

Universidade de Brasília – UnB
Faculdade de Ciências e Tecnologias em Engenharia – FCTE
Engenharia de Software

Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo

Autor: Cíbele Freitas Goudinho
Orientador: Profa. Dra. Milene Serrano

Brasília, DF
2025



Cibele Freitas Goudinho

Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Universidade de Brasília – UnB

Faculdade de Ciências e Tecnologias em Engenharia – FCTE

Orientador: Profa. Dra. Milene Serrano

Coorientador: Prof. Dr. Maurício Serrano

Brasília, DF

2025

Cibele Freitas Goudinho

Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo/ Cibele Freitas Goudinho. – Brasília, DF, 2025-

181 p. : il. (algumas color.) ; 30 cm.

Orientador: Profa. Dra. Milene Serrano

Trabalho de Conclusão de Curso – Universidade de Brasília – UnB

Faculdade de Ciências e Tecnologias em Engenharia – FCTE , 2025.

1. Sistemas de Recomendação. 2. Aprendizado Profundo. I. Profa. Dra. Milene Serrano. II. Universidade de Brasília. III. Faculdade UnB Gama. IV. Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo

CDU 02:141:005.6

Cibele Freitas Goudinho

Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo

Monografia submetida ao curso de graduação em Engenharia de Software da Universidade de Brasília, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Software.

Trabalho aprovado. Brasília, DF, 21 de Fevereiro de 2025:

Profa. Dra. Milene Serrano
Orientadora

Prof. Dr. Maurício Serrano
Coorientador

Profa. M.a. Cristiane Soares Ramos
Examinadora

M.e. José Pedro de Santana Neto
Examinador

Brasília, DF
2025

Agradecimentos

Agradeço, primeiramente, aos meus pais, que sempre estiveram ao meu lado, me apoiando em cada escolha e me auxiliando em todos os momentos. Seu amor e incentivo foram fundamentais para que eu chegasse até aqui.

À minha orientadora, Milene Serrano, e ao meu coorientador, Mauricio Serrano, pela orientação valiosa ao longo deste trabalho e pelo conhecimento compartilhado nas disciplinas que tive o privilégio de cursar com eles.

Aos meus amigos, que me incentivaram e apoiaram em toda essa jornada, tornando os desafios mais leves e as conquistas ainda mais significativas.

Aos meus familiares e professores, que, de diversas formas, contribuíram para o meu crescimento acadêmico e pessoal.

Meu sincero agradecimento a todos que, direta ou indiretamente, fizeram parte dessa caminhada.

“A IA não é sobre prever o futuro; é sobre moldar o futuro.” (Andrew Ng)

Resumo

Sistemas de Recomendação são algoritmos de software projetados para analisar o comportamento e as preferências passadas dos usuários, com o objetivo de sugerir itens que possam ser de seu interesse. Esses sistemas são amplamente utilizados em diversas plataformas, como serviços de *streaming*, lojas *online*, redes sociais e outros aplicativos, para melhorar a experiência e a satisfação do usuário ao oferecer recomendações relevantes e personalizadas. Entretanto, ainda há muito a ser explorado em termos de pesquisa no contexto de Sistemas de Recomendação, principalmente, no intuito de gerar maior satisfação aos seus usuários. Este trabalho visou criar um Sistema de Recomendação centrado em Aprendizado Profundo, e orientando-se pelo critério qualitativo de satisfação do usuário. Para isso, foi construído um modelo treinado por Filtros Colaborativos, e em seguida foi utilizado esse modelo para treinar um modelo de Rede Neural Convolucional, a fim de aprimorar as recomendações. Ao final, utilizou-se métricas, tais como: Precisão; Revocação; Erro Médio Absoluto e Erro Quadrático Médio da Raiz. A intenção foi avaliar o modelo resultante, aplicando esse modelo a uma API para que os usuários conseguissem usar e avaliar o Sistema de Recomendação. Os resultados indicaram que a maioria dos usuários que foram avaliados gostaram do modelo desenvolvido, se interessando pelas recomendações. Além disso, o modelo converge rapidamente, necessitando de apenas duas épocas para alcançar boas recomendações, o que demonstra eficiência computacional. No entanto, algumas limitações foram observadas, como a necessidade de um ambiente de execução com alta RAM, o problema de "partida a frio" nas primeiras recomendações, dificuldades na integração da API com o sistema e desafios relacionados à escalabilidade. Assim esse estudo apresenta um avanço na aplicação de Redes Neurais Convolucionais em Sistemas de Recomendação baseados em Filtros Colaborativos, oferecendo uma abordagem diferenciada e eficiente. Entretanto, melhorias ainda podem ser implementadas para tornar o modelo mais escalável e otimizado para uso real.

Palavras-chave: Sistemas de Recomendação, Aprendizado Profundo, Filtros Colaborativos, Rede Neural Convolucional.

Abstract

Recommendation Systems are software algorithms designed to analyze user behavior and past preferences to suggest items that may be of interest. These systems are widely used across various platforms, such as streaming services, online stores, social networks, and other applications, to enhance user experience and satisfaction by providing relevant and personalized recommendations. However, there is still much to be explored in terms of research in the context of Recommendation Systems, especially to increase user satisfaction. This study aimed to develop a Recommendation System focused on Deep Learning, guided by the qualitative criterion of user satisfaction. To achieve this, a model trained with Collaborative Filtering was built, and this model was then used to train a Convolutional Neural Network (CNN) to improve recommendations. In the end, metrics such as Precision, Recall, Mean Absolute Error, and Root Mean Squared Error were used to evaluate the resulting model, which was integrated into an API to allow users to interact with and assess the Recommendation System. The results indicated that most evaluated users appreciated the developed model and found the recommendations engaging. Additionally, the model converges quickly, requiring only two epochs to achieve high-quality recommendations, demonstrating computational efficiency. However, some limitations were observed, such as the need for a high-RAM execution environment, the "cold start" problem in initial recommendations, difficulties in integrating the API with the system, and scalability challenges. Thus, this study represents an advancement in the application of Convolutional Neural Networks in Recommendation Systems based on Collaborative Filtering, offering a differentiated and efficient approach. Nevertheless, further improvements can be implemented to make the model more scalable and optimized for real-world use.

Key-words: Recommender Systems, Deep Learning, Artificial Intelligence, Collaborative Filters, Convolutional Neural Network.

Lista de Ilustrações

Figura 1 – Diagrama de Ishikawa	27
Figura 2 – Ciclo dos Sistemas Recomendação	33
Figura 3 – Filtro Colaborativo	34
Figura 4 – Filtro de Conteúdo	35
Figura 5 – Combinação Individual	42
Figura 6 – Agregar Características do Filtro de Conteúdo no Filtro Colaborativo	42
Figura 7 – Agregar Características do Filtro Colaborativo no Filtro de Conteúdo	43
Figura 8 – Ambos os Filtros em um Único Modelo	43
Figura 9 – Estrutura de um <i>Autoencoder</i>	44
Figura 10 – Arquitetura RNC	45
Figura 11 – Aplicação de Filtros - RNC	46
Figura 12 – <i>Backlog</i> - Primeira Etapa do Trabalho	60
Figura 13 – <i>Backlog</i> - Segunda Etapa do Trabalho	61
Figura 14 – Fluxo de Atividades da Primeira Etapa do Trabalho	63
Figura 15 – Subprocesso - Desenvolvimento de Prova de Conceito	64
Figura 16 – Subprocesso - Levantamento do Referencial Teórico	64
Figura 17 – Fluxo de Atividades da Segunda Etapa do Trabalho	66
Figura 18 – Subprocesso - Modelo de Sistema de Recomendação	67
Figura 19 – Subprocesso - Desenvolvimento da API	67
Figura 20 – Subprocesso - Coleta de Dados do Usuário	68
Figura 21 – Subprocesso - Análise da Satisfação do Usuário	68
Figura 22 – Base de Dados	75
Figura 23 – Consumo de Memória - Filtro Colaborativo	76
Figura 24 – Falha na Conexão - Filtro de Conteúdo	76
Figura 25 – Consumo de Memória - Filtro de Conteúdo	76
Figura 26 – Preparar Dados - Filtro Colaborativo	77
Figura 27 – Treinar Modelo - Filtro Colaborativo	77
Figura 28 – Avaliar Modelo - Filtro Colaborativo	78
Figura 29 – Avaliar Modelo - Filtro Colaborativo	78
Figura 30 – Prever Avaliação - Filtro Colaborativo	79
Figura 31 – Obter Recomendação - Filtro Colaborativo	79
Figura 32 – 5% da Base - Filtro Colaborativo	80
Figura 33 – 40% da Base - Filtro Colaborativo	80
Figura 34 – 100% da Base - Filtro Colaborativo	80
Figura 35 – Preparar Dados - Rede Neural Convolutacional	82
Figura 36 – Treinar Modelo - Rede Neural Convolutacional	83

Figura 37 – Avaliar Modelo - Rede Neural Convolutacional	83
Figura 38 – Prever Avaliação - Rede Neural Convolutacional	83
Figura 39 – Obter Recomendação - Rede Neural Convolutacional	84
Figura 40 – Resultados - Rede Neural Convolutacional	84
Figura 41 – Preparar Dados - Rede Neural Recorrente	84
Figura 42 – Treinar Modelo - Rede Neural Recorrente	85
Figura 43 – Avaliar Modelo - Rede Neural Recorrente	86
Figura 44 – Prever Avaliação - Rede Neural Recorrente	86
Figura 45 – Obter Recomendação - Rede Neural Recorrente	86
Figura 46 – Resultados - Rede Neural Recorrente	87
Figura 47 – Preparar Dados - <i>Autoencoder</i>	88
Figura 48 – Treinar Modelo - <i>Autoencoder</i>	88
Figura 49 – Avaliar Modelo - <i>Autoencoder</i>	89
Figura 50 – Prever Avaliação - <i>Autoencoder</i>	89
Figura 51 – Obter Recomendação - <i>Autoencoder</i>	90
Figura 52 – Resultados - <i>Autoencoder</i>	90
Figura 53 – Calcular Previsão - RNC	91
Figura 54 – Recomendação Novo Usuário - RNC	92
Figura 55 – Resultados - RNC Modelo Final	92
Figura 56 – Resultados - Consumos de Memória e Disco RNC	92
Figura 57 – Esboço da Arquitetura da Solução Proposta (Macro Nível de Abstração)	94
Figura 58 – Esboço da Arquitetura do <i>Backend</i> (Macro Nível de Abstração)	94
Figura 59 – Esboço da Arquitetura do <i>Frontend</i> (Macro Nível de Abstração)	95
Figura 60 – Tela de <i>Login</i>	95
Figura 61 – Tela de Obter Gêneros	96
Figura 62 – Tela de Recomendação	96
Figura 63 – Fluxo do Usuário	97
Figura 64 – Resposta da Requisição de Primeira Recomendação	97
Figura 65 – Resposta da Requisição de Segunda Recomendação	98
Figura 66 – Persona 1	100
Figura 67 – Persona 2	100
Figura 68 – Antipersona	101
Figura 69 – Termo de Consentimento Livre e Esclarecido	135
Figura 70 – Coleta de Dados - Exemplo - Página 1	137
Figura 71 – Coleta de Dados - Exemplo - Página 2	138
Figura 72 – Coleta de Dados - Exemplo - Página 3	139
Figura 73 – Coleta de Dados - Exemplo - Página 4	140
Figura 74 – Coleta de Dados - Exemplo - Página 5	140
Figura 75 – Coleta de Dados - Exemplo - Página 6	141

Figura 76 – Coleta de Dados - Exemplo - Página 7	142
Figura 77 – Avaliação da Primeira Recomendação - Exemplo - Página 1	143
Figura 78 – Avaliação da Primeira Recomendação - Exemplo - Página 2	144
Figura 79 – Avaliação da Primeira Recomendação - Exemplo - Página 3	144
Figura 80 – Avaliação da Primeira Recomendação - Exemplo - Página 4	144
Figura 81 – Avaliação da Primeira Recomendação - Exemplo - Página 5	145
Figura 82 – Avaliação da Primeira Recomendação - Exemplo - Página 6	146
Figura 83 – Avaliação da Primeira Recomendação - Exemplo - Página 7	147
Figura 84 – Avaliação da Primeira Recomendação - Exemplo - Página 8	147
Figura 85 – Avaliação da Primeira Recomendação - Exemplo - Página 9	148
Figura 86 – Avaliação da Primeira Recomendação - Exemplo - Página 10	149
Figura 87 – Avaliação da Primeira Recomendação - Exemplo - Página 11	150
Figura 88 – Avaliação da Primeira Recomendação - Exemplo - Página 12	151
Figura 89 – Avaliação da Primeira Recomendação - Exemplo - Página 13	151
Figura 90 – Avaliação da Primeira Recomendação - Exemplo - Página 14	152
Figura 91 – Avaliação da Primeira Recomendação - Exemplo - Página 15	153
Figura 92 – Avaliação da Primeira Recomendação - Exemplo - Página 16	154
Figura 93 – Avaliação da Primeira Recomendação - Exemplo - Página 17	155
Figura 94 – Avaliação da Primeira Recomendação - Exemplo - Página 18	156
Figura 95 – Avaliação da Primeira Recomendação - Exemplo - Página 19	157
Figura 96 – Avaliação da Primeira Recomendação - Exemplo - Página 20	158
Figura 97 – Avaliação da Primeira Recomendação - Exemplo - Página 21	158
Figura 98 – Avaliação da Primeira Recomendação - Exemplo - Página 22	159
Figura 99 – Avaliação da Primeira Recomendação - Exemplo - Página 23	160
Figura 100 – Avaliação da Primeira Recomendação - Exemplo - Página 24	161
Figura 101 – Avaliação da Primeira Recomendação - Exemplo - Página 25	161
Figura 102 – Avaliação da Primeira Recomendação - Exemplo - Página 26	162
Figura 103 – Avaliação da Primeira Recomendação - Exemplo - Página 27	163
Figura 104 – Avaliação da Primeira Recomendação - Exemplo - Página 28	164
Figura 105 – Avaliação da Primeira Recomendação - Exemplo - Página 29	165
Figura 106 – Avaliação da Primeira Recomendação - Exemplo - Página 30	165
Figura 107 – Avaliação da Primeira Recomendação - Exemplo - Página 31	166
Figura 108 – Avaliação da Segunda Recomendação - Exemplo - Página 1	167
Figura 109 – Avaliação da Segunda Recomendação - Exemplo - Página 2	168
Figura 110 – Avaliação da Segunda Recomendação - Exemplo - Página 3	168
Figura 111 – Avaliação da Segunda Recomendação - Exemplo - Página 4	168
Figura 112 – Avaliação da Segunda Recomendação - Exemplo - Página 5	169
Figura 113 – Avaliação da Segunda Recomendação - Exemplo - Página 6	169
Figura 114 – Avaliação da Segunda Recomendação - Exemplo - Página 7	170

