 6 - Camadas CNN com campos receptivos locais retangulares.....</b>	<b>30</b>
<b>FIGURA 7 - Representação das camadas Max Pooling e Average Pooling.....</b>	<b>33</b>
<b>FIGURA 8 - Diagrama de fluxo do modelo de CNN implementado.....</b>	<b>29</b>
<b>FIGURA 9 - Imagens do dataset FER2013.....</b>	<b>35</b>
<b>FIGURA 10 - Arquitetura de um modelo CNN.....</b>	<b>37</b>
<b>FIGURA 11 - Acurácia e Perda do Modelo.....</b>	<b>39</b>
<b>FIGURA 12 - Matriz de Confusão.....</b>	<b>40</b>
<b>FIGURA 13 - Testes com o classificador de emoções.....</b>	<b>41</b>
<b>QUADRO 1 - Testes de Usabilidade Mediados.....</b>	<b>22</b>
<b>QUADRO 2 - Testes de Usabilidade Não Mediados.....</b>	<b>23</b>
<b>QUADRO 3 - Proposição de novo artefato como método de pesquisa em usabilidade.....</b>	<b>42</b>

## **LISTA DE TABELAS**

<b>TABELA 1 - Quantidades de imagens por expressão na base de dados.....</b>	<b>37</b>
--	-----------



## **LISTA DE ABREVIATURAS E SIGLAS**

ANN - Artificial Neural Network

CEO - Chief Executive Officer

CNN - Convolutional Neural Network

CSAT - Customer Satisfaction Score

CUDA - Compute Unified Device Architecture

DBSCAN - Density-Based Spatial Clustering of Applications with Noise

FER - Facial Emotion Recognizer

GPGPU - General Purpose Graphics Processing Unit

GPU - Graphics Processing Unit

GUI - Graphical User Interface

ILSVRC - ImageNet Large Scale Visual Recognition Challenge

KNN - k-Nearest Neighbour

MVP - Minimum Viable Product

NB - Naïve Bayes

NFR - Non-Functional Requirements

NPS - Net Promoter Score

PLN - Processamento de linguagem natural

SM - Similarity Measure

SVM - Support Vector Machines

TUM - Testes de Usabilidade Mediados

UI - User Interface

UX - User Experience

VGG - Visual Geometry Group

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>13</b>
<b>1.1 Objetivos da pesquisa.....</b>	<b>14</b>
<b>1.1.2 Objetivo principal.....</b>	<b>14</b>
<b>1.1.3 Objetivos secundários.....</b>	<b>14</b>
<b>2 PRODUCT DISCOVERY NO DESENVOLVIMENTO DE <i>SOFTWARE</i>.....</b>	<b>14</b>
<b>3 A PESQUISA EM USABILIDADE E SEUS MÉTODOS.....</b>	<b>17</b>
<b>3.1 A pesquisa em usabilidade.....</b>	<b>17</b>
<b>3.2 Os testes de usabilidade.....</b>	<b>22</b>
<b>4 A COMPUTAÇÃO AFETIVA NOS ESTUDOS DE USABILIDADE.....</b>	<b>24</b>
<b>5 FUNDAMENTOS TEÓRICOS DA APLICAÇÃO.....</b>	<b>26</b>
<b>5.1 Fundamentos teóricos da aplicação.....</b>	<b>26</b>
<b>5.2.1 Machine Learning e Deep Learning.....</b>	<b>26</b>
<b>5.2.2 Redes Neurais Convolucionais (CNN).....</b>	<b>27</b>
<b>5.2.3 Camada Convolucional.....</b>	<b>30</b>
<b>5.2.4 Camada Pooling.....</b>	<b>30</b>
<b>6 MATERIAIS E MÉTODOS.....</b>	<b>31</b>
<b>6.1 Design da implementação do classificador de emoções com base em uma CNN.....</b>	<b>33</b>
<b>6.2 Pacotes e bibliotecas utilizados.....</b>	<b>34</b>
<b>6.3 Pré-processamento e descrição do dataset.....</b>	<b>35</b>
<b>6.4 Divisão dos conjuntos para treinamento e validação.....</b>	<b>35</b>
<b>6.5 Arquitetura VGGNET.....</b>	<b>35</b>
<b>6.6 Gerando matriz de confusão.....</b>	<b>39</b>
<b>6.7 Classificador haar cascade.....</b>	<b>40</b>
<b>6 DISCUSSÃO DOS RESULTADOS.....</b>	<b>40</b>
<b>7 CONSIDERAÇÕES FINAIS.....</b>	<b>43</b>
<b>REFERÊNCIAS.....</b>	<b>45</b>

## 1 INTRODUÇÃO

Metodologias de validação de produtos de software têm sido utilizadas por empresas de base tecnológica cujo foco é a inovação incessante (SILVA et. al, 2020; BLANK, 2021). Ao propor a abordagem *Lean Startup*, Eric Ries (2012) aponta para a utilização de um modelo cíclico de gestão de produtos no qual ciclos de construir-medir-aprender auxiliam etapas do desenvolvimento tecnológico. Nesse sentido, tanto as startups quanto as empresas maduras têm utilizado a prática do desenvolvimento de Mínimos Produtos Viáveis, bem como a aplicação de testes não-funcionais para coleta de *feedbacks* de usuários acerca de suas primeiras impressões do produto de software em questão (FELIN, 2020).

Apesar de testes não-funcionais como TUM (Testes de Usabilidade Mediados) se consolidarem no campo empresarial como técnicas para estudos de usabilidade empregados em práticas de *product discovery*<sup>1</sup> (descoberta de produto), quando mediadas face a face essas técnicas carecem de instrumentos de mensuração e podem reproduzir vieses inerentes à análise qualitativa uma vez que contam com a coleta de dados verbais e relatos observacionais (CURCIO, 2019; JHA; MAHMOUD; 2019).

O presente estudo busca fornecer alternativas metodológicas para estudos de usabilidade ao propor o **reconhecimento de emoções** (ver WANG, et. al, 2022) como recurso para compreensão de estímulos ocasionados pela interação do usuário com uma determinada interface. Dessa forma, este estudo recorre à implementação técnica de algoritmos para o reconhecimento de emoções como recursos para estudos de usabilidade, podendo auxiliar o trabalho de Product Owners, Product Managers, Pesquisadores e Designers de Experiência.

Para a operacionalização técnica do reconhecimento de emoções nos testes de usabilidade mediados, utilizar-se-á a linguagem Python em conjunto com as seguintes bibliotecas: OpenCV para processamento de imagens; as bibliotecas Scikit-Learn e Tensor Flow 2 para recursos de Machine Learning (aprendizado de máquina).

A revisão de literatura deste estudo está dividida em quatro partes: a primeira irá discutir os principais conceitos de *product discovery* no desenvolvimento de software, a segunda parte irá apresentar um panorama entre métodos de pesquisa em usabilidade e, na terceira, será discutido o estado da arte no campo da computação afetiva; na quarta parte,

---

<sup>1</sup> Na gestão de produtos, utilizam-se os termos *product discovery* (descoberta) e *product delivery* (entrega). O primeiro é referente às etapas iniciais de um produto em que hipóteses são verificadas por meio de *benchmarking*, pesquisas de mesa e/ou estudos de usabilidade. Já o segundo é referente às etapas de gestão de entregas periódicas do projeto de software. Optamos por manter os termos em inglês em função de manter a convergência com o seu uso no jargão corporativo.

iremos apresentar os conceitos computacionais empregados na implementação de um software de reconhecimento de emoções.

Na seção Materiais e Métodos, iremos apresentar os conceitos de *Machine Learning* e *Deep Learning* (aprendizado de máquina e aprendizado profundo), em seguida, será apresentada a implementação de uma arquitetura de Redes Neurais Convolucionais para desenvolvimento de um classificador de emoções. Por fim, serão apresentadas discussões e possíveis contribuições do campo da computação afetiva para os estudos de usabilidade.

## **1.1 Objetivos da pesquisa**

### 1.1.2 Objetivo principal

Oferecer alternativa à lacuna ocasionada por possíveis vieses em teste de usabilidade mediados, propondo a utilização de solução de *Machine Learning* para auxílio na execução desses testes, sejam eles realizados de forma remota ou presencial.

### 1.1.3 Objetivos secundários

- Fornecer um panorama acerca das práticas de *product discovery*
- Apresentar revisão temática sobre técnicas utilizadas na pesquisa de usabilidade
- Propor o reconhecimento de emoções como método de pesquisa em usabilidade

## 2 PRODUCT DISCOVERY NO DESENVOLVIMENTO DE SOFTWARE

O *product discovery* é uma abordagem conceitual utilizada para criar produtos tomando como base pressupostos de desejabilidade e necessidade dos clientes (CAGAN, 2016). Práticas do *product discovery* geralmente partem da colaboração entre a gestão de produtos, o design de experiência e a engenharia de software na busca por soluções que sejam úteis, utilizáveis e viáveis tecnicamente. Tal abordagem assume premissas tomando como base a exploração de problemas de clientes prováveis (MUNCH, TRIEFLINGER & HEISLER, 2020). Dessa forma, o objetivo do *product discovery* é fornecer insumos factuais acerca de quais produtos, recursos ou serviços devem compor o *product backlog*<sup>2</sup>. Logo, busca-se antecipar a verificação empírica de hipóteses norteadoras do processo de desenvolvimento tecnológico.

A literatura acadêmica tem oferecido poucas orientações acerca de meios de se conduzir práticas de *product discovery*. Ao trazer um panorama da literatura acerca de tais práticas, os estudos de Trieflinger e seus colaboradores (MUNCH, TRIEFLINGER, HEISLER, 2020; TRIEFLINGER, MUNCH, HEISLER e LANG, 2021) discutem a proeminência do *product discovery* na chamada literatura cinza (e.g., postagens em blogs, livros de negócios). Ao analisarem dados desse tipo de gênero textual, os autores identificam que práticas do *product discovery* são utilizadas para mitigar riscos durante etapas iniciais do desenvolvimento de um produto de software, sendo que os principais desafios atribuídos a essas práticas consistem na falta de clareza do problema a ser resolvido e na prescrição de soluções concretas por parte da gestão.

Uma abordagem proeminente em práticas de *product discovery* é a utilização de MVPs (*Minimum Viable Product*) cuja notoriedade é atribuída ao livro *The Lean Startup*, de Eric Ries (2012). Ries propõe um método com base em BMV (*Business Model Validation*) no qual são feitas iterações rápidas em cima de uma versão ‘enxuta’ do produto com o objetivo de se obter uma instrumentação necessária para coleta de dados (SHEPHERD e GRUBER, 2021).

Münch, Fagerholm e Sanchez (2017) destacam a experimentação contínua por meio de MVPs como uma maneira de se obter *insights* confiáveis no qual o design e a gestão atuam desenvolvendo planos de experimento vinculados aos resultados dos testes de MVPs sustentando uma estratégia de negócios adaptável.

---

<sup>2</sup> Product Backlog é um documento geralmente escrito por um Product Owner ou Product Manager contendo uma lista priorizada de itens sobre os quais a equipe de desenvolvimento de software trabalha no decorrer de um projeto (ver AGUIAR e CAROLLI, 2021).