Figura 115 – Avaliação da Segunda Recomendação - Exemplo - Página 8	171
Figura 116 – Avaliação da Segunda Recomendação - Exemplo - Página 9	172
Figura 117 – Avaliação da Segunda Recomendação - Exemplo - Página 10	173
Figura 118 – Avaliação da Segunda Recomendação - Exemplo - Página 11	174
Figura 119 – Avaliação da Segunda Recomendação - Exemplo - Página 12	175
Figura 120 – Avaliação da Segunda Recomendação - Exemplo - Página 13	176
Figura 121 – Avaliação da Segunda Recomendação - Exemplo - Página 14	177
Figura 122 – Avaliação da Segunda Recomendação - Exemplo - Página 15	177
Figura 123 – Avaliação da Segunda Recomendação - Exemplo - Página 16	178
Figura 124 – Avaliação da Segunda Recomendação - Exemplo - Página 17	179
Figura 125 – Avaliação da Segunda Recomendação - Exemplo - Página 18	180
Figura 126 – Avaliação da Segunda Recomendação - Exemplo - Página 19	181
Figura 127 – Avaliação da Segunda Recomendação - Exemplo - Página 20	181

Lista de Quadros

Quadro 1	-	Classificação por Peso	38
Quadro 2	-	Classificação por Mistura	39
Quadro 3	-	Classificação por Troca	39
Quadro 4	-	Classificação por Combinação de <i>Features</i>	40
Quadro 5	-	Classificação por Cascata	40
Quadro 6	-	Classificação por Aumento de <i>Features</i>	41
Quadro 7	-	Classificação por <i>Meta-level</i>	42
Quadro 8	-	Principais Ferramentas Utilizadas no Trabalho	54
Quadro 9	-	Artigos Retornados na Base do Google Acadêmico	58
Quadro 10	-	Principais Artigos que Orientam a Pesquisa	58
Quadro 11	-	Cronograma de Atividades/Subprocessos da Primeira Etapa do TCC	70
Quadro 12	-	Cronograma de Atividades/Subprocessos da Segunda Etapa do TCC	70
Quadro 13	-	Andamento de Atividades e Subprocessos da Primeira Etapa do TCC	120
Quadro 14	-	Andamento de Atividades e Subprocessos da Segunda Etapa do TCC	120

Lista de Abreviaturas e Siglas

API	<i>Application Programming Interface</i> (Interface de Programação de Aplicações)
BPMN	<i>Business Process Modeling Notation</i> (Modelo e Notação de Processos de Negócio)
COVID	<i>Corona Virus Disease</i> (Doença do Corona Vírus)
DL	<i>Deep Learning</i> (Aprendizado Profundo)
EMA/MAE	Erro Médio Absoluto (<i>Mean Absolute Error</i>)
EQMR/RMSE	Erro Quadrático Médio da Raiz (<i>Root Mean Squared Error</i>)
FBCU	Filtro Baseado em Conhecimento e Utilidade
FBAP	Filtro Baseado em Aprendizado Profundo
FC	Filtro Colaborativo
FD	Filtro Demográfico
FN	Falsos Negativos
FPC	Filtro Por Conteúdo
IA	Inteligência Artificial
IMDb	<i>Internet Movie Database</i> (Base de Dados de Filmes da Internet)
ML	<i>Machine Learning</i> (Aprendizado de Máquina)
MRB	Máquina Restrita de Boltzmann
POC	Prova de Conceito
RAM	<i>Random Access Memory</i> (Memória de Acesso Aleatório)
RNC/CNN	Redes Neurais Convolucionais (<i>Convolutional Neural Networks</i>)
RNR	Redes Neurais Recorrentes
SR	Sistema de Recomendação
SVD	<i>Singular Value Decomposition</i> (Valor Singular de Decomposição)

TCC	Trabalho de Conclusão de Curso
TLE	<i>Time Limit Exceeded</i> (Tempo Limite Excedido)
TPU	<i>Tensor Processing Units</i> (Unidade de Processamento de Tensor)
VP	Verdadeiros Positivos

Sumário

1	INTRODUÇÃO	25
1.1	Contextualização	25
1.2	Justificativa	28
1.3	Questões de Pesquisa e Desenvolvimento	29
1.4	Objetivos	30
1.4.1	Objetivo Geral	30
1.4.2	Objetivos Específicos	30
1.5	Organização da Monografia	31
2	REFERENCIAL TEÓRICO	33
2.1	Sistemas de Recomendação	33
2.2	Inteligência Artificial	34
2.2.1	Filtros Híbridos	36
2.2.1.1	Classificação de Filtros Híbridos	36
2.2.1.2	Possíveis Implementações	41
2.2.2	Filtros Baseados em Aprendizado Profundo	43
2.2.2.1	Redes Neurais Convolucionais	43
2.2.2.2	Redes Neurais Recorrentes	44
2.2.2.3	Máquina de Boltzmann Restrita	44
2.2.2.4	<i>Autoencoder</i>	44
2.2.3	Modelo Utilizado	45
2.3	Satisfação do Usuário	47
2.4	Resumo do Capítulo	48
3	SUPORTE TECNOLÓGICO	51
3.1	Apoio à Pesquisa	51
3.1.1	Kaggle	51
3.1.2	HuggingFace	52
3.2	Apoio ao Desenvolvimento	52
3.2.1	Python	52
3.2.2	Colab/TPU	52
3.2.3	Google <i>Forms</i> e Google <i>Sheets</i>	53
3.3	Apoio à Elaboração da Monografia	53
3.3.1	LaTeX	53
3.3.2	Git/GitHub	53
3.3.3	Canva	53

3.3.4	Lucidchart	53
3.4	Resumo do Capítulo	54
4	METODOLOGIA	55
4.1	Classificação de Pesquisa	55
4.1.1	Abordagem	55
4.1.2	Natureza	56
4.1.3	Objetivos	56
4.1.4	Procedimentos	56
4.2	Método de Pesquisa Bibliográfica	56
4.3	Método de Desenvolvimento	57
4.4	Método de Coleta e Análise dos Resultados	59
4.5	Fluxo das Atividades/Subprocessos	63
4.5.1	Primeira Etapa do Trabalho	63
4.5.2	Segunda Etapa do Trabalho	66
4.6	Cronogramas	69
4.7	Resumo do Capítulo	69
5	SISTEMA DE RECOMENDAÇÃO COM APRENDIZADO PROFUNDO	73
5.1	Contextualização	73
5.2	Solução Inicial para o Sistema de Recomendação	74
5.2.1	Modelo com Filtros Colaborativos	75
5.2.2	Modelo com Redes Neurais Convolucionais	81
5.2.3	Modelo com Redes Neurais Recorrentes	81
5.2.4	Modelo com <i>Autoencoder</i>	87
5.3	Solução Evoluída para o Sistema de Recomendação	90
5.3.1	Filtros Colaborativos e Aprendizado Profundo	91
5.3.2	API	93
5.4	Resumo do Capítulo	98
6	ANÁLISE DE RESULTADOS	99
6.1	Perfil dos Participantes	99
6.2	Formulários	101
6.3	Divulgação de Resultados	103
6.4	Resumo do Capítulo	116
7	CONCLUSÃO	119
7.1	Atividades Realizadas	119
7.2	Status dos Objetivos Específicos	121
7.3	Contribuições e Limitações	123
7.4	Impressões da Autora em Relação ao Trabalho	124

7.5	Trabalhos Futuros	125
7.6	Resumo do Capítulo	125

REFERÊNCIAS	127
-----------------------	-----

APÊNDICES 133

APÊNDICE A – TERMO DE CONSENTIMENTO

LIVRE E ESCLARECIDO	135
-------------------------------	-----

APÊNDICE B – PRIMEIRO FORMULÁRIO - COLETA DE

DADOS DO USUÁRIO	139
----------------------------	-----

APÊNDICE C – SEGUNDO FORMULÁRIO - AVALIAÇÃO

DA PRIMEIRA RECOMENDAÇÃO	145
------------------------------------	-----

APÊNDICE D – TERCEIRO FORMULÁRIO - AVALIAÇÃO

DA SEGUNDA RECOMENDAÇÃO	169
-----------------------------------	-----

1 Introdução

Este Capítulo introduz o contexto no qual este trabalho está inserido. Inicialmente, aborda-se a [Contextualização](#) (seção 1), que traz o foco de interesse desse trabalho (Inteligência Artificial e Sistemas de Recomendação). Na sequência, consta a [Justificativa](#) (seção 2) para sua realização, além das [Questões de Pesquisa e Desenvolvimento](#) (seção 3) a serem respondidas, e os [Objetivos](#) (seção 4) que se pretende alcançar. Por fim, é abordada a [Organização da Monografia](#) (seção 5).

1.1 Contextualização

Sistemas de Recomendação (SR) são algoritmos de software projetados para analisar o comportamento passado do usuário e suas preferências, com o intuito de sugerir itens que possam ser de seu interesse ([CHEW; HAW; SUBRAMANIAM, 2020](#)). No contexto da sociedade atual, com grande conjunto de dados disponíveis, tem se tornado cada vez mais desafiador para os usuários encontrarem exatamente o que procuram em meio às vastas opções. Dessa forma, os Sistemas de Recomendação surgem como meios para facilitar a descoberta de conteúdo relevante em ambientes digitais. Grandes empresas utilizam Sistemas de Recomendação como estratégia para conquistar e auxiliar seus clientes, como o YouTube e a Netflix ([ZHANG et al., 2019](#)).

Os benefícios dos Sistemas de Recomendação na sociedade atual são significativos. Eles não apenas facilitam a descoberta de novos produtos e conteúdos, mas também aumentam o engajamento do usuário e impulsionam as vendas e a receita para empresas e plataformas digitais. Além disso, ao personalizar a experiência do usuário, os Sistemas de Recomendação podem melhorar a satisfação do cliente e a fidelidade à marca ([GUNAWARDANA; SHANI; YOGEV, 2022](#)).

No entanto, os Sistemas de Recomendação possuem desafios. De acordo com a pesquisa [Khusro, Ali e Ullah \(2016\)](#), os principais desafios que ocorrem no contexto dos Sistemas de Recomendação são:

- *Cold Start*: ocorre quando há nenhum ou poucos dados para iniciar as sugestões, acontecendo principalmente com novos usuários. Esse desafio também é conhecido como "partida a frio";
- Esparsidade dos dados: ocorre quando a matriz de dados dos usuários possui muitos espaços vazios. Percebe-se essa situação, principalmente, devido à grande quantidade

de itens, o que dificulta o preenchimento pleno desses itens; ou ainda pela falta de avaliações dos usuários para cada item;

- Escalabilidade: ocorre por haver a necessidade de uma grande rede de usuários, gerando outros problemas, por exemplo, baixo desempenho;
- Diversidade: ocorre pela dificuldade em conciliar itens diferentes (diferenciação) e itens similares (sobreposição). A diferenciação permite aos usuários receberem itens mais variados, algo que pode ser do seu interesse. Já a sobreposição confere apenas itens similares, não recomendando itens que possuem ligações mais distantes, e
- Efeito de habituação: ocorre quando o grande número de informações gera um efeito de costume no usuário, ou seja, certa rotina em suas preferências. Isso pode fazer com que várias recomendações sejam ignoradas, recomendando-se sempre a mesma coisa.

Problemas os quais estão representados pelo diagrama de Ishikawa na Figura 1, este sendo utilizado para identificar e categorizar as causas de um problema central. No caso desse diagrama, o problema principal é "Desafios nos Sistemas de Recomendação", e as causas estão representadas pelas espinhas que se ramificam a partir dele. Ao analisar o diagrama, percebe-se que a partida a frio pode estar relacionada à esparcidade, pois a falta de avaliações de itens pelos usuários dificulta a recomendação precisa. Bem como a esparcidade, por sua vez, pode estar relacionada à escalabilidade, pois um aumento no número de usuários e itens pode agravar esse problema. Já a diversidade e o efeito de habituação podem estar relacionados, visto que a falta de variedade nas sugestões pode reforçar o efeito da habituação.

Outro avanço que está ocorrendo é o da Inteligência Artificial (IA), que tem permeado diversos setores da sociedade (LAKEHOUSE, 2023). IA refere-se à capacidade de sistemas computacionais executarem tarefas que normalmente exigiriam inteligência humana, tais como: aprendizado, raciocínio, reconhecimento de padrões e tomada de decisões.

Uma área específica da IA que tem ganhado destaque é o Aprendizado Profundo (em inglês, *Deep Learning*), um ramo do aprendizado de máquina que se baseia em redes neurais artificiais para realizar tarefas complexas de forma automatizada. Essas tecnologias trazem vários benefícios, desde a automação de processos até a análise de grandes volumes de dados, otimizando operações e impulsionando a tomada de decisões em diversos setores (saúde, transporte, entretenimento, dentre outros) (SULEIMENOV et al., 2020). No intuito de mitigar os desafios inerentes aos Sistemas de Recomendação, pode-se fazer uso de Aprendizado Profundo.

Figura 1 – Diagrama de Ishikawa



Fonte: Autora

Usar Aprendizado Profundo pode ajudar a mitigar o desafio *Cold Start*, ou "partida a frio". Isso ocorre, pois Aprendizado Profundo permite aprender representações complexas a partir de dados brutos. Ao utilizar técnicas como redes neurais, os Sistemas de Recomendação podem extrair padrões e relações significativas dos dados, mesmo quando há poucos dados disponíveis para novos usuários. Isso permite que o sistema forneça recomendações mais precisas e relevantes, mesmo para usuários com histórico limitado (ANALYTICS, 2021).

No desafio de esparcidade dos dados, Aprendizado Profundo permite aprender com base em representações densas de informações, mesmo que a matriz que comporta tais informações não seja plena, com todos os itens preenchidos, restando espaços em branco (HEIDARI; MORADI; KOOCHARI, 2022). Nesse cenário, técnicas como *Autoencoders* podem ser usadas para preencher lacunas nos dados, gerando representações completas e semanticamente coerentes. Explicações complementares de como funcionam tais técnicas serão conferidas no Referencial Teórico (Cap. 2).

Por serem altamente paralelizáveis, técnicas de Aprendizado Profundo podem ser implementadas em sistemas distribuídos. Essa estratégia permite lidar com grande quan-

tidade de dados e usuários. Ao usar técnicas de treinamento distribuído e otimização eficiente, os Sistemas de Recomendação baseados em Aprendizado Profundo podem escalar para grandes conjuntos de dados e redes de usuários sem comprometer o desempenho (YAN et al., 2015).

Aprendizado Profundo ajuda a melhorar a diversidade das recomendações ao aprender representações mais abrangentes dos itens e usuários. Além disso, permite que o Sistema de Recomendação perceba similaridades com base nos itens e usuários. Como resultado, tende-se a recomendações mais equilibradas e acertivas. Nos estudos realizados pela autora, cabe menção à técnica de introdução de fatores de diversidade no processo de geração de recomendações, conforme abordado em Kingma et al. (2016). Essa técnica busca garantir que os usuários recebam uma variedade de itens relevantes. Constam outros detalhes sobre o funcionamento dessa técnica no Referencial Teórico (Cap. 2).

Sistemas de Recomendação orientados a Aprendizado Profundo são capazes de aprender continuamente, centrados nos *feedbacks* dos usuários. Adicionalmente, são sistemas capazes de adaptar as recomendações ao longo do tempo. Essas particularidades de aprendizado contínuo e adaptabilidade possibilitam que as recomendações permaneçam atualizadas e relevantes aos interesses dos usuários, mesmo após longos períodos de uso (LOMONACO, 2019). Portanto, isso mitiga o efeito de habituação comentado anteriormente nesse capítulo.