Um outro tópico referente a práticas de *product discovery* diz respeito a sua integração com atividades de ideação (TRIEFLINGER, MUNCH, HEISLER e LANG, 2021). Nesse sentido, ganham notoriedade a abordagem do *Design Sprint*, focada na resolução de problemas complexos por meio da cocriação, prototipação rápida e testes qualitativos com usuários-chave durante cinco dias; e o *Lean Inception*, focada na cocriação e priorização de ideias com o objetivo de se criar um MVP em uma semana (KNAPP et. al, 2016; CAROLI, 2021)

De acordo com a revisão de literatura de Trieflinger, Munch, Heisler e Lang (2021), as motivações para implementação do *product discovery* consistem em: 1) Construir produtos de impacto, por meio de features desejáveis e viáveis; 2) Certificar-se de que o time de produto está alinhado em termos de propósito; 3) Gerenciar riscos adequadamente, evitando construção de produtos que não possuem demandas mercadológicas; 4) Redução de incertezas, permitindo que os times tomem escolhas com base em evidências; 5) Tomar decisões assertivas para a elaboração do roadmap do produto.

Já Blaase (2020), sugere que o processo de *product discovery* consiste na resolução de três questões chave: 1) Um problema real que as pessoas têm e que vale a pena resolver (*problem-solution fit*); 2) Uma demanda de mercado pela solução oferecida (*product-market fit*); 3) Escalabilidade em massa para o produto (*distribution-conversion fit*). Por sua vez, Torres (2020), orienta a elaboração de ciclos contínuos nos quais ocorrem pontos de contato entre clientes e time de desenvolvimento na forma de pesquisas que visam incrementações do produto.

Sobre as práticas do *product discovery* como processo, Munch et. al (2021) constata a convergência de distintas abordagens identificadas na literatura cinza reconhecidas sobre os seguintes pontos:

*Alinhamento entre stakeholders*: esclarecimento dos principais aspectos do produto com as partes interessadas, como o contexto do produto a ser desenvolvido, como ele se alinha à estratégia empresarial e como mensurar seus resultados. Também são discutidas quais as proposições de valor central do produto, bem como o que o diferencia de seus concorrentes diretos e indiretos.

*Compreensão do problema*: Realização de estudos para entender as dores dos usuários, nesta etapa são utilizadas pesquisas de mesa, *surveys* e estudos qualitativos.

*Ideação*: A partir de insights obtidos na etapa anterior, os times são engajados em sessões de brainstorm e priorizações.

*Criação de artefatos:* com base nas ideias discutidas na fase de ideação, são elaborados protótipos testados sob forma de MVPs de baixa ou de alta fidelidade

*Validação dos artefatos:* busca por validação dos artefatos. Nessa etapa, geralmente são utilizadas entrevistas qualitativas ou o uso de MVPs do tipo "Concierge"<sup>3</sup> ou "Mágico de Oz"<sup>4</sup>.

*Refinamento:* incorporação de alterações tomando como base os *feedbacks* de usuários, por vezes reduzindo o conceito do produto e potencialmente começando a planejar o primeiro conjunto de lançamentos.

As práticas de *product discovery* são subsidiadas por técnicas de pesquisa em usabilidade tanto nas etapas de compreensão do problema quanto nas etapas de validação de artefatos (SHEPHERD e GRUBER, 2021). A perspectiva voltada para a prototipação rápida e coleta de *feedbacks* no âmbito do desenvolvimento ágil tem sido referida como Lean UX (*Lean User Experience*), ou seja, a utilização de práticas de design de experiência com o objetivo de validar hipóteses de negócio e/ou de produto (KLEIN, 2013). Essa abordagem vai em contrapartida a concepções *top-down* tradicionais do design nas quais toma-se como partida os métodos de design como uma operacionalização da visão de um Head de Design ou um CEO. Ao invés disso, são feitas pesquisas de usabilidade com o intuito de verificar de várias maneiras uma hipótese sobre o que o usuário realmente deseja e como ele interage com o artefato a ser desenvolvido. Além disso, diferentes níveis de maturidade de um produto ou organização tendem a aplicar técnicas distintas de pesquisa em usabilidade (LERMEN et. al, 2023). Na seção seguinte deste estudo, iremos apresentar um panorama sobre os métodos de pesquisa em usabilidade.

---

<sup>3</sup> O MVP *Concierge* é caracterizado pela sua personalização do atendimento e proximidade junto ao cliente, sendo oferecido a um seletivo grupo de clientes chave. Por ser realizado de forma analógica, torna-se inviável em termos de escalabilidade.

<sup>4</sup> O MVP *Mágico de Oz* possui características semelhantes ao *Concierge* comungando com este a sua dificuldade em se obter escalabilidade. Sua diferença consiste na disponibilização de uma interface do aplicativo ou site pronto e funcional para que os usuários possam realizar os testes, sendo que geralmente as interfaces não estão automatizadas por completo. Dessa maneira, as tarefas no produto são realizadas manualmente por pessoas com o objetivo de obter insights e compreender as necessidades dos usuários.

### 3 A PESQUISA EM USABILIDADE E SEUS MÉTODOS

#### 3.1 A pesquisa em usabilidade

A *User Experience* (UX) é o campo do conhecimento que descreve a experiência holística de um usuário antes, durante e depois de sua interação diante de uma plataforma, produto ou serviço. O aumento do interesse acadêmico pela pesquisa em UX tem se intensificado desde os anos 1980 (LUTHER, et. al, 2020). No âmbito empresarial, a obtenção de uma abordagem exitosa em termos UX em produtos digitais e físicos tende a aumentar a eficiência de processos organizacionais (KNIGHT; DAYMOND e PAROUTIS, 2020).

Na pesquisa acadêmica, o conceito de UX é caracterizado por expressões ambíguas e diversas (por exemplo, “usabilidade” ou “aceitação do usuário”), o que dificulta uma definição precisa. Uma razão para isso é que a usabilidade, ou UI é considerada como sendo um sub-conceito de UX, sendo a primeira um conceito necessário na compreensão da segunda (ELSBACH e STIGLIANI, 2018).

Já a pesquisa de usabilidade é tida como o estudo do comportamento e da motivação dos usuários, se interessando particularmente pela investigação de suas necessidades em um contexto específico, essas pesquisas buscam investigar como as pessoas compreendem e usam determinados artefatos em suas vidas diárias e podem abordar estudos sobre mudanças de aspectos da usabilidade ao longo do tempo (MARSH, 2018). A pesquisa de usabilidade se difere da pesquisa de mercado, que visa descobrir quais são as preferências e desejos do consumidor, sendo que as pesquisas de mercado e de usabilidade têm diferentes pressupostos e visam resultados com características distintas.

A pesquisa de usabilidade tem se utilizado de diferentes métodos para estudos de usuários. Nesse sentido, Marsh (2018) descreve esses métodos como sendo meios para a obtenção de respostas para questões que podem emergir da necessidade gerencial (Figura 1). Tais métodos são definidos como:

**Figura 1 - Métodos de pesquisa em estudos de usabilidade e suas questões de pesquisa**



Cenário	Teste de Usabilidade	Teste de Conteúdo	Card Sorting	Surveys	Entrevistas com usuários	Estudos diários	Validação da arquitetura da informação	Etnografia	Investigação de contexto	Testes A/B	Workshops com Stakeholders	Pesquisa de Guerrilha
Novo conceito na indústria												
Stakeholders podem não concordar												
O CEO teve uma ideia...												
Por que XXX não está funcionando?												
Por que a conversão está baixa?												
Quem são os nossos usuários												
Precisamos saber quem são os nossos usuários												
O que as os dados querem dizer?												
Como os usuários realmente usam esta tecnologia?												
Soluções diversas?												
O que os usuários pensam de XXX?												
O que os usuários estão experienciando?												
Nós precisamos começar de novo												
Mudança na forma como trabalhamos												
A organização não é centrada no usuário												

Fonte: Elaboração própria com base em Marsh (2018)

*Pesquisa de Guerrilha*: é um método rápido, flexível e de baixo custo para a realização de pesquisas; parte de uma abordagem “apenas faça”. Trata-se de uma experimentação ousada em que se pode abordar pessoas solitárias em cafés e espaços públicos e filmá-los rapidamente enquanto eles usam um produto. Pode ser combinado com testes de usabilidade, etnografias curtas e entrevistas contextuais.

*Workshops com stakeholders*: são uma forma de envolver aqueles que têm um interesse no projeto no qual se está trabalhando. “Stakeholders” são pessoas na organização que são essenciais para realizar um determinado projeto, incluindo pessoas com experiência e conhecimento do produto em desenvolvimento. Esse método permite alinhar objetivos, orçamentos, prazo e complexidade de um produto.

*Testes A/B*: Compara duas versões de um mesmo elemento para ver qual delas tem um melhor desempenho. Podem ser comparadas duas páginas da web, mostrando as variantes (A e B) para visitantes semelhantes ao mesmo tempo, auxiliando tomadas de decisão acerca de design e conteúdo. Num teste A/B, é testado apenas um novo componente por vez em que metade dos usuários vê a versão original da página (variável de controle) e metade são mostrados na versão modificada (variável de teste).

*Investigação contextual*: a investigação contextual pode ser descrita como um híbrido de entrevistas com usuários e etnografia; consiste em um método de entrevista

semiestruturada para obter informações sobre o contexto de uso, com o participante sendo observado e questionado enquanto trabalha em seus próprios ambientes.

*Etnografia:* é um campo da ciência que estuda as pessoas e a cultura. Isso também é usado em pesquisas com usuários para estudar pessoas e grupos à medida em que vivem em suas situações cotidianas. O pesquisador busca identificar reações naturais e não-verbais em contextos diários. A abordagem da Etnografia Virtual ou Netnografia também tem ganhado notoriedade nas comunidades acadêmica e profissional (KOZINETS, 2015).

*Validação de arquitetura da informação:* Arquiteturas de informação são estruturas nas quais as informações são planejadas. Uma aplicação proeminente é a organização e rotulagem do conteúdo e da funcionalidade de um site, sendo que a arquitetura da informação abrange tanto aspectos amplos (estrutura geral do site e navegação) quanto os específicos (rotulagem e conteúdo)

*Estudos diários:* Os estudos diários são usados para registrar dados qualitativos durante um período definido de tempo. Nesse método, os participantes completam um diário por conta própria, recebendo uma estrutura para registrar suas observações e dados sobre aspectos particulares em momentos específicos.

*Entrevistas com usuários:* uma investigação qualitativa na qual perguntas visam obter explicações em profundidade e esclarecimento de contexto de uso. A entrevista com usuários pode ser tanto estruturada, semi-estruturada ou aberta. É uma abordagem utilizada proeminentemente nas ciências sociais, sendo utilizada também na pesquisa de mercado, pesquisa de usuários e no estudo das interações humano-computador. Podem ser realizadas tanto individualmente como na forma de grupos focais.

*Surveys:* são pesquisas usadas comumente em pesquisas de mercado e de usuários. Elas têm limitações, porém são baratas para executar e têm potencial para atingir números elevados de amostragem. São utilizadas para alcançar grande volume de pessoas e reunir dados quantitativos e estruturados. Os questionários quantitativos podem ser realizados tanto face a face, por telefone ou questionários online (estas incluem CSAT e NPS).

*Card sorting:* é um método utilizado para entender como as pessoas pensam sobre determinado tema e por meio das associações que elas fazem, agrupando itens que julgam estarem relacionados entre si.

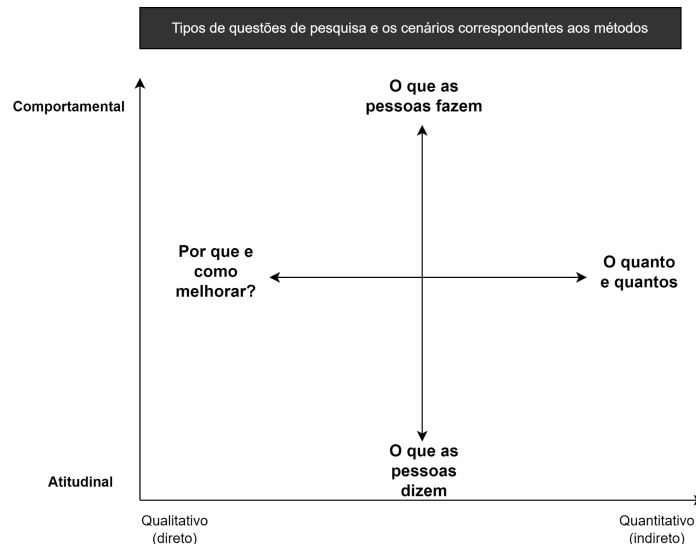
*Testes de conteúdo:* é um tipo específico de teste de usabilidade que se concentra em investigar o quão adequado e compreensível é o seu conteúdo para o público pretendido. Ele

pode ser feito como parte do teste do usuário ou separadamente. Esses testes podem utilizar vários formatos de conteúdo, como imagens, vídeos, infográficos, texto etc.

*Testes de usabilidade:* são formas de avaliar aspectos de usabilidade de um produto em que investiga-se a forma como o usuário realiza tarefas específicas em uma plataforma. Pode ser tanto qualitativo quanto quantitativo, mediado ou não mediado. Iremos discutir esse método com maiores detalhes na seção seguinte. Podem ter suporte de ferramentas com recursos de *eyetracking* e análises de *clickstream*.

De acordo com Rohrer (2022), os métodos de pesquisa em estudos de usabilidade podem ser classificados quanto às suas dimensões atitudinais e comportamentais (Figura 2). Os métodos voltados para avaliar aspectos atitudinais têm como objetivo compreender ou mensurar as crenças declaradas das pessoas, ou seja, busca compreender “o que as pessoas pensam”; sendo comumente utilizadas na pesquisa de marketing. São exemplos desse tipo surveys, testes de conteúdo, entrevistas com usuários, etnografia, pesquisa contextual, estudos diários e Validação de arquitetura da informação.

**Figura 2 - Dimensões da pesquisa de usabilidade**



**Fonte: Adaptado de Rohrer (2022)**

Já os métodos voltados para avaliação de aspectos comportamentais são aqueles que buscam trazer compreensão acerca de "o que as pessoas fazem" com o produto ou serviço em questão. Por exemplo testes A/B apresentam mudanças no design de um site para amostras aleatórias de visitantes do site a fim de se observar o efeito de diferentes escolhas de design sobre o comportamento do usuário, enquanto o recurso de *eyetracking* em testes de

usabilidade não moderados busca gerar compreensão de como os usuários interagem visualmente com determinada a interface. São exemplos de estudos da dimensão comportamental os testes A/B, os testes de usabilidade não mediados e card sorting.

Uma outra característica inerente à pesquisa em usabilidade é a sua utilização na identificação de lacunas nos chamados requisitos não funcionais (*Non-Functional Requirements* - NFRs). Os NFRs descrevem um conjunto de restrições operacionais que um sistema de software deve desempenhar. Essas restrições estão relacionadas à questão da utilidade de um sistema tais como a como usabilidade, confiabilidade, segurança e acessibilidade (JHA e MAHMOUD, 2019). Ao contrário dos requisitos funcionais (que podem ser explicitamente satisfeitos), os NFRs podem ser atendidos parcialmente por meio de medidas funcionais. Por exemplo, a usabilidade pode ser satisfeita usando elementos GUI (*Graphical User Interface*) amigáveis, enquanto a segurança pode ser satisfeita por algoritmos de criptografia e mecanismos de autenticação multifator.

### **3.2 Os testes de usabilidade**

Nos testes de usabilidade, um pesquisador observa usuários desempenhando tarefas ao utilizarem um produto ou serviço com o intuito de mapear problemas na experiência de usuário (MARSH, 2018). Um teste de usabilidade pode ser aplicado tanto de forma qualitativa quanto quantitativa, a primeira ocorre por meio da obtenção de entrevistas, observações ou questionários abertos; já a segunda ocorre por meio do uso de ferramentas de *data tracking*. Além disso, testes de usabilidade podem ser mediados ou não mediados, presenciais ou remotos.

Os testes de usabilidade compõem aspectos de uma etapa de *product discovery*, sendo eles uma forma de avaliar a usabilidade de um produto ou serviço em desenvolvimento, auxiliando na identificação de problemas antes do lançamento de um produto no mercado (MUNCH et. al 2020) . Durante o *product discovery*, busca-se compreender as necessidades, expectativas e comportamentos dos usuários em relação ao produto. Logo, os testes de usabilidade contribuem para a obtenção de *feedbacks* dos usuários em relação a aspectos da usabilidade, fluxo de trabalho e design em geral. Ao medir a facilidade de uso e a satisfação do usuário, é possível identificar oportunidades de melhorias, evitando retrabalhos e desperdício de recursos (KNIGHT et. al, 2020)

Em testes de usabilidade mediados (Quadro 1), há a interação entre o pesquisador e o usuário. Nesse tipo de teste, são gerados dados qualitativos e os moderadores podem responder eventuais perguntas dos participantes, além de poderem tomar notas sobre dúvidas que ocorrem durante a jornada de usabilidade, podendo tirar *insights* a partir da identificação das dificuldades apontadas pelo usuário (MARSH, 2018). Esse tipo de teste ainda pode fornecer uma visão aprofundada da experiência de usuário por meio de conversas com os participantes, permitindo compreender o que eles fazem e como eles se sentem.

**Quadro 1 - Testes de Usabilidade Mediados**

Vantagens	
Controle na obtenção de dados	Permite moderadores interagirem com testadores na obtenção de feedbacks
Adaptabilidade	Pode revelar pontos que não haviam sido considerados pelo moderador, permitindo fazer adaptações nos próximos testes
Baixo custo	Não há necessidade do uso de equipamentos sofisticados
Profundidade dos dados	Permite obtenção de dados com detalhamento por parte dos informantes
Tecnologia assistiva	Possibilita a adaptação do ambiente do teste para pessoas com necessidades especiais
Desvantagens	
Amostragem insuficiente	Não possui amostragem estatisticamente relevante
Viés da interação	Não é recomendado para estudar comportamento natural dos usuários, uma vez que a interação tende a gerar vieses

**Fonte: O autor, tomando como base Marsh (2018)**

Já os testes de usabilidade não mediados são aqueles em que os participantes concluem tarefas definidas sem a presença ou intervenção de um moderador (Quadro2). Quando realizados de forma remota, os participantes podem finalizar a pesquisa a qualquer momento e em qualquer lugar a partir de seus dispositivos, conferindo maior flexibilidade ao participante. Muitos autores sugerem que este tipo de teste é mais imparcial do que o teste mediado (MARSH, 2018).

**Quadro 2: Testes de Usabilidade Não Mediados**

Vantagens	
Ampla amostragem de dados	Por não haver a necessidade de moderação, pode ser aplicado por um número amplo de usuários
Usuários dispersos	Pode suprir dificuldades de agendamento ou acesso presencial
Curto prazo	Pode ser executado em curtos períodos de tempo
Múltiplos contextos	Quando produtos são utilizados em contextos fora de um laboratório ou escritório, como, por exemplo, apps utilizados ao ar livre ou por equipes médicas
Baixo custo	Por serem realizados online, os custos são respectivos ao licenciamento de ferramentas de software utilizadas
Desvantagens	
Socialmente excludente	Exclusão de participantes que não possuem acesso a computadores e internet ou que não possuem letramento digital
Descontextualização dos dados	Por não haver interação de um moderador, os dados tendem a ser apenas descritivos, não sendo recomendados quando se busca obter explicações detalhadas sobre o contexto de uso
Ausência de controle	Os usuários podem finalizar o teste ou passar para uma próxima fase de forma prematura

**Fonte: O autor com base Marsh (2018)**

Os testes de usabilidade não mediados são utilizados quando se faz necessário obter informações de grande número de usuários, gerando dados quantitativos a partir do rastreamento das ações dos usuários para uma ulterior análise. Podem ser utilizados por meio de *wireframes* simples ou protótipos digitais (telas navegáveis), protótipos de baixa e alta fidelidade, produtos inacabados em versão de teste, websites disponibilizados publicamente ou quaisquer outras aplicações testáveis, incluindo artefatos físicos em casos de testes face a face não mediados (MARSH, 2018).

Além disso, os testes de usabilidade podem ser qualitativos ou quantitativos, sendo que os testes qualitativos ocorrem a partir de interações sob forma de mediação ou por meio de perguntas após um teste não mediado. Já os testes de usabilidade quantitativos ocorrem na

sua versão não mediada com a utilização de softwares para análise de clickstream e eyetracking (ROHER, 2022).

A seção seguinte deste estudo, irá discutir o estado da arte no campo da computação afetiva nos estudos de usabilidade (Seção 4). Em seguida, iremos apresentar os fundamentos que irão nortear a o desenvolvimento da aplicação (seção 5) a ser apresentada na seção materiais e métodos (Seção 6), iremos apresentar uma operacionalização técnica das premissas desse campo de estudos na forma de análise de sentimentos para estudos de usabilidade bem como suas implicações para as práticas de *product discovery*.

## 4 A COMPUTAÇÃO AFETIVA NOS ESTUDOS DE USABILIDADE

A computação afetiva é um campo de estudo que reúne tópicos de pesquisa de reconhecimento de emoções e análise de sentimentos, podendo ser realizada utilizando dados unimodais e multimodais. Sendo que as pesquisas realizadas nesse campo consistem na extração de dados a partir de informações físicas (e.g., texto, áudio e visual) ou sinais fisiológicos (e.g., eletroencefalogramas, eletrocardiogramas) (WANG, et. al, 2022).

O conceito de computação afetiva foi proposto inicialmente por Rosalind W. Picard (1997), pesquisadora do MIT (*Massachusetts Institute of Technology*) e tem norteado pesquisas computacionais com o objetivo de identificar emoções ou desenvolver respostas algorítmicas inteligentes a emoções humanas.

O estudo original de Pickford (1997), propôs a quantificação automática e o reconhecimento das emoções humanas como sendo um campo interdisciplinar baseado na psicofisiologia, ciência da computação, engenharia biomédica e inteligência artificial. Logo, consiste no reconhecimento automático da manifestação de emoções humanas usando medidas implícitas que podem ser aplicadas transversalmente a todos os tópicos de comportamento humano.