Diante do exposto, usar recursos da IA em Sistemas de Recomendação representa uma oportunidade para melhorar as recomendações conferidas aos usuários. Ao combinar algoritmos avançados de IA com dados detalhados sobre o comportamento do usuário, é possível gerar recomendações mais precisas e relevantes, que atendam às necessidades e preferências individuais do usuário de forma mais eficaz (KARATZOGLOU; HIDASI, 2017). Essa abordagem híbrida permite que os Sistemas de Recomendação aprendam e se adaptem continuamente, melhorando sua capacidade de prever e antecipar as preferências do usuário ao longo do tempo.

1.2 Justificativa

Os Sistemas de Recomendação com IA trazem diversos benefícios às companhias, tais como: aumento do número de vendas; maior diversidade de itens vendidos; aumento da satisfação dos usuários, maior fidelidade desses usuários, e melhor atendimento quanto às necessidades dos usuários (GUNAWARDANA; SHANI; YOGEV, 2022). Usando a Netflix como exemplo, a combinação de recomendação e personalização salva por ano cerca de \$ 1 B (KOMPAN et al., 2022). Ainda de acordo com Kompan et al. (2022), incorporar um Sistema de Recomendação pode resultar em 30-35% de aumento no número de compras em *e-commerce*. Outros exemplos de grandes companhias que utilizam Sistemas

de Recomendação centrados na IA são: Amazon, IMDb e YouTube ([KONSTAN; RIEDL, 2012](#)).

Apesar dos benefícios proporcionados pelos Sistemas de Recomendação, também é importante reconhecer suas limitações e desafios. De acordo com [Mishra et al. \(2021\)](#), algumas dessas limitações podem ser: a "partida a frio" (*cold start*), a qual ocorre quando usuários não fornecem *feedbacks* ou avaliações aos sistemas de recomendação, e a escalabilidade, que ocorre à medida que a rede de usuários cresce ou a base de dados necessita de atualização.

Em meio a esse cenário, torna-se relevante considerar o impacto da crescente utilização da IA no desenvolvimento e na evolução de Sistemas de Recomendação. Estudos indicam que o número de empresas que utilizam serviços baseados em IA cresceu significativamente nos últimos anos, o que evidencia a importância e o potencial dessas tecnologias ([LAKEHOUSE, 2023](#)). É nesse contexto que se destaca a relevância da aplicação de técnicas avançadas de IA, como redes neurais, para aprimorar a capacidade dos Sistemas de Recomendação em entender e antecipar as preferências dos usuários ([ZHANG et al., 2019](#)).

Considerando os benefícios apontados anteriormente no que compreende o uso de IA em Sistemas de Recomendação, o presente trabalho visou a realização de um estudo, procurando investigar estratégias e técnicas de IA, mais especificamente Aprendizado Profundo para o desenvolvimento de Sistemas de Recomendação mais eficientes e precisos. Sistemas esses capazes de proporcionar uma experiência personalizada e satisfatória aos usuários em diferentes contextos e aplicações.

1.3 Questões de Pesquisa e Desenvolvimento

Ao final desse trabalho, pretende-se responder às seguintes questões:

Questão de Desenvolvimento: Como um Sistema de Recomendação centrado em Aprendizado Profundo pode ser desenvolvido?

Questão de Pesquisa: Esse Sistema de Recomendação centrado em Aprendizado Profundo, de fato, proporciona maior satisfação aos seus usuários?

Com a exposição das questões, cabem dois esclarecimentos adicionais.

O primeiro compreende deixar claro sobre o escopo do Sistema de Recomendação pretendido. É um sistema de recomendação que faz uso de algumas técnicas de Aprendizado Profundo, sendo essas as mais recomendadas pela literatura em estudo. Tais técnicas foram escolhidas com base nos desafios acordados anteriormente, orientando-se pelos autores [Khusro, Ali e Ullah \(2016\)](#). É um sistema de escopo menor, mas que foi documentado para permitir que outros interessados consigam usá-lo de base.

Segundo [Esperidiao \(2005\)](#), satisfação de usuário é um critério qualitativo. Portanto, trata-se de um requisito não funcional, ou seja, algo subjetivo. No intuito de conferir maior clareza sobre como "mensurar" a satisfação do usuário, pretende-se fazer uso de questionários, junto ao público alvo de cada domínio de interesse utilizado ao longo desse trabalho, para coletar *feedbacks* sobre as impressões desses usuários quanto às recomendações recebidas, sem e com o auxílio de um Sistema de Recomendação centrado em Aprendizado Profundo. A decisão de usar um questionário baseou-se na literatura.

Segundo a autora [Esperidiao \(2005\)](#), dentre os métodos quantitativos, destacam-se *surveys*, incluindo o uso de questionários fechados. Nesses questionários, as questões estão associadas a uma escala de valores que procuram mensurar a satisfação dos usuários. Na verdade, costumam mensurar aspectos associados à satisfação, no caso: expectativas e percepções.

Ainda de acordo com [Esperidiao \(2005\)](#), há possibilidade de usar métodos qualitativos, mas deve-se ter em mente que os mesmos costumam ser criticados pelos especialistas, por incorrerem em pesquisas com viés, dentre outros problemas.

O presente trabalho faz uso de uma abordagem híbrida, quantitativa e qualitativa, conforme alguns autores defendem ([MINAYO; SANCHES, 1993](#)). Nesse contexto, cabe colocar que, nessa abordagem híbrida, a parte quantitativa e a parte qualitativa não se encontram em oposição, mas sim em continuidade e complementaridade. O questionário orientou-se por uma escala numérica, ou seja, quantitativa, mas que seguiu uma análise interpretativa sobre esses números, sendo assim conferido, adicionalmente, um viés qualitativo. Colocações complementares sobre o método de análise de resultados ocorrerão em [Metodologia](#) (Cap. 4).

1.4 Objetivos

Os objetivos do trabalho foram organizados em [Objetivo Geral](#) (seção 4.1), que confere uma visão mais abrangente do que se pretendeu atingir com a realização do trabalho, e [Objetivos Específicos](#) (seção 4.2), compreendendo metas de menor escopo a serem cumpridas para que fosse viável atingir o objetivo geral.

1.4.1 Objetivo Geral

Esse trabalho teve como objetivo geral um estudo sobre Sistemas de Recomendação centrados em Aprendizado Profundo, revelando como desenvolvê-los e analisando se os mesmos melhoram a satisfação dos usuários.

1.4.2 Objetivos Específicos

Na perspectiva de alcançar o objetivo geral, foram definidos objetivos específicos, conforme apresentado a seguir:

1. Levantamento sobre Sistemas de Recomendação;
2. Levantamento, no contexto da IA, de algoritmos de Aprendizado Profundo que mais atendam à proposta desse trabalho;
3. Identificação de quais informações são necessárias para especificar uma base de dados para o projeto;
4. Treinamento da base de dados orientando-se pelos algoritmos de Aprendizado Profundo levantados;
5. Desenvolvimento de um Sistema de Recomendação centrado em IA, mais especificamente usando algoritmos de Aprendizado Profundo, e
6. Análise da satisfação do usuário sem e com o uso do Sistema de Recomendação desenvolvido.

1.5 Organização da Monografia

Esta monografia está organizada em capítulos, conforme consta a seguir:

Capítulo 2 - Referencial Teórico: aborda fundamentos, conceitos, princípios e práticas relevantes para o tema, em específico, Sistemas de Recomendação centrados em Inteligência Artificial;

Capítulo 3 - Suporte Tecnológico: apresenta as principais tecnologias utilizadas para o desenvolvimento do trabalho;

Capítulo 4 - Metodologia: esclarece sobre os métodos que guiaram a elaboração do trabalho, apresentando desde a classificação da pesquisa, e detalhando os métodos investigativo, de desenvolvimento e de análise de resultados;

Capítulo 5 - Sistema de Recomendação com Aprendizado Profundo: confere uma visão mais detalhada do sistema, procurando retomar o contexto, e apresentando outras nuances sobre os algoritmos de Aprendizado Profundo escolhidos, da preparação da base de dados e da aplicação;

Capítulo 6 - Análise de Resultados: revela os resultados obtidos, através das métricas pré-estabelecidas, do Sistema de Recomendação. Orienta-se pelo Método de Análise de Resultados estabelecido como base, e

Capítulo 7 - Conclusão: acorda algumas considerações finais do trabalho, retomando os objetivos alcançados com a realização do mesmo, bem como respostas às questões de desenvolvimento e de pesquisa estabelecidas no trabalho. Além disso, procura-se destacar sobre as principais contribuições e limitações do trabalho, assim como evoluções que ainda podem ser realizadas no contexto desse estudo.

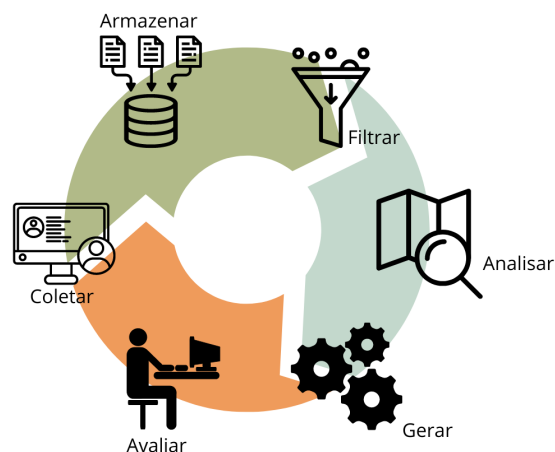
2 Referencial Teórico

Este Capítulo apresenta as bases teóricas que apóiam esse trabalho. Organizado em seções, tem-se: [Sistemas de Recomendação](#) (seção 1), que apresenta especificamente o conceito de Sistemas de Recomendação; [Inteligência Artificial](#) (seção 2), a qual confere conceitos e recursos inerentes à Inteligência Artificial, e que foram utilizados neste trabalho; [Satisfação do Usuário](#) (seção 3), que aborda a experiência do usuário e sua importância no que tange a satisfação do usuário. Por fim, no [Resumo do Capítulo](#) (seção 4), são apresentadas as considerações finais do capítulo.

2.1 Sistemas de Recomendação

Os Sistemas de Recomendação são aplicações de *software* que analisam e processam dados dos usuários com o propósito de sugerir itens que possam ser de interesse desses usuários. Assim, eles ajudam os usuários a descobrir novos produtos, conteúdos ou serviços que possam ser do seu interesse, personalizando suas experiências ([PHAM, 2019](#)).

Figura 2 – Ciclo dos Sistemas Recomendação



Fonte: Autora

O processo utilizado em um Sistema de Recomendação pode ser dividido em várias etapas, como demonstrado na Figura 2. O ciclo começa na coleta de dados (Coletar). Nesta etapa, o sistema coleta dados relevantes para análise do perfil do usuário, tais como: interações com o sistema; *feedbacks*, e fontes externas como as fontes de cunho social. Todos esses são dados válidos. Como [Jacobson et al. \(2016\)](#) exemplificam, o Spotify

coleta informações dos artistas, sinais acústicos das músicas, histórico, além de músicas ouvidas em tempo real.

Após a coleta, ocorrem o armazenamento e a filtragem desses dados (Armazenar e Filtrar). Em geral, o intuito é popular a base de dados. Quanto maior a quantidade de dados, melhor para a análise. Há necessidade ainda de separar esses dados para melhorar a acurácia do modelo (PHAM, 2019). Em seguida, tem-se a análise desses dados (Analisar).

No caso desse trabalho, foram utilizados algoritmos de Aprendizado Profundo para detectar padrões, e encontrar as preferências do usuário. Dessa forma, foi possível gerar recomendações em potencial (Gerar), as quais foram sugeridas para o usuário, entrando na última parte do ciclo, que é a avaliação pelo usuário da recomendação (Avaliar). Com base nessa avaliação, o modelo de recomendação recebeu mais dados do perfil daquele usuário e o ciclo começou, retornando à etapa de coleta de dados (Coletar).

2.2 Inteligência Artificial

Inteligência Artificial é um conjunto de tecnologias que permitem aos computadores executar uma variedade de funções avançadas, incluindo a capacidade de ver, entender e traduzir idiomas falados e escritos; analisar dados; fazer recomendações, e muito mais (SULEIMENOV et al., 2020).

No contexto de Sistemas de Recomendação, é feito o uso principalmente de cinco técnicas da Inteligência Artificial (PAJORSKA, 2023):

- Filtros colaborativos: essa técnica foca na similaridade entre diferentes usuários e itens. Usuários que são similares, provavelmente, possuem o mesmo tipo de interesse (PHAM, 2019), como demonstrado na Figura 3;

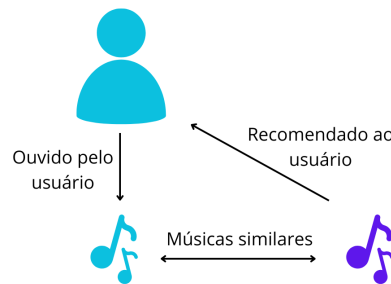
Figura 3 – Filtro Colaborativo



Fonte: Autora

- Filtros de conteúdo: essa técnica avalia a similaridade entre os itens. Itens similares aos que o usuário gostou/visualizou possuem mais probabilidade de serem de seu interesse ([PAJORSKA, 2023](#)), como demonstrado na Figura 4;

Figura 4 – Filtro de Conteúdo



Fonte: Autora

- Filtros demográficos: essa técnica busca categorizar os usuários com base em atributos, fazendo recomendações com base em características demográficas específicas dos usuários ([BURKE, 2002](#)). Por exemplo, no comércio eletrônico, essa técnica pode separar os clientes por gênero, segmentando os produtos com base em categorias tradicionalmente associadas ao gênero, tais como uma seção de roupas femininas e outra seção de roupas masculinas. Ou ainda, pode separar os clientes por localização, fornecendo roupas mais frescas para clientes, cujas localizações estão em áreas mais quentes, e roupas mais agasalhadas para clientes, cujas localizações estão em áreas mais frias;
- Filtros Baseados em Conhecimento e Utilidade: essa técnica recomenda com base nas necessidades dos usuários, calculando a utilidade de cada item para o usuário. Para isso, é necessário conhecimento do que o usuário precisa e quais os itens que suprem essa necessidade ([BURKE, 2002](#)). Por exemplo, um usuário pesquisa por um restaurante italiano específico (como Peccorino). Então, o Sistema de Recomendação buscaria quais são os itens que possuem maior utilidade para o usuário. Nesse caso, sugerindo outro restaurante italiano similar;
- Filtros Híbridos: essa técnica combina outras técnicas de filtragem, como por exemplo Filtros Colaborativos e Filtros de Conteúdo. Possui várias formas de combinar essas técnicas, que serão abordadas na seção [Classificação de Filtros Híbridos](#) (Cap. 2, seção 2.1.1). Esta técnica consegue lidar com as limitações das técnicas que ela combina. Pode-se, por exemplo, lidar com usuários de gostos diferenciados, além

de situações nas quais se tem poucos dados de interação do usuário. Portanto, isso confere uma forma de tratar o problema de "partida a frio" ([PAJORSKA, 2023](#)), e

- Filtros baseados em Aprendizado Profundo: essa técnica utiliza-se de outras técnicas de filtragem, e aplica sistemas de redes neurais para fazer previsões e recomendações, como Redes Neurais Convolucionais (RNC), para dados mais voltados para imagens, e Redes Neurais Recorrentes (RNR), para dados sequenciais ([NVIDIA, 2024](#)).