De acordo com a revisão de literatura apresentada por Wang e seus colaboradores (2022), os modelos para estudos de reconhecimento de emoções podem ser divididos em três grandes abordagens: modelagem emocional, classificação de emoções e eliciação de emoções.

*Modelagem emocional*: essa abordagem pode ser dividida em discreta e dimensional. Modelos discretos caracterizam o sistema emocional como um conjunto de emoções básicas, incluindo raiva, nojo, medo, alegria, tristeza e surpresa, e as emoções complexas que resultam da combinação delas.

*Reconhecimento de emoções*: essa abordagem utiliza sinais biométricos e algoritmos de Machine Learning para classificar emoções automaticamente. São utilizados sinais de expressões faciais, voz, neuroimagem ou sinais fisiológicos. Estudos que utilizam a combinação de recursos fisiológicos com algoritmos de aprendizado de máquina e redes neurais têm alcançado significativos níveis de precisão na inferência dos estados emocionais dos indivíduos.

*Eliciação de emoções*: por sua vez, essa abordagem trata da capacidade de eliciar estados afetivos de maneira confiável e ética. Esta eliciação é um fator crítico para o desenvolvimento de sistemas cujo propósito é detectar, interpretar e se adaptar ao afeto



humano, tendo sido utilizados junto a recursos de música ou narrativas que provocam emoções. Quanto ao seu uso em estímulos audiovisuais, muitos estudos usam o filme para induzir excitação e valência, sendo que esses métodos de eliciação de emoções apresentam limitações respectivas à falta de imersão e interatividade. Tais limitações têm sido supridas por tecnologias de AR (*Augmented Reality*) (KOWALCZUK, 2021).

Dentre os estudos da computação afetiva o reconhecimento de emoções tem ganhado popularidade em função da disponibilidade de bancos de dados públicos; no entanto, esses estudos tornam-se desafiadores na revelação de uma emoção interior em situações nas quais as emoções são propositalmente escondidas de expressões faciais, tons de áudio, gestos corporais, etc. Nesse sentido, sinais fisiológicos tendem a identificar traços emocionais de forma mais precisa e confiável (MARIN-MORALES, et. al 2020).

No âmbito de estudos de usabilidade a compreensão e o reconhecimento das respostas emocionais se mostra crucial para o entendimento do comportamento de usuários. A pesquisa na computação afetiva nesses estudos tem se beneficiado de imagens e vídeos para provocar estados emocionais (ver ANJUM et. al, 2022). Por exemplo, no campo da pesquisa de usabilidade aplicada à educação, o estudo de Wang et. al (2021) evidencia que interações em tempo real aumentam o interesse do usuário pelo aprendizado, proporcionando o aprofundamento da experiência de aprendizado por meio de um sistema interativo de tutoria afetivo-emocional.

Um outro campo de investigação em usabilidade é o da psicologia afetiva na qual se estuda o estado de *flow*: um estado de foco e concentração mental otimizados, geralmente atingido quando o indivíduo lida com atividades desafiadoras e ao mesmo tempo engajantes. Nesse sentido, Arnab et. al (2022) apresentam o uso da computação afetiva como base adaptativa de Inteligência Artificial para *serious games* cujo propósito é promover o estado de *flow* nos usuários.

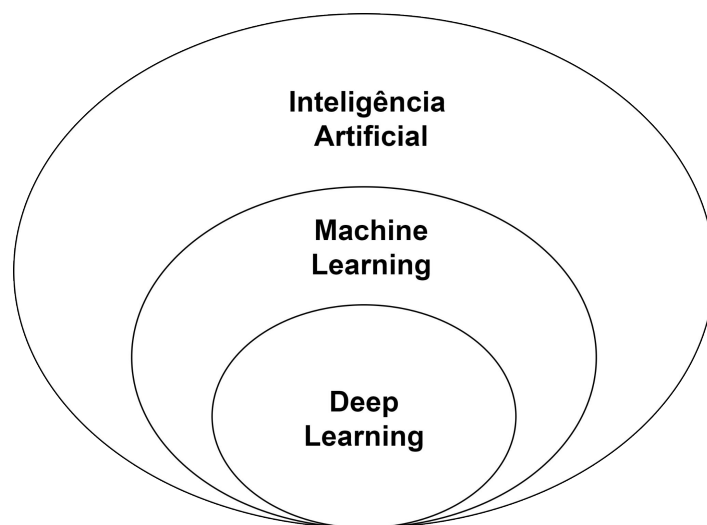
## 5 FUNDAMENTOS TEÓRICOS DA APLICAÇÃO

Nesta seção, iremos apresentar os fundamentos computacionais que guiarão o desenvolvimento de nossa proposta para uma aplicação com reconhecimento de emoções. Detalhes técnicos acerca do desenvolvimento de tal aplicação serão discutidos na seção Materiais e Métodos.

### 5.1 Machine Learning e Deep Learning

O *Machine Learning* (Aprendizado de Máquina) é um subcampo da Inteligência Artificial (Figura 3) e vem se desenvolvendo desde os anos 1950 no qual há o desenvolvimento de sistemas que “aprendem” com dados, apresentando uma otimização de determinada tarefa a partir de alguma medida de desempenho (YINGGE; ALI e LEE, 2020).

**Figura 3 - Desdobramentos da Inteligência Artificial**



**Fonte: Elaboração própria**

De acordo com Gerón (2018) esse campo de estudos se mostra eficiente para: i) o desenvolvimento de soluções para problemas complexos os quais não há solução algorítmica eficaz; ii) substituição de necessidades de ajustes manuais em listas de regras extensas; iii) construção de sistemas com capacidade de adaptação a cenários instáveis; iv) para auxílio de humanos no aprendizado, a exemplo da mineração de dados.

O *Machine Learning* pode ser classificado como supervisionado ou não supervisionado, em batch (por ciclo) ou online (incremental), baseado em instâncias ou em modelos (GERÓN, 2018). No aprendizado supervisionado há um conjunto de dados para treinamento utilizado para gerar classificadores (rotulações). Já no aprendizado não supervisionado os dados de treinamento não são rotulados e emergem a partir de técnicas de clusterização. Há ainda o aprendizado semi supervisionado, que combina características de ambas as técnicas.

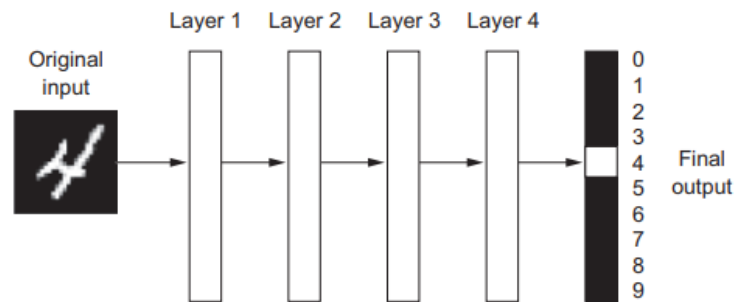
No aprendizado em batch o sistema deve aprender a partir de um fornecimento prévio de dados disponíveis, sendo normalmente realizado offline. Por sua vez, o aprendizado permite o treinamento incremental do sistema, sendo ideal para sistemas que recebem fluxos de dados de forma contínua, a exemplo de sistemas de monitoramento de dados da web.

No aprendizado baseado em instância é uma forma relativamente comum de aprendizado por memorização, fazendo associações entre dados por meio de medidas de similaridade (SM) entre dados. Logo, o sistema aprende por meio de memorização e realiza generalizações a partir da identificação de SM entre os exemplos compreendidos. Por fim, o aprendizado baseado em modelo consiste no fornecimento apriorístico de um modelo de arquitetura para o sistema, no qual predições de dados são geradas a partir de tal modelo.

Exemplos de algoritmos utilizados em soluções de Machine Learning são Classificador Linear, Regressão Logística, Naïve Bayes (NB), Redes Bayesianas, Support Vector Machines (SVM), Árvores de Decisão, Random Forest, AdaBoost, Bootstrapped Aggregation (Bagging), k-Nearest Neighbour (k-NN) e Redes Neurais Artificiais (Artificial Neural Network - ANN) (ASLAM et. al, 2019)

Já o Deep Learning é uma subvertente do Machine Learning, geralmente consiste de uma rede neural com grande número de camadas e parâmetros, sendo comumente utilizadas arquiteturas de redes neurais em suas soluções (Figura 4).

**Figura 4 - Exemplos de arquiteturas de uma rede neural profunda.**



Fonte: Choset (2021)

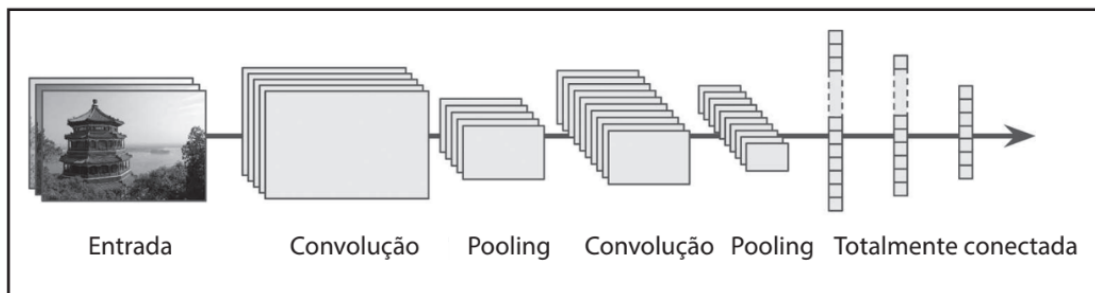
Em suma, o *Deep Learning* emprega modelos de cascata em várias camadas de unidades de processamento não lineares visando extrair características ou transformações (CHOLLET, 2021). As camadas inferiores perto de uma entrada de dados aprendem recursos simples, enquanto as camadas superiores aprendem mais complexos. Dessa maneira, a arquitetura de Redes Neurais forma uma representação de recurso hierárquica, sendo adequada para analisar e extrair conhecimento útil de grandes quantidades de dados e dados coletados a partir de fontes distintas (SHINDE; SHAH, 2018)

## 5.2 Redes Neurais Convolucionais (CNN)

As chamadas Redes Neurais Convolucionais (*Convolutional Neural Network* - CNN) surgiram a partir de estudos sobre o córtex visual do cérebro, tendo sido utilizadas para reconhecimento de imagens desde a década de 1980 (GÉRON, 2018). Graças aos avanços no poder de processamento computacional as CNNs alcançam um desempenho acima da capacidade humana em tarefas visuais complexas, tendo sido utilizadas em serviços de pesquisa de imagens, carros autônomos, sistemas automáticos de classificação de vídeo e outros fins. Além disso, sua utilização se estende a funções de reconhecimento de voz ou processamento de linguagem natural (PLN) (CHOLLET, 2021).

As arquiteturas das CNN consistem no empilhamento de camadas convolucionais (frequentemente seguidas por camadas ReLU), seguidas por uma camada de Pooling e depois adicionadas de camadas convolucionais (+ReLU), seguidas por mais uma camada de pooling e assim sucessivamente (GÉRON, 2018; SHINDE; SHAH, 2018). Logo, a imagem é reduzida proporcionalmente ao seu avanço na rede, tornando-se mais profunda (maior número de mapas de características). Isso ocorre em função das camadas convolucionais (Figura 5).

**Figura 5: Estrutura típica de uma CNN**



**Fonte: Gerón (2018)**

No topo da pilha, é adicionada uma rede neural do tipo *feedforward* (ver TISSOT et.al, 2012), sendo ela composta de por camadas conectadas (+ReLUs) e a camada final gera uma previsão, a exemplo do uso de uma camada *softmax* cuja aplicação gera probabilidades estimadas de classe.

A função de ativação ReLU (ou rede ReLU) é caracterizada por possuir em sua saída sempre uma função linear por partes da entrada. Nas redes ReLU, são dadas todas as funções de rede com arquitetura de rede fixa e o algoritmo escolhido é tipicamente uma versão de gradiente descendente estocástico. Dessa maneira, um método pode ter um desempenho ruim se a classe de função for rico o suficiente ou se o algoritmo selecionar um elemento ruim em sua classe (ECKLE, SCHMIDT-HIEBER, 2019).

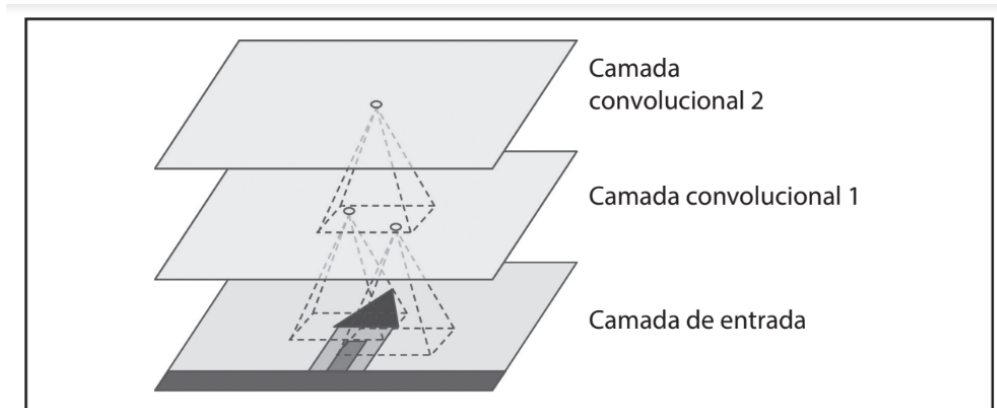
Dois conceitos fundamentais na composição de uma CNN consistem na compreensão acerca de Camada Convolutiva e Camada Pooling, discutidas seguidamente.

### 5.2.3 Camada convolutiva

Uma camada convolutiva é o bloco central de uma CNN. Cada convolução consiste numa operação matemática que sobrepõe suas funções, medindo o cálculo da integral da sua multiplicação pontual (GERÓN, 2018). Na primeira camada convolutiva em uma imagem de entrada, os neurônios não estão conectados a cada pixel, mas somente aos pixels em seus campos receptivos (Figura 6). Já na segunda camada, cada neurônio convolutivo se conecta apenas aos neurônios localizados dentro de um pequeno retângulo da primeira camada. Tal arquitetura possibilita a concentração da rede nas características de baixo nível na primeira

camada oculta, seguida da reunião das características de nível superior e assim sucessivamente (SHINDE; SHAH, 2018).

**Figura 6 - Camadas CNN com campos receptivos locais retangulares**



Fonte: Gerón (2018)

A estrutura hierárquica de uma CNN se assemelha com imagens do mundo real, sendo esse um dos motivos de seu desempenho exitoso no uso de reconhecimento de imagens.

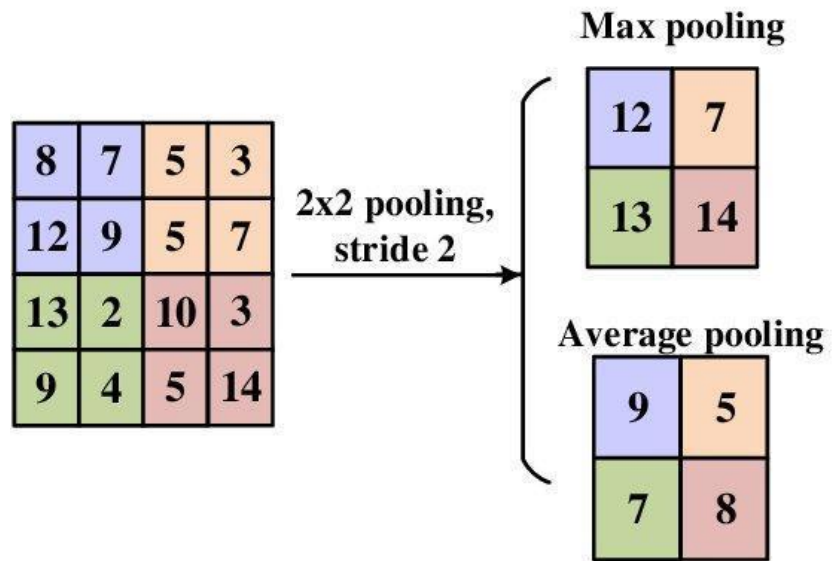
### 5.3 Camada Pooling

As camadas Pooling têm como objetivo subamostrar (reduzir) a imagem de entrada para reduzir a carga computacional (GERÓN, 2018). Como nas camadas convolucionais, cada neurônio na camada pooling é conectado às saídas de um número limitado de neurônios na camada anterior, localizados dentro de um pequeno campo receptivo retangular. Um neurônio de pooling não tem pesos, sendo que ele apenas une as entradas, como a *max pooling*<sup>5</sup> ou *average pooling*<sup>6</sup> (Figura 7)(TEO et. al, 2021).

**Figura 7: Representação das camadas Max Pooling e Average Pooling**

<sup>5</sup> Camadas de Max Pooling, ver [https://keras.io/api/layers/pooling\\_layers/max\\_pooling2d/](https://keras.io/api/layers/pooling_layers/max_pooling2d/)

<sup>6</sup> Camadas de Average Pooling, ver [https://keras.io/api/layers/pooling\\_layers/average\\_pooling2d/](https://keras.io/api/layers/pooling_layers/average_pooling2d/)



Fonte: Teo et. al (2021)

## 6 MATERIAIS E MÉTODOS

A orientação epistemológica que norteia este estudo é caracterizada como pragmatismo, o objetivo de pesquisa deste estudo é a pesquisa de design de natureza primária. Na pesquisa de design busca-se descobrir uma ferramenta ideal que ainda não foi proposta para determinada aplicação (DRESCH; LACERDA; ANTUNES, 2014).

Quanto aos procedimentos técnicos, este estudo se caracteriza como sendo um estudo de caso confirmatório (ou prova de conceito), no qual demonstra-se na prática o funcionamento de uma teoria que se confirma diante do caso elucidado (WASLAWICK, 2016). A confirmação no âmbito do caso é tida como evidência de sua validade e não necessariamente como sendo uma prova de validade.

Sobre a maturidade da pesquisa, essa caracteriza-se como a **apresentação de algo diferente** cujo enfoque é a proposição de um elemento técnico alternativo e sua comparação entre as demais técnicas existentes.

Ao propormos uma prova de conceito de solução em Machine Learning para classificar emoções em processos de *product discovery*, iremos discutir a implementação de um sistema de **aprendizado baseado em instância**, no qual o sistema aprende os exemplos por meio da memorização e depois generaliza em novos casos ao empregar uma medida de similaridade (SM) a fim de compará-los a outros exemplos aprendidos (GÉRON, 2018).

A solução apresentada nas seções seguintes deste estudo consiste na construção de um software de reconhecimento de emoções para estudos de usabilidade em práticas de *product discovery*. Sendo que a implementação técnica da solução irá apresentar partes do desenvolvimento de uma aplicação utilizando Machine Learning para utilização em testes de usabilidade mediados, já que esses carecem de instrumentos de mensuração em suas análises, conforme abordamos em nosso problema de pesquisa na introdução.

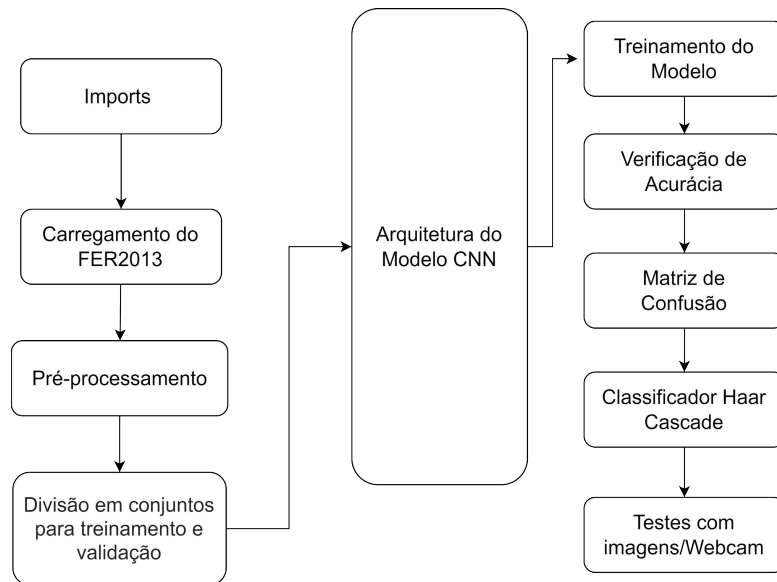
### 6.1 Design da implementação do classificador de emoções com base em uma CNN

Partindo da nossa proposta de pesquisa na qual consiste em fornecer solução tecnológica como alternativa para mensuração em testes de usabilidade mediados em práticas de *product discovery*. Propomos a solução apresentada na Figura 8, na qual aplicamos a fundamentação teórica discutida em tópicos anteriores sob a forma de um artefato tecnológico



como produto da pesquisa de design (WASLAWICK, 2016). As próximas seções irão discutir fundamentos teóricos e procedimentos técnicos utilizados na solução.

**Figura 8: Diagrama de fluxo do modelo de CNN implementado**



Fonte: Elaboração própria

## 6.2 Pacotes e bibliotecas utilizados

Para desenvolver a aplicação proposta neste estudo, utilizamos a linguagem Python e as bibliotecas OpenCV, Numpy, Pandas, Matplotlib e TensorFlow. Keras e Sci-kit Learn.

**Python** é uma linguagem de programação de alto nível para propósito geral, possui tipagem dinâmica e é caracterizada como multi-paradigma, sendo interpretada por script, imperativa, orientada a objetos, funcional e forte (PHYTON.ORG, 2022). Por ser uma linguagem *Open Source* com amplo engajamento da comunidade de desenvolvedores, uma de suas principais vantagens é a quantidade de frameworks e bibliotecas disponíveis para uso.

**OpenCV**(*Open Source Computer Vision Library*) é uma biblioteca de machine learning e visão computacional desenvolvida pela Intel no ano de 2000 livre para fins acadêmicos e comerciais; Possui mais de 2.500 algoritmos otimizados, incluindo recursos de visão computacional clássica de última geração e algoritmos de aprendizado de máquina. Esses algoritmos podem ser utilizados para detecção e reconhecimento de faces, identificação de objetos, classificação de ações humanas em vídeos, rastrear movimentos de câmeras, rastrear objetos em movimento, extrair modelos 3D de objetos, etc (OPENCV, 2022).