Para esse trabalho, foram abordados de forma mais aprofundada os seguintes filtros: [Filtros Híbridos](#) (Cap. 2, seção 2.1) e [Filtros Baseados em Aprendizado Profundo](#) (Cap. 2, seção 2.2).

2.2.1 Filtros Híbridos

A filtragem híbrida aborda mais de um tipo de técnica de filtragem, visando mitigar aspectos negativos que essas filtragens possuem ao serem tratadas separadamente, tais como os problemas de *cold start*; diversidade, e esparsidade dos dados. Esse modelo também surgiu na tentativa de melhorar a acurácia do processo de recomendação ([THORAT; GOUDAR; BARVE, 2015](#)). Como [Jung, Park e Lee \(2004\)](#) apresenta, a filtragem híbrida de Filtros Colaborativos com Filtros de Conteúdo apresenta uma melhora significativa nas recomendações de um Sistema de Recomendação, principalmente ao comparar os resultados dessa abordagem híbrida com os Filtros Colaborativos e os Filtros de Conteúdo isolados.

2.2.1.1 Classificação de Filtros Híbridos

De acordo com [Burke \(2002\)](#), os modelos com filtragem híbrida voltados para Sistemas de Recomendação podem possuir várias classificações com base em como combinam os tipos de filtro. Alguns dessas classificações são:

- Por peso (*Weighted*): Apresenta um sistema de peso, onde o peso de um item é calculado com base no resultado de sua avaliação por todas as técnicas presentes no sistema. Esse tipo toma como base o fato de que todas as técnicas produzem valores relativos similares. Entretanto, isso nem sempre é o caso. Assim, alguns itens com poucas classificações podem ser ignorados;
- Por troca (*Switching*): Ocorre troca entre diferentes técnicas de recomendação dependendo do contexto. Esse tipo adiciona uma complexidade a mais ao sistema, já que precisa definir o critério para troca entre as técnicas, mas também permite aproveitar os benefícios de várias técnicas;

- Por mistura (*Mixed*): Apresenta recomendações de mais de uma técnica juntas. Esse modo reduz parcialmente o problema de "patida a frio". Para novos itens, é possível classificá-los com base em suas descrições. Porém, para novos usuários, o problema de não ter informações suficiente permanece;
- Por combinação de *features* (*Feature Combination*): Usa os dados do filtro colaborativo como dados adicionais na base de dados. Adicionalmente, usa filtros de conteúdos na base aumentada que será gerada. Esse tipo considera os dados gerados pelos filtros colaborativos, sem necessariamente depender deles. Assim, reduz o problema da falta de dados: *cold start*;
- Por cascata (*Cascade*): Usa uma técnica para produzir uma classificação grosseira dos dados. Em seguida, usa outra técnica para refinar essa classificação. Esse tipo evita que itens sejam ignorados, como pode ocorrer na técnica de peso, na qual se aplica a técnica a todos os itens. Cabe ressaltar ainda que, no tipo cascata, as classificações são apenas refinadas e não substituídas;
- Por aumento de *features* (*Feature Augmentation*): Usa um modelo gerado por uma técnica de filtro para produzir uma classificação ou avaliação de um item. Essa informação produzida vai ser introduzida em outro modelo, gerado por outra técnica de filtro, a ser utilizada na sequência. Esse tipo de sistema híbrido pode aumentar o desempenho do sistema sem modificá-lo diretamente, e
- Por *Meta-level* (*Meta-level*): Usa um dos modelos gerados pelas técnicas como entrada para o outro modelo, sendo esse último gerado por outra técnica. Por exemplo, usar o modelo gerado por filtros colaborativos como entrada para um modelo gerado por filtro de conteúdo. Nesse caso em específico, o *Meta-level* tem o benefício de que o modelo final vai ser uma representação fiel dos interesses do usuário, facilitando a operação do sistema ao usar essa representação na comparação com dados crus de avaliação.

Os Quadros 1 a 7 mostram exemplos de sistemas existentes, possíveis ou propostos com as possíveis combinações de filtros híbridos. Cada quadro representa uma das classificações dos diferentes tipos de filtros híbridos, por exemplo, por combinação de *features*. Nas linhas, têm-se as possíveis combinações entre filtros, por exemplo, FC/FPC que seria Filtro Colaborativo em conjunto com Filtro Por Conteúdo. Nas colunas, têm-se os autores de modelos para essas classificações, se for possível a implementação.

Nesse sentido, têm-se exemplos de modelos, citados por autores, que existem ou foram propostos para cada combinação entre os filtros em cada uma das classificações. Há casos, inclusive, que ou não é possível a implementação dessa combinação de filtros com a classificação específica (ex. a combinação de Filtro Colaborativo com Filtro demográfico

na aplicação por Meta-level); ou é redundante o uso da combinação de filtros com a classificação (ex. a combinação de Filtro Por Conteúdo com Filtro Colaborativo em quatro casos: por peso, por mistura, por troca e por combinação de *features*). Nesses quatro casos, é indiferente a ordem dos filtros. Então, usando primeiro o Filtro Por Conteúdo ou o Filtro Colaborativo, o resultado seria o mesmo. Sendo assim, trata-se de algo redundante.

- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Peso
 - O Quadro 1 apresenta os exemplos de modelos para a classificação Por Peso;

Quadro 1 – Classificação por Peso

Filtros	Implementação
FC/FPC	P-Tango
FC/FD	(Pazzani 1999)
FC/FBCU	(Towle & Quinn 2000)
FPC/FC	Redundante
FPC/FD	(Pazzani 1999)
FPC/FBCU	
FD/FC	Redundante
FD/FPC	Redundante
FD/FBCU	
FBCU/FC	Redundante
FBCU/FPC	Redundante
FBCU/FD	Redundante

Fonte: (BURKE, 2002) - Tradução (Autora)

Legenda: (FC = Filtro Colaborativo, FPC = Filtro Por Conteúdo, FBCU = Filtro Baseado em conhecimento e utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam que não existem exemplos de modelos para os casos

- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Mistura - O Quadro 2 apresenta os exemplos de modelos para a classificação Por Mistura;
- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Troca
 - O Quadro 3 apresenta os exemplos de modelos para a classificação Por Troca;
- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Combinação de *Features* - O Quadro 4 apresenta os exemplos de modelos para a classificação Por Combinação de *Features*;

Quadro 2 – Classificação por Mistura

Filtros	Implementação
FC/FPC	PTV, ProfBuilder
FC/FD	
FC/FBCU	
FPC/FC	Redundante
FPC/FD	
FPC/FBCU	
FD/FC	Redundante
FD/FPC	Redundante
FD/FBCU	
FBCU/FC	Redundante
FBCU/FPC	Redundante
FBCU/FD	Redundante

Fonte: (BURKE, 2002) - Tradução
(Autora)

Legenda: (FC = Filtro Colaborativo,
FPC = Filtro Por Conteúdo, FBCU
= Filtro Baseado em conhecimento e
utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam
que não existem exemplos de
modelos para os casos

Quadro 3 – Classificação por Troca

Filtros	Implementação
FC/FPC	DailyLearner
FC/FD	
FC/FBCU	(Tran & Cohen, 2000)
FPC/FC	Redundante
FPC/FD	
FPC/FBCU	
FD/FC	Redundante
FD/FPC	Redundante
FD/FBCU	
FBCU/FC	Redundante
FBCU/FPC	Redundante
FBCU/FD	Redundante

Fonte: (BURKE, 2002) - Tradução
(Autora)

Legenda: (FC = Filtro Colaborativo,
FPC = Filtro Por Conteúdo, FBCU
= Filtro Baseado em conhecimento e
utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam
que não existem exemplos de
modelos para os casos

Quadro 4 – Classificação por Combinação de *Features*

Filtros	Implementação
FC/FPC	(Basu, Hirsh, & Cohen 1998)
FC/FD	
FC/FBCU	Não é possível
FPC/FC	Redundante
FPC/FD	(Condliff, et al. 1999)
FPC/FBCU	Não é possível
FD/FC	Redundante
FD/FPC	Redundante
FD/FBCU	Não é possível
FBCU/FC	Redundante
FBCU/FPC	Redundante
FBCU/FD	Redundante

Fonte: (BURKE, 2002) - Tradução (Autora)

Legenda: (FC = Filtro Colaborativo, FPC = Filtro Por Conteúdo, FBCU = Filtro Baseado em conhecimento e utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam que não existem exemplos de modelos para os casos

- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Cascata - O Quadro 5 apresenta os exemplos de modelos para a classificação Por Cascata;
- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por Aumento de *Features* - O Quadro 6 apresenta os exemplos de modelos para a classificação Por Aumento de *Features*, e
- Modelos Possíveis, Existentes (ou Propostos) de Recomendação Híbrida: Por *Meta-level* - O Quadro 7 apresenta os exemplos de modelos para a classificação Por *Meta-level*.

2.2.1.2 Possíveis Implementações

Os filtros híbridos podem ser implementados ao combinar outros filtros. No quesito de implementação da combinação desses filtros, fortemente baseada nas classificações dos filtros híbridos, encontram-se as possibilidades exemplificadas por Thorat, Goudar e Barve (2015):

Quadro 5 – Classificação por Cascata

Filtros	Implementação
FC/FPC	Fab
FC/FD	
FC/FBCU	
FPC/FC	
FPC/FD	
FPC/FBCU	
FD/FC	
FD/FPC	
FD/FBCU	
FBCU/FC	EntreeC
FBCU/FPC	
FBCU/FD	

Fonte: (BURKE, 2002) - Tradução
(Autora)

Legenda: (FC = Filtro Colaborativo,
FPC = Filtro Por Conteúdo, FBCU
= Filtro Baseado em conhecimento e
utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam
que não existem exemplos de
modelos para os casos

Quadro 6 – Classificação por Aumento de *Features*

Filtros	Implementação
FC/FPC	Libra
FC/FD	
FC/FBCU	
FPC/FC	
FPC/FD	
FPC/FBCU	
FD/FC	
FD/FPC	
FD/FBCU	
FBCU/FC	GroupLens (1999)
FBCU/FPC	
FBCU/FD	

Fonte: (BURKE, 2002) - Tradução
(Autora)

Legenda: (FC = Filtro Colaborativo,
FPC = Filtro Por Conteúdo, FBCU
= Filtro Baseado em conhecimento e
utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam
que não existem exemplos de
modelos para os casos

Quadro 7 – Classificação por *Meta-level*

Filtros	Implementação
FC/FPC	
FC/FD	Não é possível
FC/FBCU	
FPC/FC	Fab, (Condliff, et al. 1999), LaboUr
FPC/FD	Não é possível
FPC/FBCU	
FD/FC	Não é possível
FD/FPC	
FD/FBCU	
FBCU/FC	
FBCU/FPC	
FBCU/FD	Não é possível

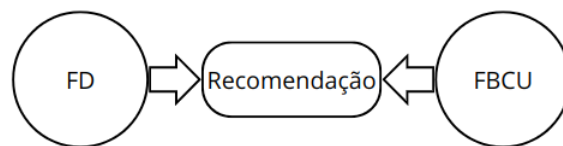
Fonte: (BURKE, 2002) - Tradução (Autora)

Legenda: (FC = Filtro Colaborativo, FPC = Filtro Por Conteúdo, FBCU = Filtro Baseado em conhecimento e utilidade, FD = Filtro Demográfico)

OBS: Espaços em branco significam que não existem exemplos de modelos para os casos

- Implementar os filtros separadamente, combinando-os depois para realizar a recomendação, como exemplificado na Figura 5, que combina os filtros demográficos e baseados em conhecimento e utilidade;

Figura 5 – Combinação Individual

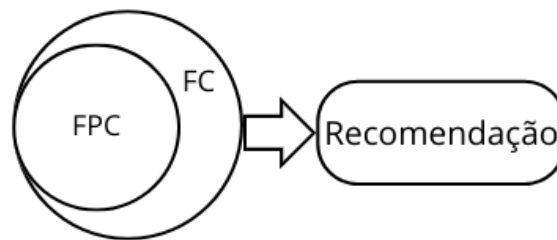


Fonte: (THORAT; GOUDAR; BARVE, 2015) - Tradução e Adaptação (Autora)

Legenda: (FBCU = Filtro Baseado em conhecimento e utilidade, FD = Filtro Demográfico)

- Implementar características do primeiro filtro no segundo filtro, como exemplificado na Figura 6, que combina os filtros de conteúdo e colaborativo;
- Implementar características do segundo filtro no primeiro filtro, como exemplificado na Figura 7, que combina os filtros de conteúdo e colaborativo, e
- Unificar os filtros em um único modelo, como exemplificado na Figura 8, que combina os filtros colaborativo e baseados em Aprendizado Profundo.

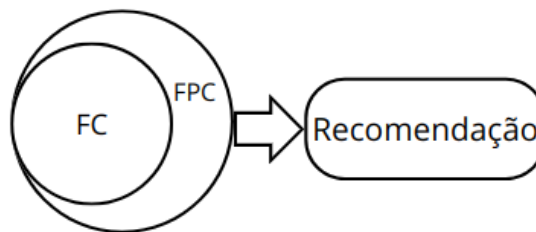
Figura 6 – Agregar Características do Filtro de Conteúdo no Filtro Colaborativo



Fonte: (THORAT; GOUDAR; BARVE, 2015) - Tradução e Adaptação (Autora)

Legenda: (FC = Filtro Colaborativo, FPC = Filtro Por Conteúdo)

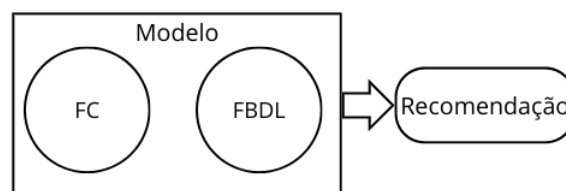
Figura 7 – Agregar Características do Filtro Colaborativo no Filtro de Conteúdo



Fonte: (THORAT; GOUDAR; BARVE, 2015) - Tradução e Adaptação (Autora)

Legenda: (FC = Filtro Colaborativo, FPC = Filtro Por Conteúdo)

Figura 8 – Ambos os Filtros em um Único Modelo



Fonte: (THORAT; GOUDAR; BARVE, 2015) - Tradução e Adaptação (Autora)

Legenda: (FC = Filtro Colaborativo, FBAP = Filtro Baseado em Aprendizado Profundo)

2.2.2 Filtros Baseados em Aprendizado Profundo

Aprendizado Profundo é uma sub-área de Inteligência Artificial que se concentra no treinamento de algoritmos para realização de tarefas complexas de forma autônoma. Permite ainda que esses algoritmos aprendam representações de dados com múltiplas camadas de abstração (LECUN; BENGIO; HINTON, 2015). Focando em Sistemas de Recomendação, pode-se melhorar outros filtros existentes ao usar Aprendizado Profundo. Nesse sentido, há quatro possíveis abordagens: Redes Neurais Convolucionais; Redes Neurais Recorrentes; Máquina de Boltzmann Restrita e *Autoencoder* (SISI et al., 2020).

2.2.2.1 Redes Neurais Convolucionais

Rede Neural Convolucional é um tipo de rede neural com camadas convolucionais e operações de agrupamento (SISI et al., 2020).

No escopo de Sistemas de Recomendação, são ideais para processar dados de multimídia, conseguindo associar dados de diferentes formatos, melhorando a precisão das recomendações.

2.2.2.2 Redes Neurais Recorrentes

Rede Neural Recorrente é um tipo de rede neural que possui *loops* e guarda cálculos anteriores. Ideal para modelar dados sequenciais (SISI et al., 2020).

Aplicando esse método em Sistemas de Recomendação, consegue-se identificar padrões no comportamento e nas interações dos usuários, e criar recomendações baseadas na navegação do usuário na rede, sem precisar de dados iniciais informados pelos usuários. Esse último caso ocorre na geração de *cookies* (SISI et al., 2020). Dessa forma, pode-se lidar com o problema da "partida a frio".

2.2.2.3 Máquina de Boltzmann Restrita

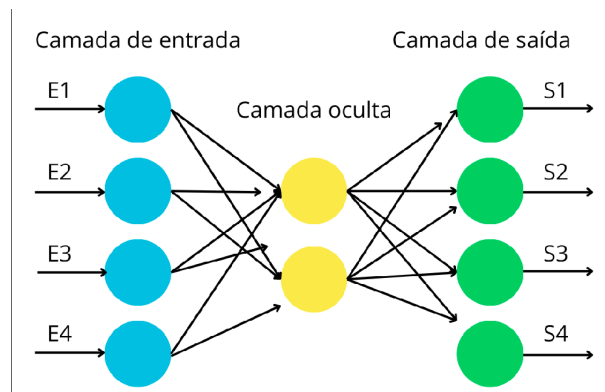
A Máquina de Boltzmann Restrita é uma rede neural de dupla camada, sendo uma visível e outra escondida (SISI et al., 2020).

Para Sistemas de Recomendação, pode ser interligada com filtros colaborativos, aumentando o tamanho da base de dados, e melhorando o processo de recomendação (SISI et al., 2020). Ela também pode ser treinada para aprender uma representação dos padrões de interação entre usuários e itens para, posteriormente, gerar recomendações com base nas preferências dos usuários e nas características dos itens.

2.2.2.4 Autoencoder

Autoencoder é uma rede neural que tenta reconstruir seu dado de entrada em um dado similar na saída, ao usar uma camada mediadora (SISI et al., 2020). Sendo assim, possui três camadas, nas quais o número de neurônios na entrada deve ser igual ao número de neurônios na saída, como exemplificado na Figura 9.

No contexto de Sistemas de Recomendação, essa rede neural tem a habilidade de reduzir e reconstruir os dados, extraindo os atributos destes. Por isso, um *Autoencoder* pode ser treinado para aprender representações latentes dos itens, as quais capturam as características essenciais dos itens e podem ser usadas para recomendar itens semelhantes com base em similaridade. Essa rede neural pode ser usada para aprender representações dos usuários e itens a partir de interações históricas de usuários com itens, provendo

Figura 9 – Estrutura de um *Autoencoder*

Fonte: (SISI et al., 2020) - Tradução (Autora)

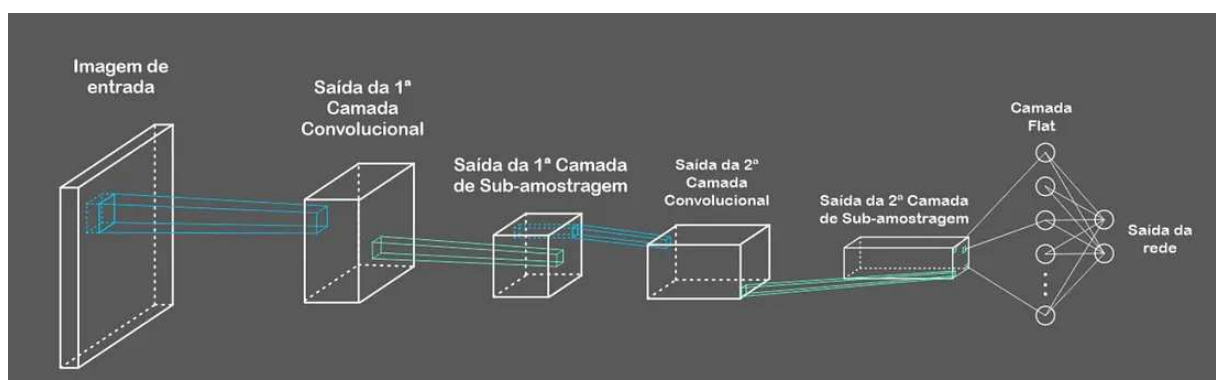
recomendações com base em padrões de comportamento. Por fim, pode-se ainda aprender diferentes tipos de informações com base em diferentes tipos de dados de entrada (ex. áudio, imagem e texto), combinando essas modalidades para gerar a recomendação (ZHANG; LIU; JIN, 2020).

2.2.3 Modelo Utilizado

O modelo escolhido pra este trabalho foi a Rede Neural Convolutiva. A justificativa dessa escolha está mais detalhada na [Solução Inicial para o Sistema de Recomendação](#) (Cap. 5, seção 2).

As CNNs dividem-se em camada convolutiva, a de *pooling* e a de *flat*. A Figura 10 representa a arquitetura dessa rede neural.

Figura 10 – Arquitetura RNC



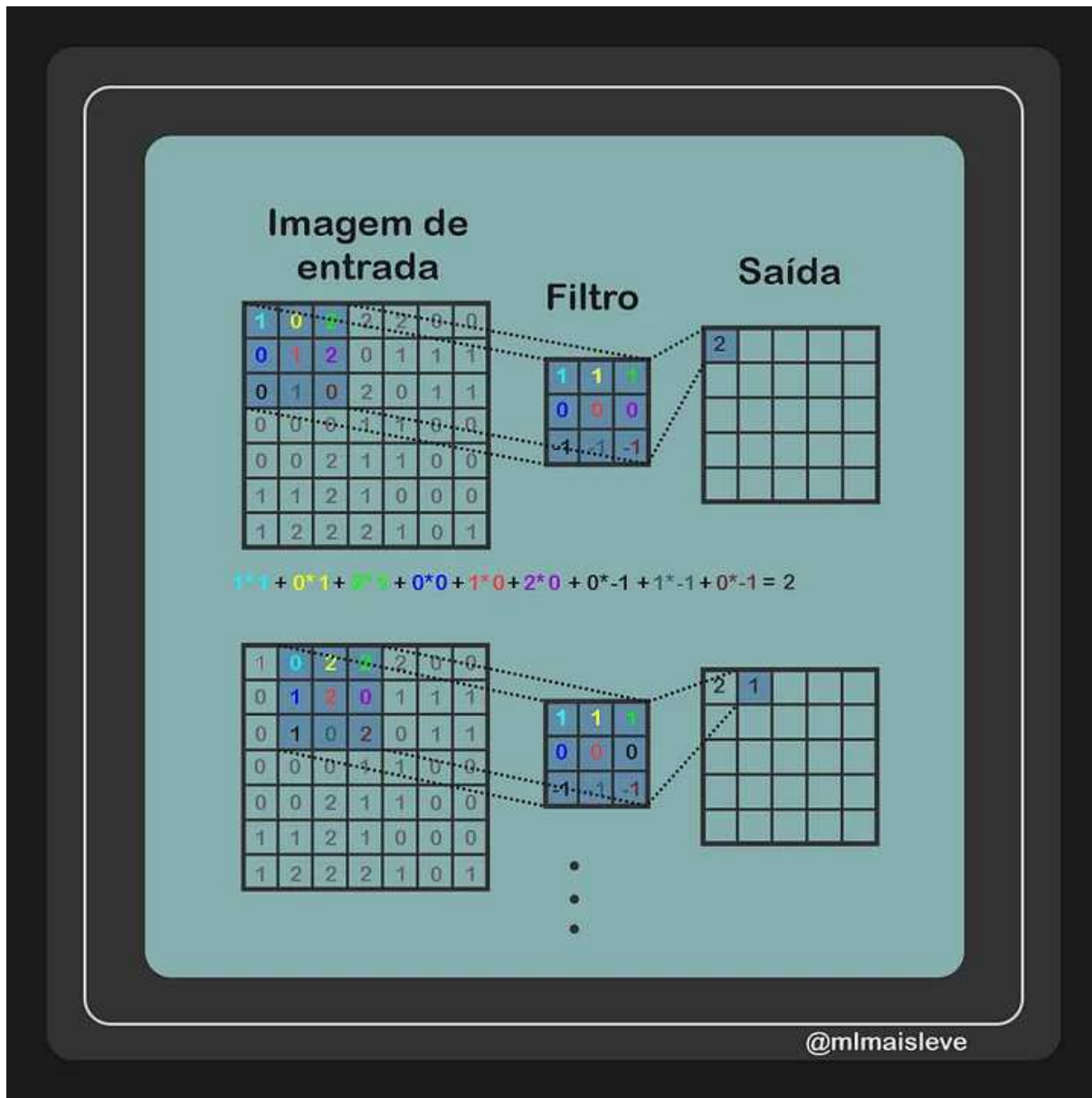
Fonte: Rodrigues (2019)

- Camada Convolutiva: é o resultado da aplicação de filtros (matrizes de valores). A cada passo, os elementos dos filtros são multiplicados pelo elemento de mesma posição na região em que o filtro está sendo aplicado. No fim, os resultados dessas

multiplicações são somados para ter um único valor como saída (RODRIGUES, 2019).

A Figura 11 representa esse processo.

Figura 11 – Aplicação de Filtros - RNC



Fonte: Rodrigues (2019)

- Camada de *Pooling*: reduz as dimensões das camadas, preservando a profundidade das mesmas. Nesse caso, carrega-se para a próxima camada o maior valor dentro dos campos de recepção (RODRIGUES, 2019).
- Camada de *Flat*: transforma as camadas convolucionais em uma camada *flat 1D*, ou seja, em vetores de características.

Após passar por essas camadas, a RNC passa a se comportar como uma rede neural tradicional e apresenta uma saída, podendo essa ser previsão ou classificação.

2.3 Satisfação do Usuário

Segundo [Griffiths, Johnson e Hartley \(2007\)](#), a satisfação de um usuário durante o uso de um sistema é subjetiva, e varia de acordo com os elementos característicos do sistema, os gostos e as percepções do usuário, bem como a tarefa que estão realizando naquele sistema. Ainda de acordo com [Griffiths, Johnson e Hartley \(2007\)](#), a satisfação do usuário tem o objetivo de prover ao sistema pontos relevantes sobre seu desempenho e sobre a própria experiência do usuário ao usar o sistema.

No quesito de experiência do usuário, esta pode ser definida como as respostas e percepções de uma pessoa ao usar um produto, sistema ou serviço ([ISO, 2009](#)). Entretanto, também é possível dizer que a experiência de usuário vai além de avaliar a usabilidade/funcionalidade durante a interação do usuário. Na verdade, a experiência de usuário tende a cobrir não só os comportamentos do sistema, bem como a eficiência e a efetividade ([ALLAM; DAHLAN, 2013](#)).

[Applegate \(1993\)](#) mostra que existem três modelos de satisfação do usuário: modelo de satisfação material, modelo de satisfação emocional simples, e modelo de satisfação emocional múltiplo. Entretanto, esses modelos são subjetivos e difíceis de mensurar. Assim, de acordo com [Zviran e Erlich \(2003\)](#), vários estudiosos desenvolveram diferentes ferramentas para tentar mensurar a satisfação do usuário, como Bailey e Pearson que criaram um questionário de 39 itens com diferentes métodos de construção. Porém, [Zviran e Erlich \(2003\)](#) ainda mencionam que, apesar de todos os estudos a cerca do tema, nem sempre as respostas são precisas. Dessa forma, os estudos sobre o tema continuaram e novos modelos foram desenvolvidos.

No contexto desse trabalho, foi considerado o modelo de [Mahmood et al. \(2000\)](#) que tem três focos principais:

- Benefícios e conveniência: voltado para expectativa do usuário, facilidade de uso e utilidade percebida;
- Antecedentes do usuário: voltado para a experiência do usuário ao utilizar o sistema; habilidades do usuário com sistemas similares, e envolvimento do usuário no desenvolvimento do sistema, e
- Suporte e estímulo organizacional: voltado para a atitude do usuário em relação ao sistema de informação, o qual avalia o grau de satisfação com o sistema de informação provido pela organização, e suporte organizacional, o qual indica o grau de apoio da organização para facilitar o uso e a adoção do sistema.

Em um primeiro momento, a ideia é não lidar de forma mais aprofundada com o foco "benefícios e conveniência", uma vez que não há pretensão em usar testes de usabilidade e similares para aferir aspectos como usabilidade e associados. No caso do foco "antecedentes do usuário", também não foi intenção da autora desse trabalho avaliar o usuário em suas habilidades com sistemas similares ou mesmo envolver o usuário no desenvolvimento do sistema proposto. Entretanto, naturalmente, ao conferir se o Sistema de Recomendação proposto atende de forma mais adequada ou não o usuário nas recomendações, permitiu concluir que a proposta - comparativamente com o que o usuário conhece (i.e. experiência do usuário com sistemas similares) - é competitiva ou não, respectivamente.

Para o foco "suporte e estímulo organizacional", foi considerada a própria autora como a "organização provedora do sistema", uma vez que a autora desenvolveu o Sistema de Recomendação proposto sozinha, e não em equipe. Além disso, esse foco foi o principal embasador para aferir a satisfação do usuário, tanto com o Sistema de Recomendação proposto, quanto com a organização (lê-se autora) em prover facilitadores para uso e adoção do sistema.

Dessa forma, esse trabalho teve como intuito levantar considerações dos usuários acerca do sistema, e usar esses dados para avaliar um Sistema de Recomendação centrado em Inteligência Artificial. Para isso, foi desenvolvido e treinado um modelo de Sistema de Recomendação, o qual foi integrado em uma API (em português, traduzido como Interface de Programação de Aplicações). Entretanto, a API apresentou um grande tempo para responder os *requests*, maiores detalhes na API (Cap. 5, seção 3.2), e optou-se por realizar mais um questionário com os usuários para adquirir informações para gerar a primeira recomendação. Em seguida, após geradas as primeiras recomendações, foi apresentado um questionário para os usuários, baseados nos focos do modelo de Mahmood et al. (2000), para determinar a satisfação deles com o sistema.

Com base nas respostas do questionário, foi incrementada a base de dados do modelo, permitindo treinar novamente esse modelo. Posteriormente, o modelo retreinado foi apresentado aos usuários, gerando um segundo ciclo de respostas. Essas respostas foram comparadas com as respostas da primeira avaliação, possibilitando analisar evoluções ou não do sistema em termos de recomendação.