**NumPy** é uma biblioteca criada em 2005 de código aberto que possibilita a aplicação da computação numérica com Python. O NumPy é desenvolvido abertamente no GitHub, por meio do consenso do NumPy e da comunidade científica e desenvolvedora do Python (NUMPY, 2022).

**Pandas** é uma biblioteca criada para a linguagem Python para manipulação e análise de dados. Essa biblioteca fornece estruturas e operações para a manipulação de tabelas numéricas e de séries temporais.

**Matplotlib** é uma biblioteca de plotagem de gráficos para o Python. Ela fornece um API orientada a objetos para incorporar gráficos em aplicativos usando kits de ferramentas GUI de uso geral como Tkinter, wxPython, PyQt ou GTK. Nela também há a interface "pylab" baseada em um máquina de estado (como OpenGL).

**TensorFlow** é uma biblioteca de Machine Learning que com código aberto desenvolvida pelo Google e facilita a utilização de Deep Learning e Machine Learning (TENSORFLOW, 2022). Pode ser utilizada por várias linguagens de programação, sendo comumente utilizada com Python. Há possibilidades de utilização de forma normal ou com aceleração de GPU (*Graphics Processing Unit*), sendo que a versão regular pode ser utilizada em qualquer computador, já a versão com aceleração de GPU pode ser utilizada apenas com a utilização de uma placa de vídeo da NVIDIA, uma vez que utiliza o recurso GPGPU (*General Purpose Graphics Processing Unit*) e CUDA (*Compute Unified Device Architecture*). O CUDA é uma extensão feita em linguagem C, que possibilita a utilização de computação paralela para fins de processamento gráfico, possibilitando aumento de performance em operações para este fim.

**Keras** é uma biblioteca Python para Machine Learning que pode ser executado com o TensorFlow. Foi desenvolvida para implementação de modelos de deep learning otimizados para pesquisa e desenvolvimento, podendo ser executado em GPUs e CPUs. É utilizado de forma open-source e disponibilizado pelo MIT.

**Sci-Kit Learn** é uma biblioteca de Machine Learning open-source para a linguagem de Python que inclui vários algoritmos de classificação, regressão e agrupamento incluindo máquinas de vetores de suporte, random forest, gradient boosting, k-means e DBSCAN. Essa biblioteca é projetada para interagir com outras bibliotecas numéricas e científicas do Python.

### 6.3 Pré-processamento e descrição do dataset

Para implementação da solução proposta, utilizamos uma base de dados treinada disponibilizada no site Kaggle (<https://www.kaggle.com/>); no qual utilizaremos a base de dados FER2013 (<https://www.kaggle.com/datasets/msambare/fer2013>).

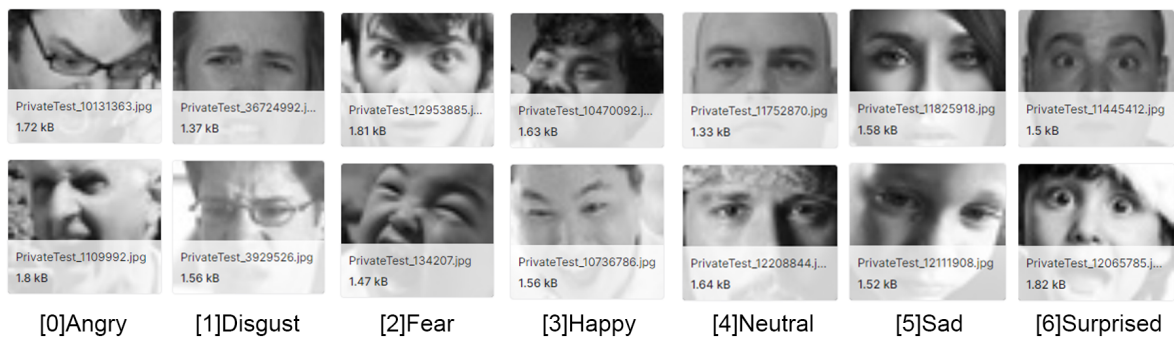
O FER2013 (*Facial Emotion Recognizer*) é um conjunto de dados *open-source* criado por Pierre-Luc Carrier e Aaron Courville, compartilhado publicamente para a competição Kaggle em 2013. Esse conjunto de dados consiste em 35.887 imagens de faces em tom cinza e 48x48 pixels, com o total de 7 emoções, todas elas rotuladas. Os rostos foram registrados automaticamente para que haja uma centralização da face, ocupando aproximadamente a mesma quantidade de espaço em cada imagem. A tarefa é categorizar cada rosto com base na emoção mostrada na expressão facial em uma das sete categorias (0=raiva; 1=nojo; 2=medo; 3=feliz; 4=triste; 5=surpresa; 6=neutro). Na tabela 1, podemos ver um a quantidade de imagens respectivas a cada expressão rotulada no FER2013. Já na figura 9 podemos ver algumas das imagens utilizadas no banco de dados.

**Tabela 1: Quantidades de imagens por expressão na base de dados**

Conjunto de dados FER2013		
% de distribuição	Número de imagens	Expressão
13,82%	4593	Angry (Raiva) [0]
1,52%	547	Disgust (Nojo) [1]
25,11%	5121	Fear (Medo) [2]
16,98%	8989	Happy (Feliz) [3]
14,22%	6077	Sad (Triste) [4]
11,10%	4002	Surprise (Surpresa) [5]
17,25%	6198	Neutral (Neutro) [6]

**Fonte: Elaboração própria.**

**Figura 9: Imagens do dataset FER2013**



Fonte: Elaboração própria

## 6.4 Divisão dos conjuntos para treinamento e validação

Após as importações dos módulos, bibliotecas e do dataset FER2013, dividimos os conjuntos de imagens em dados para treinamento, testes e validação:

Número de imagens no conjunto de treinamento: 29068

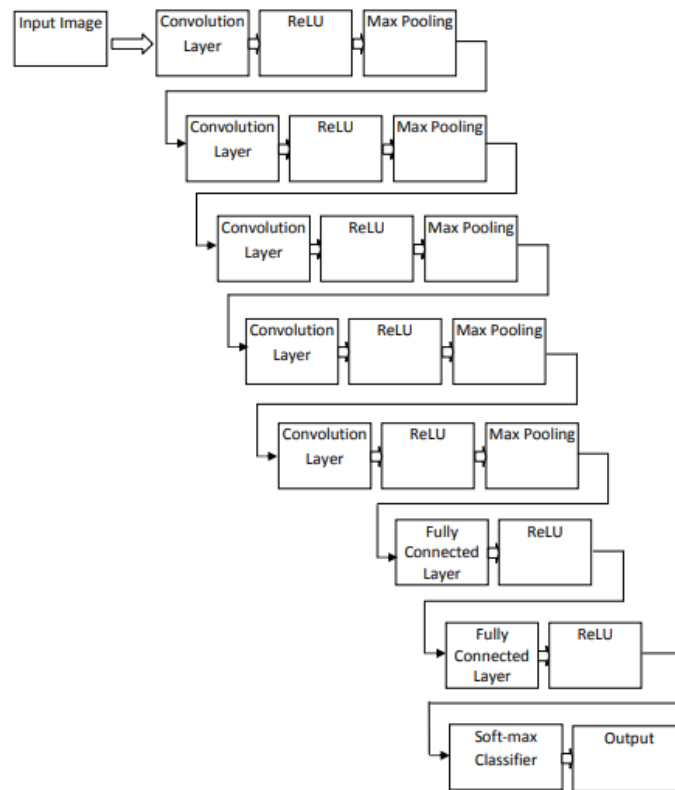
Número de imagens no conjunto de testes: 3589

Número de imagens no conjunto de validação: 3230

## 6.5 Arquitetura VGGNET

Desenvolvida por Karen Simonyan e Andrew Zisserman do laboratório de pesquisa *Visual Geometry Group* (VGG) na Universidade de Oxford, sendo foi vice-campeã ILSVRC (*ImageNet Large Scale Visual Recognition Challenge*) em 2014. Uma arquitetura relativamente simples de ser implementada, possuindo duas ou três camadas de convolução e uma camada de pooling, seguida por 2 ou 3 camadas de convolução, mais uma de pooling e assim sucessivamente, podendo chegar a um total de 16 ou 19 camadas convolucionais, a depender da variação da própria VGG (KUMAR, et. al, 2021). Além disso, é composta por uma rede densa final com 2 camadas ocultas e a camada de saída (Figura 10)(GERÓN, 2018).

**Figura 10 - Arquitetura de um modelo CNN**



**Fonte: Kumar et.al (2021)**

Para a implementação da solução, utilizamos o modelo LittleVGG (RATTAN, 2022) disponibilizado no GitHub, no qual há três sequências de duas camadas de convolução e uma de pooling, totalizando 6 camadas convolucionais. Seguem os detalhes do trecho de código da implementação do modelo:

```

num_features = 32
num_classes = 7
width, height = 48, 48
batch_size = 16
epochs = 100

model = Sequential()

model.add(Conv2D(num_features, (3, 3), padding = 'same', kernel_initializer="he_normal",
                input_shape = (width, height, 1)))
model.add(Activation('elu'))

```

```

model.add(BatchNormalization())
model.add(Conv2D(num_features, (3, 3), padding = "same", kernel_initializer="he_normal",
                input_shape = (width, height, 1)))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.2))

model.add(Conv2D(2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.2))

model.add(Conv2D(2*2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(2*2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.2))

model.add(Conv2D(2*2*2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Conv2D(2*2*2*num_features, (3, 3), padding="same", kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(Dropout(0.2))

model.add(Flatten())
model.add(Dense(2*num_features, kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())
model.add(Dropout(0.5))

model.add(Dense(2*num_features, kernel_initializer="he_normal"))
model.add(Activation('elu'))
model.add(BatchNormalization())

```

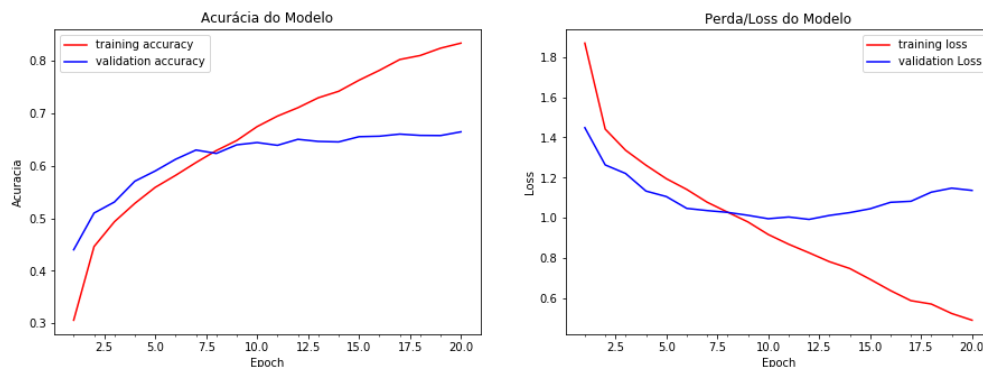
```
model.add(Dropout(0.5))
```

```
model.add(Dense(num_classes, kernel_initializer="he_normal"))
```

```
model.add(Activation("softmax"))
```

Em seguida, partimos para as etapas de compilação e treinamento do modelo, resultando em 29068 amostras de treinamento e 3230 amostras de validação. Em seguida, plotamos um gráfico onde podemos verificar os níveis de acurácia e perda (Figura 11).