2.4 Resumo do Capítulo

Esse capítulo apresentou uma ideia geral do que são Sistemas de Recomendação. Trata-se de um sistema que visa recomendar algo que seja do interesse dos usuários. Esse processo envolve etapas, sendo as mesmas: Coletar Dados, Armazenar Dados, Filtrar Dados, Analisar Dados, Gerar Recomendações e Avaliar Recomendações. Foram abordados

diversos filtros para Sistemas de Recomendação usando Inteligência Artificial, sendo eles: colaborativo, por conteúdo, demográfico, baseado em conhecimento e utilidade, híbridos e baseados em Aprendizado Profundo.

Com foco em filtros híbridos e filtros baseados em Aprendizado Profundo, foi detalhado como eles funcionam e suas definições. Para os filtros híbridos, que são combinações de outros filtros (colaborativo e por conteúdo, no contexto deste trabalho), foi apresentada sua classificação: por peso, que pondera os resultados dos filtros que são combinados; por mistura, que apresenta em conjunto os resultados dos outros filtros; por troca, que alterna entre os filtros com base em um critério; por combinação de *features*, que usa dados de um dos filtros para aumentar a base de dados do outro filtro a ser usado; por cascata, que usa um filtro para produzir uma classificação e outro filtro para refinar essa classificação; por aumento de *features*, que o resultado de um filtro é a entrada de outro filtro; e por *meta-level*, que o modelo gerado por um filtro é usado de entrada em outro filtro.

Para os filtros baseados em Aprendizado Profundo, foram cobertos quatro métodos, sendo eles: Redes Neurais Convolucionais, que auxiliam na interpretação e na associação de diferentes formatos de dados; Redes Neurais Recorrentes, que aprendem padrões de uso do usuário; Máquina de Boltzmann Restrita, que geram representações de usuários e itens para melhorar o processo de recomendação; e *Autoencoder*, que pode tanto combinar diferentes multimídias, quanto gerar representações de usuários para melhorar as recomendações. Detalha-se ainda o funcionamento de uma RNC, modelo de rede neural que foi utilizado no estudo.

Essa capítulo ainda trata a satisfação do usuário em Sistemas de Recomendação, e como isso foi abordado no trabalho.

3 Suporte Tecnológico

Este Capítulo descreve as ferramentas e tecnologias utilizadas na elaboração deste trabalho, sendo separadas em seções com base no tipo de apoio provido. As ferramentas que apóiam a pesquisa dos tópicos de interesse desse trabalho estão em [Apoio à Pesquisa](#) (seção 1). Já as ferramentas que conferem apoio mais técnico, destacando linguagem de programação e plataforma de desenvolvimento, estão em [Apoio ao Desenvolvimento](#) (seção 2). As ferramentas que auxiliam na escrita deste trabalho são apresentadas em [Apoio à Elaboração da Monografia](#) (seção 3). Por fim, tem-se o [Resumo do Capítulo](#) (seção 4).

3.1 Apoio à Pesquisa

Essa seção é voltada para as ferramentas utilizadas no apoio à pesquisa de modelos de Inteligência Artificial, bem como à exemplificação de bases de dados.

3.1.1 Kaggle

O Kaggle é uma comunidade *online* de cientistas de dados e praticantes de aprendizado de máquina, que oferece acesso a conjuntos de dados diversificados e desafios competitivos que impulsionam a inovação. Possui uma grande diversidade de conjuntos de dados e modelos disponíveis, os quais os pesquisadores podem explorar e experimentar para aprimorar seus modelos de dados ([KAGGLE, 2024](#)). Foi utilizada no trabalho no intuito de explorar modelos de Inteligência Artificial já aplicados na comunidade, servindo de apoio na construção do modelo desse trabalho. Além disso, esse suporte apresenta conjuntos de dados que são relevantes para a montagem da base de dados usada para treinamento do modelo. Foi escolhida essa comunidade pela grande quantidade de base de dados disponíveis e modelos pré treinados que puderam servir de base para o desenvolvimento. Assim, os modelos escolhidos para auxiliar no desenvolvimento, iniciado com as provas de conceito, foram:

- Filtros Colaborativos, de Conteúdo e Híbridos: modelos do [Pathak \(2020\)](#), e do [Ahmed \(2024\)](#);
- Rede Neural Convolucional: modelo do [Datai \(2020\)](#);
- Rede Neural Recorrente: modelo do [Bazdyrev \(2019\)](#);
- Máquina Restrita de Boltzmann: modelo do [Musib \(2023\)](#), e

- *Autoencoder*: modelo do [Datascience \(2024\)](#).

Já para base de dados, foi selecionada uma porcentagem da base [Parashar \(2023\)](#).

3.1.2 HuggingFace

O HuggingFace oferece uma biblioteca de modelos pré-treinados, bem como uma plataforma para compartilhamento e colaboração na criação de modelos de linguagem natural. Com sua vasta coleção de modelos, os pesquisadores têm à disposição recursos vantajosos para aprimorar a eficiência e a precisão de modelos ([FACE, 2024](#)). No presente trabalho, foi utilizada essa biblioteca visando apoiar o aprendizado durante o treinamento do modelo, tendo à disposição vários modelos pré-treinados que servirão de inspiração e possíveis referências, e também visando apoiar a implementação do modelo na API (*Application Programming Interface*).

3.2 Apoio ao Desenvolvimento

Essa seção foca em ferramentas utilizadas no processo de desenvolvimento prático desse trabalho.

3.2.1 Python

O Python é uma das linguagens de programação mais populares no campo da Inteligência Artificial e da ciência de dados, a qual oferece bibliotecas e *frameworks* especializados, tais como *TensorFlow*, *PyTorch* e *Scikit-learn*. Esses recursos facilitam o desenvolvimento e a implementação de algoritmos de recomendação. Sua sintaxe clara e expressiva e sua flexibilidade tornam a linguagem Python uma das melhores opções para projetos com Aprendizado de Máquina ([FOUNDATION, 2024](#)). Dessa forma, é a linguagem que foi utilizada nesse trabalho para desenvolver o modelo de Aprendizado Profundo.

3.2.2 Colab/TPU

O Google Colab é uma plataforma baseada em nuvem que permite o desenvolvimento de código Python de forma colaborativa e totalmente gratuita. Além disso, oferece acesso a recursos de TPU (*Tensor Processing Units*) diretamente do Google, sem precisar consumir *hardware* local. Entende-se por TPU como um conjunto integrado de aplicações específicas, provido pelo Google, para justamente lidar com redes neurais. Considerando que o treinamento de modelos demanda muita memória, ter esse acesso torna mais simples e eficiente ao treinar os modelos durante o desenvolvimento ([GOOGLE, 2024](#)). Por

isso, essa plataforma foi utilizada para o desenvolvimento do modelo em conjunto com a linguagem Python.

3.2.3 Google *Forms* e Google *Sheets*

O Google *Forms* é uma plataforma gratuita de formulários *online*, que possibilita a criação de questionários, pesquisas, formulários, entre outros. A ferramenta permite uma variedade de tipos de perguntas, oferecendo flexibilidade na criação de questionários. Além de ser de fácil acesso, através de qualquer dispositivo conectado à internet, também há integração com o Google *Sheets*, levando as respostas dos formulários para planilhas, o que facilita a visualização e a análise dos dados. Por isso, para validar a satisfação dos usuários após o desenvolvimento da aplicação foram criados questionários no Google *Forms*. Para melhor visualização dos dados, os mesmos foram transferidos para o Google *Sheets*, conferindo facilitadores para uma análise mais adequada.

3.3 Apoio à Elaboração da Monografia

Essa seção apresenta as ferramentas utilizadas para redigir esse trabalho.

3.3.1 LaTeX

O LaTeX é um sistema de preparação de documentos que se concentra na estruturação lógica do conteúdo, permitindo uma formatação consistente e padronizada do trabalho (TEAM, 2024). O uso do LaTeX deu-se por sua versatilidade no quesito de formatar os documentos, utilizando um modelo já pré-estabelecido pela instituição de ensino.

3.3.2 Git/GitHub

O Git e o GitHub são ferramentas para controle de versionamento, sendo possível acompanhar as alterações no texto; facilitar a revisão, e armazenar o projeto como um todo, garantindo a consistência e a integridade do trabalho (GITHUB, 2024).

3.3.3 Canva

O Canva (CANVA, 2024) é um site que permite a criação e a edição de imagens, sendo possível o *download* em diferentes formatos. Dessa forma, foi a ferramenta utilizada na elaboração e na edição de imagens desse trabalho. O uso dessa ferramenta deu-se devido à, em sua versão gratuita, conferir várias ferramentas de criação de edição de imagem, além de diferentes formatos para baixar as imagens.

3.3.4 Lucidchart

O Lucidchart é uma plataforma web de diagramação e visualização de informações, com diversos modelos pré-definidos para diferentes tipos de diagramas, como fluxogramas, organogramas, entre outros ([LUCIDCHART, 2024](#)). Sua versão gratuita disponibiliza não só vários modelos de diagramas, como também ferramentas e formas. Por essa facilidade, a plataforma foi usada no trabalho para a criação de diagramas.

3.4 Resumo do Capítulo

Neste Capítulo, foram descritas as tecnologias que auxiliaram na elaboração deste trabalho. Foram apresentadas as ferramentas que apoiaram a pesquisa, com menção ao Kaggle e ao HuggingFace. Como ferramentas de apoio ao desenvolvimento, destacou-se a linguagem Python e a plataforma Colab. Por fim, foram apresentadas as ferramentas que auxiliam na elaboração dessa monografia, com foco em LaTeX, Git/GitHub e Canva.

O Quadro 8 apresenta as ferramentas utilizadas nas suas respectivas versões, com uma breve descrição e a documentação ou site oficial.

Quadro 8 – Principais Ferramentas Utilizadas no Trabalho

Ferramenta	Versão	Descrição	Documentação
Kaggle	-	Comunidade de cientistas de dados voltado para IA	(KAGGLE, 2024)
HuggingFace	-	Comunidade de cientistas de dados voltado para IA	(FACE, 2024)
Python	3	Linguagem de programação de alto nível	(FOUNDATION, 2024)
Scikit-surprise	1.1.4	Biblioteca do Python	(DOCUMENTATION, 2024a)
Tensorflow	2.15.0	Biblioteca do Python	(DOCUMENTATION, 2024b)
Wurlitzer	3.1.1	Biblioteca do Python	(DOCUMENTATION, 2024c)
Colab	-	Plataforma de hospedagem de código	(GOOGLE, 2024)
LaTeX	LaTeX 2e2	Linguagem de marcação	(TEAM, 2024)
Git	2.44.0	Sistema de versionamento de dados	(GIT, 2024)
GitHub	-	Sistema de versionamento de dados	(GITHUB, 2024)
Canva	-	Edição de imagem	(CANVA, 2024)
Lucidchart	-	Plataforma de diagramação	(LUCIDCHART, 2024)
Google Forms	-	Ferramenta de formulários	(FORMS, 2024)
Google Sheets	-	Ferramenta de planilhas	(SHEETS, 2024)

Fonte: Autora

4 Metodologia

Este Capítulo aborda a metodologia usada no desenvolvimento desse trabalho. Organizado em seções, tem-se: [Classificação de Pesquisa](#) (seção 1), que apresenta a abordagem, a natureza, os objetivos e os procedimentos desse trabalho; [Método de Pesquisa Bibliográfica](#) (seção 2), na qual o processo de consulta à literatura especializada é apresentado; [Método de Desenvolvimento](#) (seção 3), a qual traz o método usado para auxiliar durante o desenvolvimento; [Satisfação do Usuário](#) (seção 4), que detalha o processo de coleta e análise dos resultados obtidos; [Fluxo das Atividades/Subprocessos](#) (seção 5), que apresenta o fluxograma das atividades desse trabalho, bem como um detalhamento de cada etapa; [Cronogramas](#) (seção 6), que mostra os cronogramas definidos para esse trabalho. Por fim, no [Resumo do Capítulo](#) (seção 7), são apresentadas as considerações finais do Capítulo.

4.1 Classificação de Pesquisa

De acordo com [Gil \(2010\)](#), pesquisa é um procedimento racional e sistemático que visa trazer soluções a problemas propostos. Adicionalmente, pesquisa desenvolve-se mediante o uso do conhecimento disponível, da utilização de técnicas e de outros procedimentos científicos. Nesse contexto, a metodologia entra como um validador do caminho escolhido para conduzir a pesquisa, vai além das descrições técnicas dos métodos, cuja função é guiar e estabelecer regras e procedimentos para a realização da pesquisa ([GERHARDT; SILVEIRA, 2009](#)). A pesquisa classifica-se quanto sua abordagem, sua natureza, seus objetivos e procedimentos. Adiante, será abordado em quais classificações esse trabalho está inserido, contextualizando sua relevância e a aplicação desses conceitos na prática de pesquisa acadêmica e científica.

4.1.1 Abordagem

A abordagem é híbrida, trazendo aspectos da abordagem qualitativa na avaliação da satisfação do usuário ao utilizar o sistema, de maneira subjetiva, levando em conta a particularidade de cada usuário, principalmente ao considerar que Sistemas de Recomendação são voltados unicamente para os usuários, e a experiência de cada um pode variar de acordo com seu perfil. Adicionalmente, apresenta aspectos da abordagem quantitativa ao utilizar métricas de precisão e acurácia de modelos de Aprendizado Profundo, trazendo dados numéricos resultantes de uma avaliação estatística do modelo.

4.1.2 Natureza

A natureza da pesquisa é aplicada, pois é voltada para a solução de um problema específico de como se pode desenvolver um Sistema de Recomendação com IA, e como esse sistema proporciona ou não maior satisfação aos seus usuários.

4.1.3 Objetivos

Esse trabalho é caracterizado como pesquisa exploratória, já que esse tipo de pesquisa facilitam a familiaridade do pesquisador com o objeto de pesquisa, para assim tornar a questão mais clara, no caso, o uso de Aprendizado Profundo em Sistemas de Recomendação, e como esse uso pode trazer ou não satisfação aos seus usuários.

4.1.4 Procedimentos

Quanto aos procedimentos, esse trabalho utilizou pesquisa bibliográfica, sendo esta um trabalho exploratório que buscou aumentar o conhecimento da autora sobre o assunto abordado, conferindo a base teórica do estudo ([NASCIMENTO; SOUSA, 2016](#)). Além desse procedimento, esse trabalho ainda utilizou pesquisa-ação, onde certa situação-problema é investigada ([ENGEL, 2000](#)). No caso, a situação investigada foi o uso de Aprendizado Profundo em Sistemas de Recomendação, orientando-se pela satisfação do usuário, e visando insumos que gerassem aprendizado sobre a situação. Nesse contexto, este trabalho visou apresentar a implementação de um Sistema de Recomendação, e validar com os usuários sua satisfação.