**Figura 11: Acurácia e Perda do Modelo**



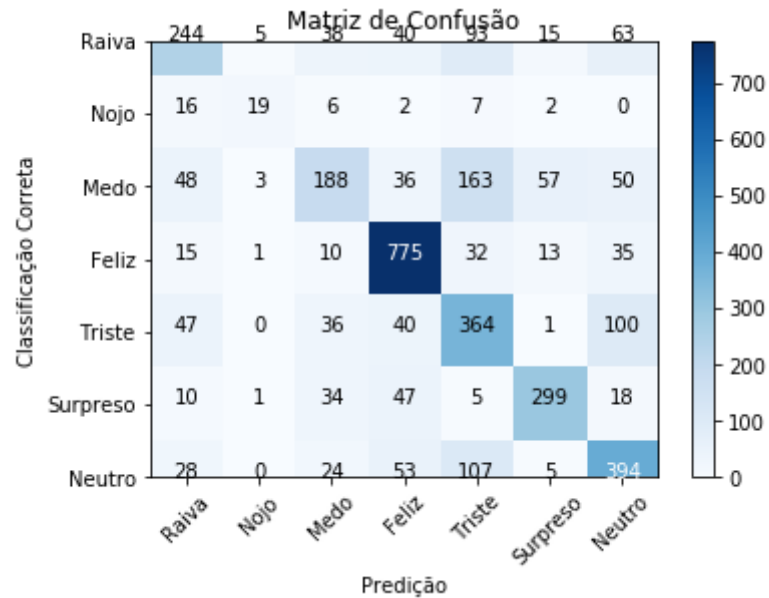
**Fonte: Elaboração própria**

Como podemos observar nos gráficos acima, tivemos os resultados de acurácia: 0.6436333; e perda/loss: 1.23772555968329. Esses resultados de performance do modelo de CNN foram suficientes para a base FER2013 ter se sagrado campeã da competição Kaggle em 2013.

## 6.6 Gerando matriz de confusão

Uma matriz de confusão (ou tabela de erros) é um termo usado em Machine Learning para se referir à tabulação de layout para a elaboração do desempenho algorítmico (KUMAR et. al, 2021; SCI-KIT LEARN, 2022). Geralmente é utilizado para compreender as previsões de diferentes expressões definidas pelo classificador diante de um conjunto de dados fornecidos, no caso, aqui se trata dos conjuntos de dados do FER2013. A Matriz de Confusão trata-se de um tabela em que são mostradas as frequências de classificação para cada classe do modelo. Apresentamos o resultado da Matriz de Confusão gerada no modelo:

Figura 12 - Matriz de Confusão



Fonte: O autor

De acordo com Kumar et. al (2021), podemos interpretar os resultados da matriz de confusão com base na classificação numérica quanto às frequências. Dessa maneira, a matriz denota a precisão de cada expressão no padrão diagonal. A diagonal do canto superior esquerdo para o canto inferior direito define os valores de precisão, ou seja, aqueles que foram corretamente identificados pelas imagens pelo classificador. A diagonal do canto superior direito para o canto inferior esquerdo define as imagens que foram ignoradas pelo classificador.

## 6.7 Classificador Haar Cascade

Proposto por Paul Viola e Michael Jones (2001) em seu artigo intitulado "*Rapid Object Detection using a Boosted Cascade of Simple Features*" e também conhecido como modelo Viola-Jones, o Haar Cascade trata-se de um algoritmo utilizado para identificação de objetos em imagens ou vídeos (OPENCV, 2022). É uma abordagem de Machine Learning na qual grandes volumes de imagens positivas e negativas são utilizados para treinar um classificador. Além disso, o classificador Haar Cascade permite a detecção de objetos em

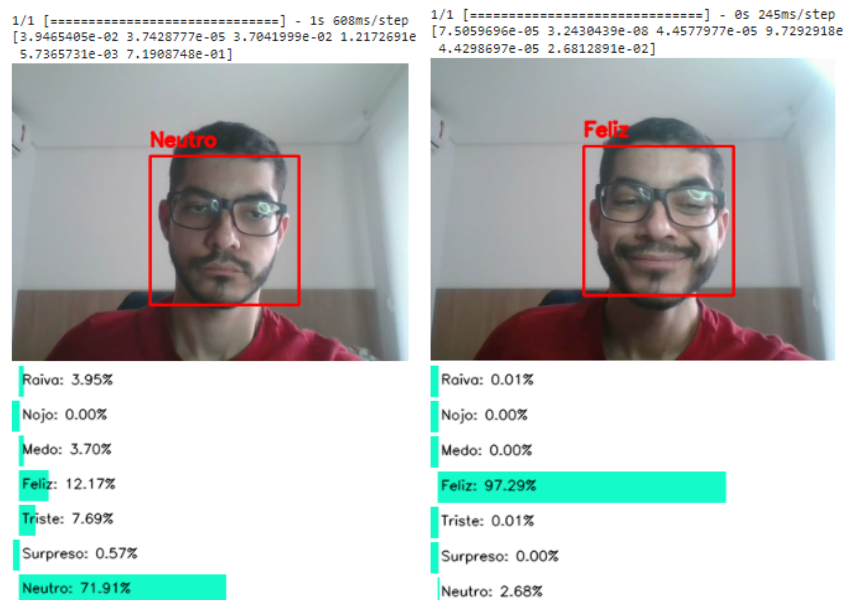


tempo real (KUMAR, et. al, 2021). Dessa forma, utilizamos neste estudo o Haar Cascade a partir da biblioteca OpenCV para detectar a face de uma pessoa e classificar suas emoções.

## 7 DISCUSSÃO DOS RESULTADOS

Os testes com a implementação da aplicação de software proposta neste estudo revelaram-se promissores no sentido de que a acurácia tende a ser relativamente alta quando as expressões se assemelham àquelas presentes na base de dados (Figura 13). Entretanto, há limitações significativas no modelo, dentre as quais se destacam o fato de que o detector do Haar Cascade tende a ter maior eficácia para imagens frontais do rosto; isso é potencializado também pelo fato da base FER2013 possuir apenas imagens frontais, o que torna o modelo inviável para imagens de rostos em outras posições.

**Figura 13: Testes com o classificador de emoções**



Fonte: Elaboração própria

Uma outra limitação do Haar Cascade consiste na sua propensão para falsos positivos, sendo que o Haar Cascade pode facilmente relatar um rosto em uma imagem quando nenhum rosto está presente. Uma terceira limitação da aplicação consiste na necessidade ocasional de ajustes manuais nos parâmetros de detecção do OpenCV, sendo que há ocasiões nas quais as regiões de uma imagem tornam-se falsamente classificadas como rostos e também há ocasiões em que os rostos são totalmente perdidos.

Há ainda situações nas quais uma determinada emoção é classificada de forma errônea, sendo que a assertividade de suas interpretações está diretamente proporcional à quantidade de imagens em cada categoria de emoção. Por exemplo, há um número significativamente maior de imagens de rostos com expressões Feliz (8989 imagens) e Neutro

(6198 imagens) do que imagens com expressões de Nojo (547 imagens) e Surpresa (4002 imagens) na base FER2013. Isso dificulta a classificação das expressões que possuem menor número de referências na base carregada.

No que concerne à utilização da solução tecnológica apresentada neste estudo como ferramenta em testes de usabilidade mediados, a aplicação pode fornecer insumos para triangulação de dados qualitativos (relatos verbais e observação) e quantitativos (mensuração e categorização das expressões). Os segundos podem servir como indicação de reações do usuário a elementos da interface gráfica inserida no teste, sendo que o conceito de ‘mensuração’ da expressão humana deve ser considerado com cautela; pois, além de haver limitações técnicas de acurácia no modelo, deve-se levar em consideração a complexidade e subjetividade humanas.

Os mecanismos de mensuração apresentados neste estudo fornecem um suporte complementar à interação do moderador no teste de usabilidade mediado. Apesar de suas limitações, tal aparato tecnológico pode fornecer *insights* valiosos durante etapas de *product discovery*.

Dessa forma, propomos aos métodos em estudos de usabilidade o acréscimo da classificação de sentimentos como alternativa metodológica. Conforme elucidado no Quadro 3.

**Quadro 3: Proposição de novo artefato como método de pesquisa em usabilidade**

	Qualitativo	Quantitativo	Comportamental	Atitudinal
Pesquisa de Guerrilha	X			X
Workshops com stakeholders	X			X
Testes A/B		X	X	
Investigação contextual	X			X
Etnografia	X		X	X
Validação de arquitetura da informação	X			X
Estudos diários	X			X
Entrevistas com usuários	X			X
Surveys		X		X
Card sorting	X	X	X	
Testes de conteúdo	X		X	X
Testes de usabilidade não mediados		X	X	
Testes de usabilidade	X			X

mediados				
Classificador de emoções		X	X	

**Fonte: Elaboração própria**

Como podemos observar, a classificação computacional de sentimentos acrescenta dados quantitativos acerca de características comportamentais do usuário, sendo útil para utilização em conjunto com testes de usabilidade mediados. Futuras implicações dessa contribuição do estudo serão discutidas na seção seguinte.

## 8 CONSIDERAÇÕES FINAIS

O presente estudo apresentou a classificação de emoções como uma alternativa metodológica para estudos de usabilidade em práticas de *product discovery*. Para tal objetivo, realizamos uma pesquisa de design na qual desenvolvemos uma prova de conceito de aplicação de software em que são utilizados a linguagem Python com auxílio das bibliotecas Keras e TensorFlow 2 para classificar emoções em uma abstração numérica. Versões futuras dessa aplicação podem incluir otimizações na base treinada ou arquiteturas que levem a uma maior acurácia. Além disso, otimizações de performance poderiam ocorrer por meio de recursos como paralelismo além do desenvolvimento de uma GUI amigável por meio de ferramentas como GTK, Tkinter, Kivy ou PyQt. Uma versão ideal da aplicação contaria com um traqueamento das emoções em tempo real extraíndo suas expressões médias e associando-as a cada interação em uma interface.

Durante o estudo, verificou-se que os testes de usabilidade mediados podem se beneficiar da solução apresentada como uma forma complementar à coleta de dados qualitativos, tendo uma fonte de triangulação entre relatos e mensurações de expressões humanas reconhecidas por meio de Inteligência Artificial, residindo neste fato a principal contribuição do estudo.

O artefato tecnológico apresentado neste estudo poderá beneficiar Product Managers, Designers e pesquisadores em práticas de *product discovery* ao possibilitar compreensão sobre sentimentos humanos diante de interfaces digitais. Além disso, esse artefato poderá auxiliar acadêmicos das áreas sociais e de ciências da saúde. Uma outra contribuição deste estudo reside na aproximação dos estudos da computação afetiva com campos da usabilidade e do desenvolvimento de produtos. Futuros trabalhos podem proporcionar ampliações a esse diálogo por meio de soluções em Machine Learning e Deep Learning.

Limitações do estudo residem em lacunas técnicas de performance, acurácia e da própria dificuldade de estabelecer mensuração a aspectos da subjetividade humana discutidos na discussão dos resultados. Além disso, em termos de estudos de usabilidade o método aqui proposto possui limitações referentes a: i) custos elevados: uma vez que há a necessidade de utilização de computador, podendo ser socialmente excludente para indivíduos que não possuem acesso a tecnologias; ii) baixa escalabilidade: uma vez que a solução aqui proposta foca em testes de usabilidade mediados; iii) Vieses de interação: no caso de utilização em testes de usabilidade moderados, as interações com o mediador podem gerar reações que não necessariamente foram ocasionadas por uma interação com a interface; e, iv) possíveis vieses

de etnicidade: cabe uma verificação da base de dados forma a garantir que sua eficiência ocorra diante de múltiplos grupos étnicos, uma vez que bases de dados de imagem tendem a ter padrões predominantemente ocidentais e brancos.

Considerações éticas da pesquisa em pessoas também são pertinentes ao se utilizar a solução proposta neste estudo, uma vez que protocolos de ética devem incluir termos de permissão do uso de imagem do usuário para fins de estudo.

## REFERÊNCIAS

- AGUIAR, F.; CAROLI, P. **Product Backlog Building**. Rio de Janeiro: Editora Caroli. 1ª ed. 2021.
- ANJUM, T.; LAWRENCE, S; SHABANI, A. Augmented Reality and Affective Computing on the Edge Makes Social Robots Better Companions for Older Adults. *In: ROBOVIS*. p. 196-204. 2021.
- ARNAB, S., PETRIDIS, P., KARAVIDAS, L., APOSTOLIDIS, H., & TSIATSOS, T. Usability Evaluation of an Adaptive Serious Game Prototype Based on Affective Feedback. *Information*, v.13, n.425, p.1–13, 2022. Disponível em: <https://doi.org/10.3390/info>. Acesso em: 10 fev, 2023.
- BLANK, S. **Do Sonho À Realização Em 4 Passos: Estratégias para Criação de Empresas de Sucesso** (1st ed.). São Paulo: Alta Books. 2021.
- CAGAN, Marty. **Inspired: How to create tech products customers love**. John Wiley & Sons, 2017.
- CAROLI, Paulo. **Lean inception**. São Paulo: Caroli.Org, 2017.
- CHOLLET, Francois. **Deep learning with Python**. Simon and Schuster, 2021.
- CURCIO, K., SANTANA, R., REINEHR, S., & MALUCELLI, A. Usability in agile software development: A tertiary study. *In: Computer Standards and Interfaces*. Vol. 64, pp. 61–77. 2019. Disponível em: <https://doi.org/10.1016/j.csi.2018.12.003>. Acesso em: 10 fev, 2023.
- DRESCH, A., LACERDA, D. P., ANTUNES, J. A. V. J. **Design Science Research: Método de Pesquisa para Avanço da Ciência e Tecnologia**. 1 ed. São Paulo: Bookman. 2014.
- ECKLE, K.; SCHMIDT-HIEBER, J. A comparison of deep networks with ReLU activation function and linear spline-type methods. *Neural Networks*, 110, p.232–242. 2019. Disponível em: <https://doi.org/10.1016/j.neunet.2018.11.005>. Acesso em: 10 fev, 2023.
- ELSBACH, K. D.; STIGLIANI, I. Design Thinking and Organizational Culture: A Review and Framework for Future *Research*. *Journal of Management*, v. 44, n.6, p.2274–2306. 2018. Disponível em: <https://doi.org/10.1177/0149206317744252>. Acesso em: 10 fev, 2023.
- FELIN, T.; GAMBARDELLA, A., STERN, S.; ZENGER, T. Lean startup and the business model: Experimentation revisited. *In: Long Range Planning*. v. 53, n. 4, 2020. Disponível em: <https://doi.org/10.1016/j.lrp.2019.06.002>. Acesso em: 10 fev, 2023.
- JHA, N., & MAHMOUD, A. Mining non-functional requirements from App store reviews. *Empirical Software Engineering*, 24(6), 3659–3695. 2019. Disponível em: <https://doi.org/10.1007/s10664-019-09716-7>. Acesso em: 10 fev, 2023.
- MÜNCH, J., FAGERHOLM, F., H.G.A. Sanchez. The RIGHT model for continuous experimentation. *Journal of Systems and Software*, v. 123, p. 292-305, 2017.

NUMPY. **NumPy Documentation**. 2022. Disponível em: <https://numpy.org/doc/stable/>. Acesso em: 03.fev.2023.

OPENCV. **OpenCV Documentation**. 2022. Disponível em: <https://docs.opencv.org/4.x/> Acesso em: 03.fev 2023.

KLEIN, Laura. **UX for lean startups: Faster, smarter user experience research and design**. " O'Reilly Media, Inc.", 2013.

KNIGHT, E., DAYMOND, J., & PAROUTIS, S. Design-Led Strategy: How To Bring Design Thinking Into The Art of Strategic Management. **California Management Review**, v.62, n. 2, p. 30-52, 2020. Disponível em: <https://doi.org/10.1177/0008125619897594>. Acesso em: 10 fev, 2023.

KOWALCZUK, P.; SIEPMANN, C.; ADLER, J. Cognitive, affective, and behavioral consumer responses to augmented reality in e-commerce: A comparative study. **Journal of Business Research**, v. 124, p. 357-373, 2021.

KOZINETS, R. **Netnografia: realizando pesquisa etnográfica online**. São Paulo: Penso. 2014

KNAPP, J; ZERATSKY, J; KOWITZ, B.. **Sprint: How to solve big problems and test new ideas in just five days**. Simon and Schuster, 2016.

KUMAR, A.; NAIR, A. B.; JENA, S.; RANA, D.; PRADHAN, S. K. Facial Expression Recognition Using Python Using CNN Model. **Current Journal of Applied Science and Technology**, v.40, n.20, p. 7-16, 2021.

LERMEN, F.; MOURA, P.; BERTONI, V.; GRACIANO, P.; TORTORELLA, G.. Does maturity level influence the use of Agile UX methods by digital startups? Evaluating design thinking, lean startup, and lean user experience. **Information and Software Technology**, v. 154, p. 101-107, 2023.

LUTHER, L., TIBERIUS, V., BREM, A. User experience (UX) in business, management, and psychology: A bibliometric mapping of the current state of research. **Multimodal Technologies and Interaction**, v.4, n.2. 2020. Disponível em: <https://doi.org/10.3390/mti4020018>. Acesso em: 10 fev, 2023.

MARSH, S. **User research: a practical guide to designing better products and services**. Kogan Page Publishers, 2018.

MUNCH, J., TRIEFLINGER, S., & HEISLER, B. Product Discovery - Building the Right Things: Insights from a Grey Literature Review. **Proceedings - 2020 IEEE International Conference on Engineering, Technology and Innovation, ICE/ITMC 2020**. Disponível em: <https://doi.org/10.1109/ICE/ITMC49519.2020.9198328>. Acesso em: 10 fev, 2023.

PICARD, R.W. **Affective Computing**, Cambridge, MA, USA: MIT Press, 1997. Raul Sidnei Wazlawick. (2021).



PYTHON.ORG. **Python 3.11.1 documentation**. 2022. Disponível em: <https://docs.python.org/>. Acesso em: 3 fev.2023.

RATTAN. **Using LittleVGG for Emotion Detection**. 2022. disponível em: <https://github.com/rajeevratan84/DeepLearningCV>. Acesso em: 25 fev.2023.

RIES, E. **A startup enxuta: como os empreendedores atuais utilizam a inovação contínua para criar empresas extremamente bem-sucedidas**. São Paulo: Lua de Papel, 2012.

ROHRER, C. When to use which user-experience research methods. **Nielsen Norman Group**, v. 12, p. 21, 2022. Disponível em: <https://www.nngroup.com/articles/which-ux-research-methods/> Acesso em: 02 fev 2023.

SHEPHERD, D. A., GRUBER, M. The Lean Startup Framework: Closing the Academic-Practitioner Divide. **Entrepreneurship Theory and Practice**, v.45 ,n.5, p.967–998. 2021.

SHINDE, P. P., SHAH, S. **A Review of Machine Learning and Deep Learning Applications**. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA). 2018. doi:10.1109/iccubea.2018.8697857. Acesso em: 10 fev, 2023.

SILVA, D. S., GHEZZI, A., DE AGUIAR, R. B., CORTIMIGLIA, M. N., CATEN, C. S. Lean Startup, Agile Methodologies and Customer Development for business model innovation: A systematic review and research agenda. **International Journal of Entrepreneurial Behavior & Research**. 2020.

TENSORFLOW. **Visão geral do TensorFlow**. 2022. Disponível em: <https://www.tensorflow.org/overview?hl=pt-br> . Acesso em: 03.02.2023

Teo, Y. S., Shin, S., Jeong, H., Kim, Y., Kim, Y. H., Struchalin, G. I., Kovlakov, E. v., Straupe, S. S., Kulik, S. P., Leuchs, G., & Sánchez-Soto, L. L. (2021). Benchmarking quantum tomography completeness and fidelity with machine learning. **New Journal of Physics**, 23(10). Disponível em: <https://doi.org/10.1088/1367-2630/ac1fcb>. Acesso em: 10 fev, 2023.

TISSOT, Hegler C.; CAMARGO, Luiz C.; POZO, Aurora TR. Treinamento de redes neurais feedforward: comparativo dos algoritmos backpropagation e differential evolution. *In: Brazilian Conference on Intelligent Systems*. 2012.

Trieflinger, S., Münch, J., Heisler, B., & Lang, D. (2021). Essential Approaches to Dual-Track Agile: Results from a Grey Literature Review. **Lecture Notes in Business Information Processing**, 407, 55–69. Disponível em: [https://doi.org/10.1007/978-3-030-67292-8\\_5](https://doi.org/10.1007/978-3-030-67292-8_5). Acesso em: 10 fev, 2023.

VIOLA, Paul; JONES, Michael. Rapid object detection using a boosted cascade of simple features. *In: Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition*. CVPR 2001. Ieee, p. I-0, 2001.

Wang, T. H., Lin, H. C. K., Chen, H. R., Huang, Y. M., Yeh, W. T., & Li, C. T. (2021). Usability of an affective emotional learning tutoring system for mobile devices.

**Sustainability (Switzerland)**, 13(14). Disponível em: <https://doi.org/10.3390/su13147890>. Acesso em: 10 fev, 2023.

WANG, Y., SONG, W., TAO, W., LIOTTA, A., YANG, D., LI, X., GAO, S., SUN, Y., GE, W., ZHANG, W., ZHANG, W. **A Systematic Review on Affective Computing: Emotion Models, Databases, and Recent Advances**. 2022. Disponível em: <http://arxiv.org/abs/2203.06935>, Acesso em: 10 fev, 2023.

WAZLAWICK, R.S. **Metodologia de Pesquisa para Ciência da Computação**. 3rd ed., vol. 1. São Paulo: LTC. 2021.

YINGGE, H., ALI, I., LEE, K. Y. Deep neural networks on chip - A survey. Proceedings - 2020 IEEE **International Conference on Big Data and Smart Computing, BigComp**. p. 589–592. 2020. Disponível em: <https://doi.org/10.1109/BigComp48618.2020.00016>. Acesso em: 10 fev, 2023.