4.2 Método de Pesquisa Bibliográfica

A fim de aumentar o conhecimento sobre os assuntos abordados neste trabalho, a autora utilizou a base do Google Acadêmico, que possui diversos artigos de diferentes áreas do conhecimento, para coletar artigos relacionados. Para realizar a busca desses artigos, utilizou-se palavras-chave das principais áreas que compreendem esse trabalho, dando preferência para palavras-chave em inglês que aumentavam significativamente os resultados da busca. As principais palavras-chave utilizadas foram:

- *Deep Learning* - Usada para aumentar o conhecimento acerca de aprendizado de máquina; suas classificações, e atuais aplicações, bem como para filtrar quais algoritmos se encaixavam melhor com Sistemas de Recomendação;
- *Recommender System* - Usada para entender o funcionamento de um Sistema de Recomendação; seus principais problemas, e quais soluções já existiam para estes problemas;

- *Artificial Intelligence in Recommendation* - Usada para ver as principais aplicações de Inteligência Artificial em Sistemas de Recomendação; os problemas conhecidos dessas aplicações, e se já existiam soluções aplicadas para estes problemas, e
- *Deep Learning in Recommendation* - Usada em conjunto com os artigos encontrados para a palavra *Deep Learning*, para separar os possíveis modelos de aprendizado de máquina adequados aos Sistemas de Recomendação.

Além dessas palavras-chave, para cada solução e/ou modelo encontrada(o) com o uso dessas palavras-chave, foram realizadas novas pesquisas usando como base novas palavras-chave associadas à solução ou ao modelo. A intenção foi compreender ainda mais aplicações, conceitos e problemas inerentes a esses contextos. Por exemplo, a partir da palavra-chave "*Recommender System*", encontrou-se o conceito de "*Collaborative Filter*". Portanto, também ocorreu nova pesquisa usando a palavra-chave encontrada, no caso, "*Collaborative Filter*". Entende-se por Filtros Colaborativos como uma forma de aplicar Inteligência Artificial para desenvolver um Sistema de Recomendação.

Para selecionar os artigos dentre os resultados encontrados na base do Google Acadêmico, usou-se como principal filtro a quantidade de citações que o artigo possui. Já como filtro secundário, optou-se pelo ano de publicação do artigo. A partir dessa filtragem inicial, foi realizada a leitura do resumo dos artigos. Orientando-se por essa leitura, foram selecionados os artigos mais relevantes para os temas abordados nesse trabalho. Além disso, esses artigos foram lidos na íntegra, servindo de base bibliográfica nesse trabalho. Visando uma maior compreensão sobre os artigos obtidos, bem como os principais artigos que embasam o presente estudo, tem-se os Quadros 9 e 10.

4.3 Método de Desenvolvimento

Essa seção apresenta o método utilizado para o desenvolvimento do Sistema de Recomendação. Para este trabalho, optou-se por elementos da metodologia ágil Scrum, que se adaptavam ao contexto do que foi desenvolvido.

O Scrum é uma metodologia que atua como um guia, com papéis bem definidos, e que confere visibilidade ao autor do projeto sobre exatamente como está seu andamento (PEREIRA; TORREÃO; MARÇAL, 2007). Trata-se de uma metodologia que possui um conjunto de papéis, eventos, artefatos e regras que auxiliam os desenvolvedores a se orientarem durante o projeto.

Ao fazer uso do Scrum nesse trabalho, levando em consideração principalmente sua versatilidade e sua agilidade no acompanhamento do projeto, buscou-se aplicar alguns eventos como *sprints*, reuniões de planejamento, retrospectivas. Foram escolhidos esses eventos do Scrum, ao considerar que a autora atuou sozinha no desenvolvimento do

Quadro 9 – Artigos Retornados na Base do Google Acadêmico

Palavra Chave Utilizada	Artigos Retornados
<i>"Deep Learning"</i>	6.060.000
<i>"Recommender System"</i>	782.000
<i>"Recommender Systems Profits"</i>	28.800
<i>"Recommender Systems Costs"</i>	159.000
<i>"Artificial Intelligence in Recommendation"</i>	2.280.000
<i>"Deep Learning in Recommendation"</i>	2.510.000
<i>"Collaborative Filter"</i>	2.650.000
<i>"Hybrid Filtering Recommender Systems"</i>	149.000
<i>"Content Filtering Recommender Systems"</i>	290.000
<i>"Filtering in Artificial Intelligence"</i>	1.590.000
<i>"Convolutional Neural Networks"</i>	2.050.000
<i>"Boltzmann Restricted Machine"</i>	187.000
<i>"Recurrent Neural Network"</i>	2.430.000
<i>"Autoencoder"</i>	535.000
<i>"User Satisfaction"</i>	3.800.000
<i>"User Satisfaction and Performance Metrics"</i>	1.330.000

Fonte: Autora

Quadro 10 – Principais Artigos que Orientam a Pesquisa

Artigos Selecionados	Referência
<i>"Guide to Recommendation System"</i>	(PAJORSKA, 2023)
<i>"What are Recommendation Systems?"</i>	(PHAM, 2019)
<i>"Recommender Systems: Issues, Challenges, and Research Opportunities"</i>	(KHUSRO; ALI; ULLAH, 2016)
<i>"Hybrid recommender systems: Survey and experiments"</i>	(BURKE, 2002)
<i>"Survey on collaborative filtering, content-based filtering and hybrid recommendation system"</i>	(THORAT; GOUDAR; BARVE, 2015)
<i>"A Review for Recommender System Models and Deep Learning"</i>	(SISI et al., 2020)
<i>"Variables Affecting Information Technology End-User Satisfaction: A Meta-Analysis of the Empirical Literature"</i>	(MAHMOOD et al., 2000)
<i>"Measuring IS user satisfaction: review and implications"</i>	(ZVIRAN; ERLICH, 2003)

Fonte: Autora

trabalho. Sendo assim, algumas outras atividades do Scrum, como *Dailys*, as quais são mais voltadas para um time, perdiam o sentido de serem realizadas. De toda forma, foi acompanhado o progresso por *sprint* através do *Backlog*. Seguem outras observações sobre os eventos selecionados:

- **Sprint:** As *sprints* são uma série de iterações com um tempo definido. No caso desse trabalho, optou-se por *sprints* de duas semanas, nas quais foram realizadas atividades que precisaram ser planejadas, feitas e entregues dentro do ciclo. Essas atividades incrementaram o produto final de alguma forma. Assim, a cada iteração, era gerado um novo produto de valor;
- **Reuniões de Planejamento:** As reuniões de planejamento tiveram como objetivo definir quais e como as atividades deveriam ser realizadas na *sprint*. Para isso, propôs-se um *backlog* para a parte de desenvolvimento. Além disso, foram mapeadas quais atividades deveriam ser realizadas durante cada *sprint*, e
- **Restrospectiva:** Reuniões para passar pelas atividades realizadas na semana e seus pontos altos e baixos. Foram realizadas com a orientadora e co-orientador.

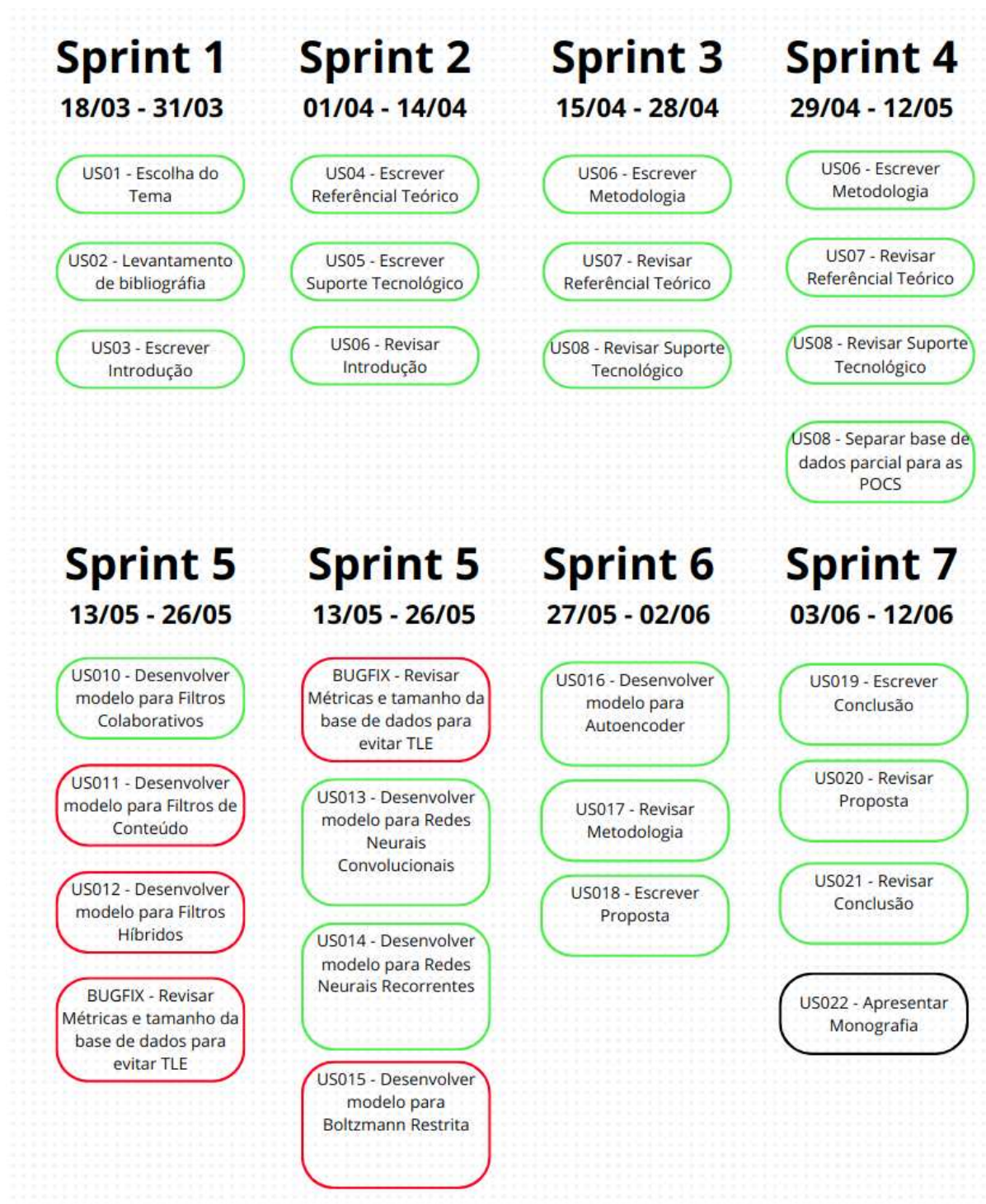
Para a primeira etapa do Trabalho, o *backlog* está definido na Figura 12, já mapeado por *sprints*. Algumas atividades foram previstas para o *backlog* da primeira etapa do trabalho. Entretanto, devido a complicações no desenvolvimento, como será abordado na seção [Solução Inicial para o Sistema de Recomendação](#) (Cap. 5, seção 2), essas atividades não foram finalizadas, sendo necessário utilizar-se de outras alternativas. Para a segunda etapa do Trabalho, o *backlog* está definido na Figura 13. Assim como ocorreu na primeira etapa, devido a complicações no desenvolvimento, algumas atividades tiveram pontos de atenção, mais detalhados na seção [API](#) (Cap. 5, seção 3.2).

4.4 Método de Coleta e Análise dos Resultados

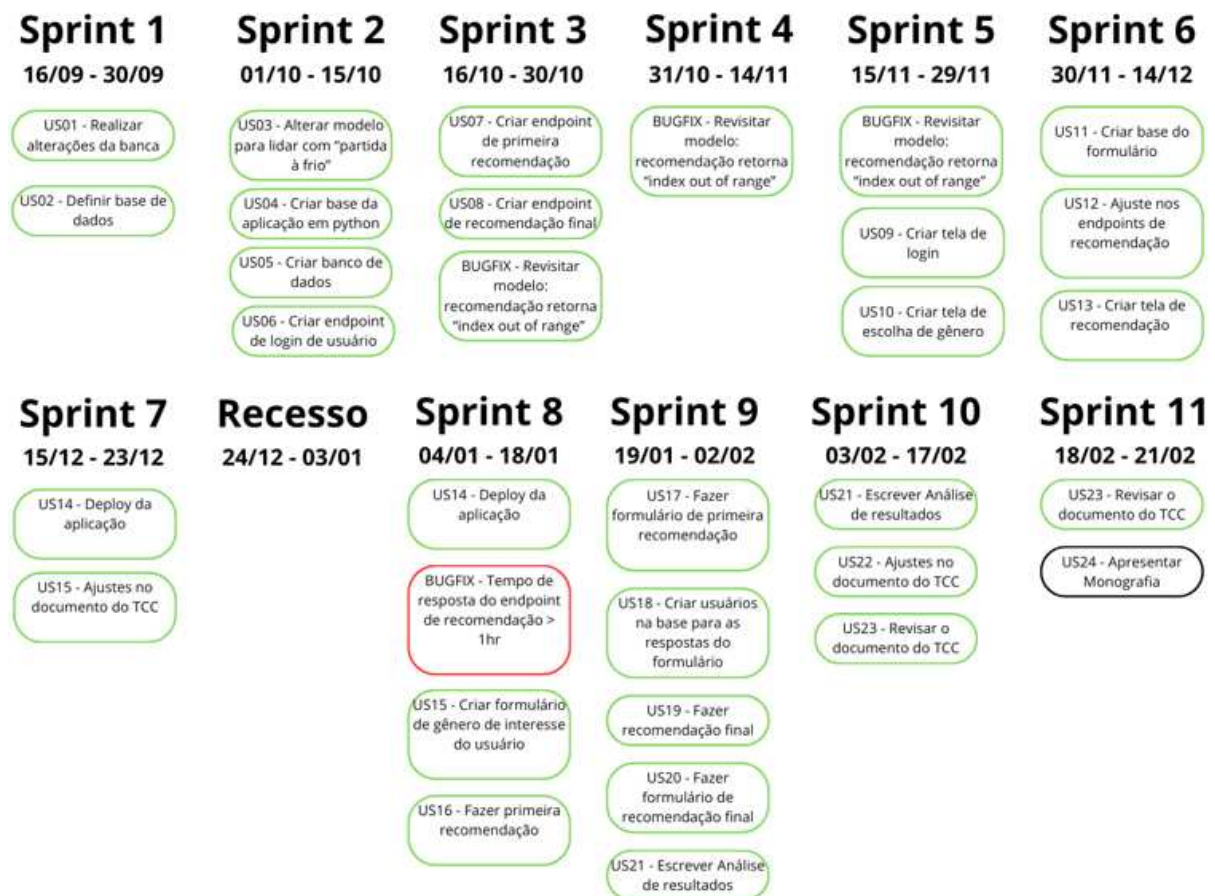
Como [Engel \(2000\)](#) conceitua, a pesquisa-ação procura unir a pesquisa à ação ou prática, desenvolvendo o conhecimento como parte da prática. No contexto desse trabalho, ocorreu uma pesquisa realizada em Sistemas de Recomendação com aplicação de Inteligência Artificial. Já como prática, ocorreu a elaboração de um modelo de Sistema de Recomendação com Inteligência Artificial. Em termos efetivos, os resultados foram avaliados com base na satisfação do usuário, definida orientando-se por questionários. A elaboração desses questionários é apresentada a seguir, com base nas métricas de avaliação de desempenho do modelo, também definidas mais adiante.

Como abordado na seção de [Satisfação do Usuário](#) (Cap. 2, seção 3), para a coleta de dados de satisfação do usuário, foi utilizado o modelo de [Mahmood et al. \(2000\)](#), com

Figura 12 – Backlog - Primeira Etapa do Trabalho



Fonte: Autora
 Legenda: Verde - Concluído; Vermelho - Não Concluído, e Preto - A Concluir

Figura 13 – *Backlog* - Segunda Etapa do Trabalho

Fonte: Autora Legenda: Verde - Concluído; Vermelho - Não Concluído, e Preto - A Concluir

foco em: benefícios e conveniência; antecedentes do usuário, e apoio e estímulo organizacional. Para elaboração do questionário, foram utilizadas questões em escalas (0 a 5), e questões subjetivas de opinião geral da aplicação pela visão do usuário.

As perguntas foram elaboradas considerando a base de dados escolhida, pois seus resultados precisavam ser usados para treinar novamente o modelo e gerar uma versão mais precisa. O mesmo questionário foi aplicado após o modelo re-treinado ser aplicado no sistema. Assim, foi analisado via as respostas, caso ocorressem uma melhora no Sistema de Recomendação.

Ainda foi utilizado um questionário adicional para coletar os gêneros de filmes que os usuários tinham interesse, visando contornar o problema encontrado na aplicação, detalhado na seção [API](#) (Cap. 5, seção 3.2).

Além da análise com base na satisfação do usuário, optou-se por métricas objetivas para avaliar o desempenho dos modelos, sendo elas:

- **Precisão:** é uma métrica que mede a proporção de itens recomendados que são relevantes para o usuário. Uma precisão alta indica que a maioria das recomendações

feitas pelo modelo são relevantes. Trata-se de uma métrica dada pela equação (1):

$$Precisão = \frac{VP}{VP + FP} \quad (1)$$

Onde:

- VP é o número de verdadeiros positivos, e
 - FP é o número de falsos positivos.
- Revocação: é uma métrica que mede a proporção de itens relevantes que foram corretamente recomendados pelo sistema. Uma revocação alta indica que o modelo é capaz de identificar a maioria dos itens relevantes. Trata-se de uma métrica dada pela equação (2):

$$Revocação = \frac{VP}{VP + FN} \quad (2)$$

Onde:

- VP é o número de verdadeiros positivos, e
 - FN é o número de falsos negativos.
- Erro Quadrático Médio da Raiz (EQMR): ou *Root Mean Square Error (RMSE)*, é uma métrica voltada para avaliar a precisão do modelo. Essa métrica calcula a raiz quadrada da média das diferenças quadradas entre os valores observados e os valores preditos. Um valor de EQMR menor indica melhores previsões, pois mostra que a média das diferenças ao quadrado entre os valores previstos e reais é baixa (WANG; LU, 2018). Trata-se de uma métrica dada pela equação (3):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (3)$$

Onde:

- y_i é o valor observado;
 - \hat{y}_i é o valor predito, e
 - n é o número de observações.
- Erro Médio Absoluto (EMA): ou *Mean Absolut Error (MAE)*, é uma métrica que também avalia a precisão do modelo, calculando a média das diferenças absolutas entre os valores observados e os valores preditos pelo modelo. Um valor de MAE menor é melhor, indicando que as previsões estão próximas dos valores reais (WANG; LU, 2018). O MAE é menos sensível a grandes erros comparado ao RMSE (CHAI; DRAXLER et al., 2014). Trata-se de uma métrica dada pela equação (4):

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (4)$$

Onde:

- y_i é o valor observado;
- \hat{y}_i é o valor predito, e
- n é o número de observações.

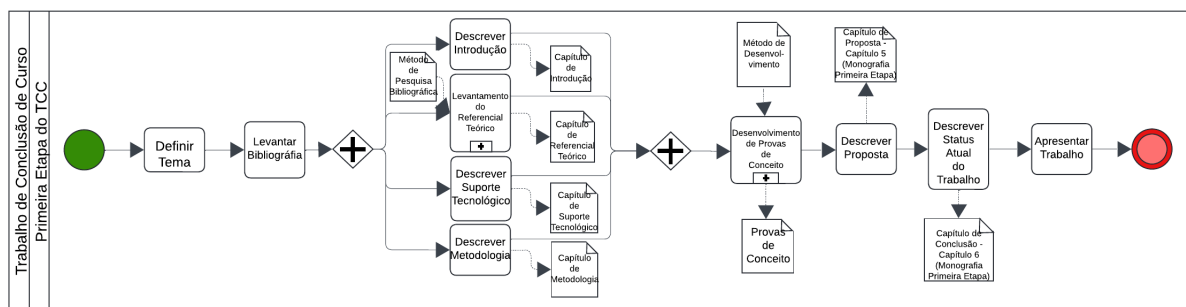
4.5 Fluxo das Atividades/Subprocessos

Essa seção é dedicada a apresentar as atividades realizadas durante a execução desse trabalho. Dividida em duas subseções, constam as atividades desenvolvidas durante a execução da Primeira Etapa do Trabalho, cujo foco foi embasar a questão de pesquisa com referenciais bibliográficos e apresentar uma proposta. Já na Segunda Etapa do Trabalho, o foco foi desenvolver a proposta apresentada na primeira etapa, cumprindo na íntegra os objetivos definidos na seção [Objetivos](#) (Cap. 1, seção 4).

4.5.1 Primeira Etapa do Trabalho

A Figura 14, a Figura 15 e a Figura 16 apresentam em formato de gráfico BPMN (*Business Process Modeling Notation*) o fluxo de atividades realizadas na Primeira Etapa do Trabalho:

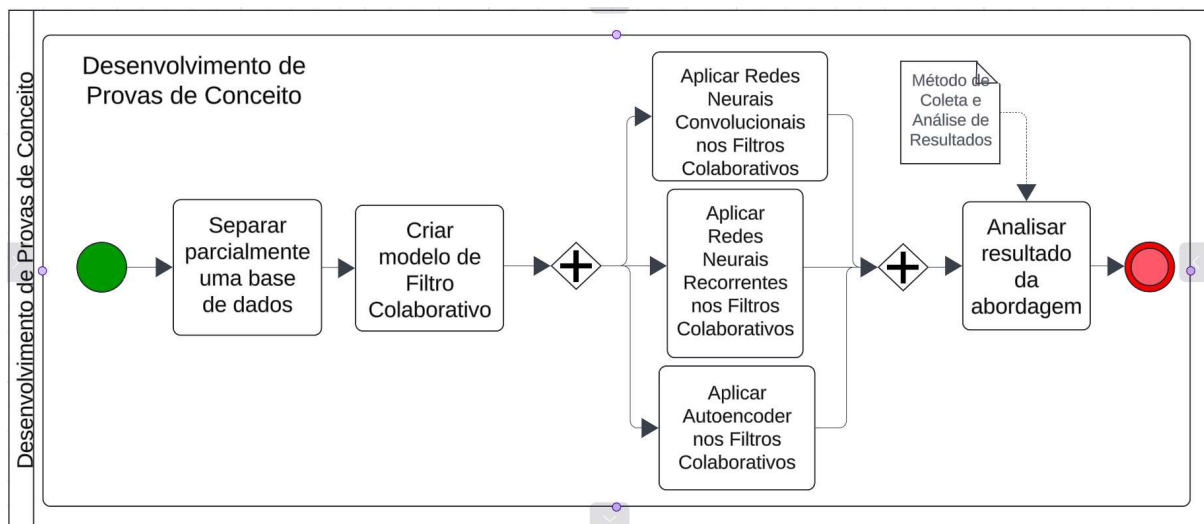
Figura 14 – Fluxo de Atividades da Primeira Etapa do Trabalho



Fonte: Autora

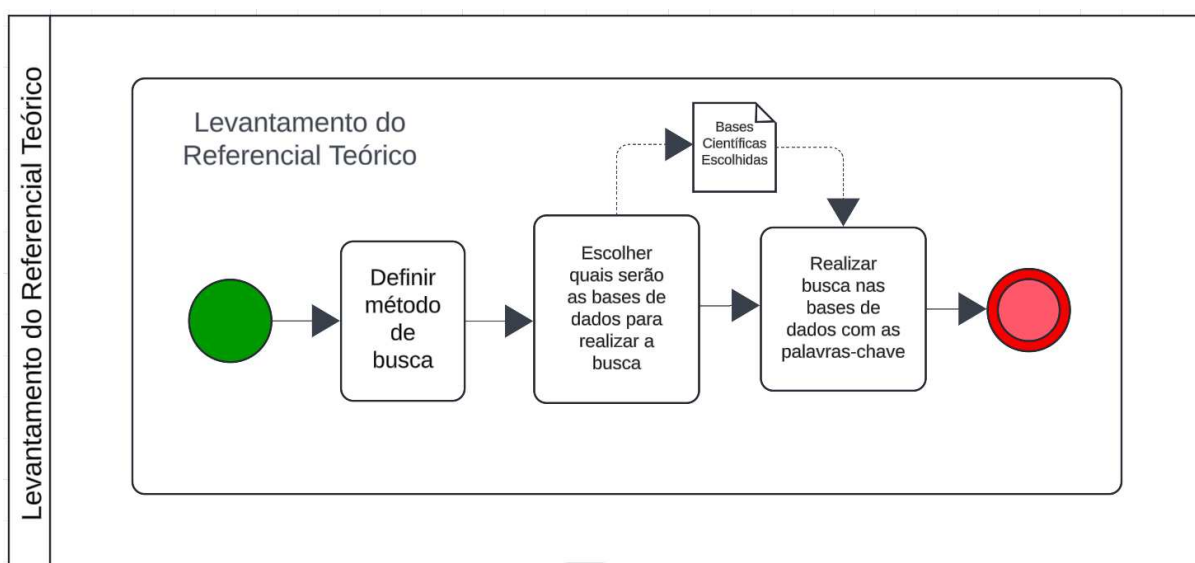
- Definir Tema: definiu-se o tema no qual o trabalho seria realizado. Status: Concluída. Resultado: "Satisfação de Usuário em um Sistema de Recomendação Centrado em Aprendizado Profundo";
- Levantar Bibliografia: levantou-se a base teórica, a qual fundamentou esse trabalho. Status: Concluída. Resultado: Autores de Referência nos tópicos de interesse desse projeto, e que constam citados ao longo dessa monografia.

Figura 15 – Subprocesso - Desenvolvimento de Prova de Conceito



Fonte: Autora

Figura 16 – Subprocesso - Levantamento do Referencial Teórico



Fonte: Autora

- Descrever Introdução: desenvolveu-se a introdução do trabalho, trazendo uma ideia geral do tema e os objetivos desse trabalho. Status: Concluída. Resultado: [Introdução](#) (Cap. 1).
- Levantamento do Referencial Teórico: abordou-se em mais detalhes as bases teóricas que auxiliaram na construção do trabalho. Status: Concluído. Resultado: [Referencial Teórico](#) (Cap. 2), orientou-se pelo [Método de Pesquisa Bibliográfica](#) (Cap. 4, seção 2);
- Descrever Suporte Tecnológico: apresentou-se sobre as tecnologias que apoiaram o desenvolvimento do trabalho Status: Concluída. Resultado: [Suporte Tecnológico](#) (Cap.3);
- Descrever Metodologia: especificou-se a metodologia e o cronograma desse trabalho. Status: Concluída. Resultado: Presente Capítulo;
- Desenvolvimento de Provas de Conceito: iniciou-se parte do desenvolvimento do trabalho. Com o objetivo de selecionar o melhor filtro e o melhor modelo de Aprendizado Profundo para ser usado no trabalho, foram desenvolvidas provas de conceito. Isso permitiu que, com insumos mais práticos, fosse possível identificar o modelo usado no trabalho. Status: Concluído. Resultado: seção de [Sistema de Recomendação com Aprendizado Profundo](#) (Cap. 5), orientando-se pelo [Método de Desenvolvimento](#) (Cap. 4, seção 3);
 - Separar parcialmente uma base de dados: selecionar uma pequena parte da base de dados para testar os modelos;
 - Criar modelo de Filtro Colaborativo: com a base selecionada, aplicar os Filtros Colaborativos nela e treinar um modelo;
 - Aplicar Redes Neurais Convolucionais nos filtros escolhidos: usando Filtro Colaborativo, implementar Redes Neurais Convolucionais e treinar novamente o modelo;
 - Aplicar Redes Neurais Recorrentes nos filtros escolhidos: usando Filtro Colaborativo, implementar Redes Neurais Recorrentes e treinar novamente o modelo;
 - Aplicar *Autoencoder* nos filtros escolhidos: usando Filtro Colaborativo, implementar *Autoencoder* e treinar novamente o modelo, e
 - Analisar resultado da abordagem: analisar os resultados da aplicação dos métodos de Aprendizado Profundo no modelo, e destacar usando métricas (precisão, revocação, erro médio absoluto e erro quádratico médio da raiz).
- Descrever Proposta: descreveu-se a proposta desse trabalho, visando responder as [Questões de Pesquisa e Desenvolvimento](#) (Cap. 1, seção 3). Status: Concluída. Resultado: Capítulo 5 - Proposta (Monografia da Primeira Etapa do Trabalho). Vários

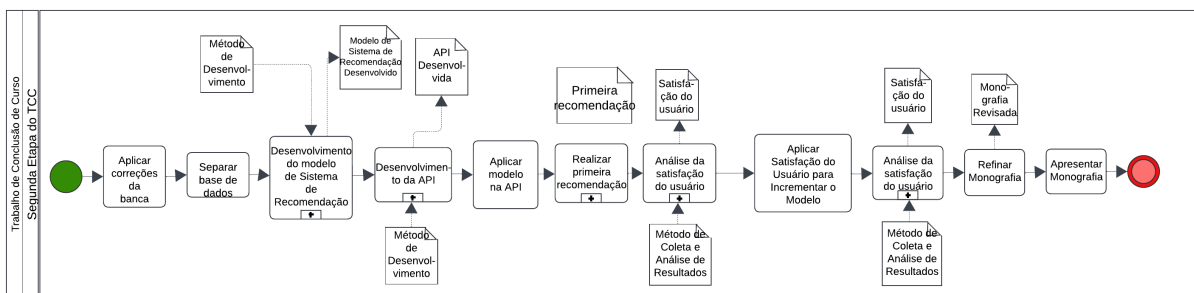
insumos constam no Capítulo 5 - Sistema de Recomendação com Aprendizado Profundo, na presente monografia.

- Descrever Status Atual do Trabalho: focou-se nas conclusões iniciais, com base nas provas de conceito desenvolvidas e na pesquisa realizada ao longo da primeira etapa do Trabalho. Status: Concluída. Capítulo 6 - Status Atual do Trabalho (Monografia da Primeira Etapa do Trabalho). Vários insumos constam no Capítulo 7 - Conclusão, na presente monografia, e
- Apresentar Trabalho: apresentou-se o trabalho para à banca. Status: Concluída.

4.5.2 Segunda Etapa do Trabalho

A Figura 17, a Figura 18, a Figura 19, a Figura 20, e a Figura 21 apresentam em formato de gráfico BPMN (*Business Process Modeling Notation*) o fluxo de atividades previstas para a segunda etapa do Trabalho:

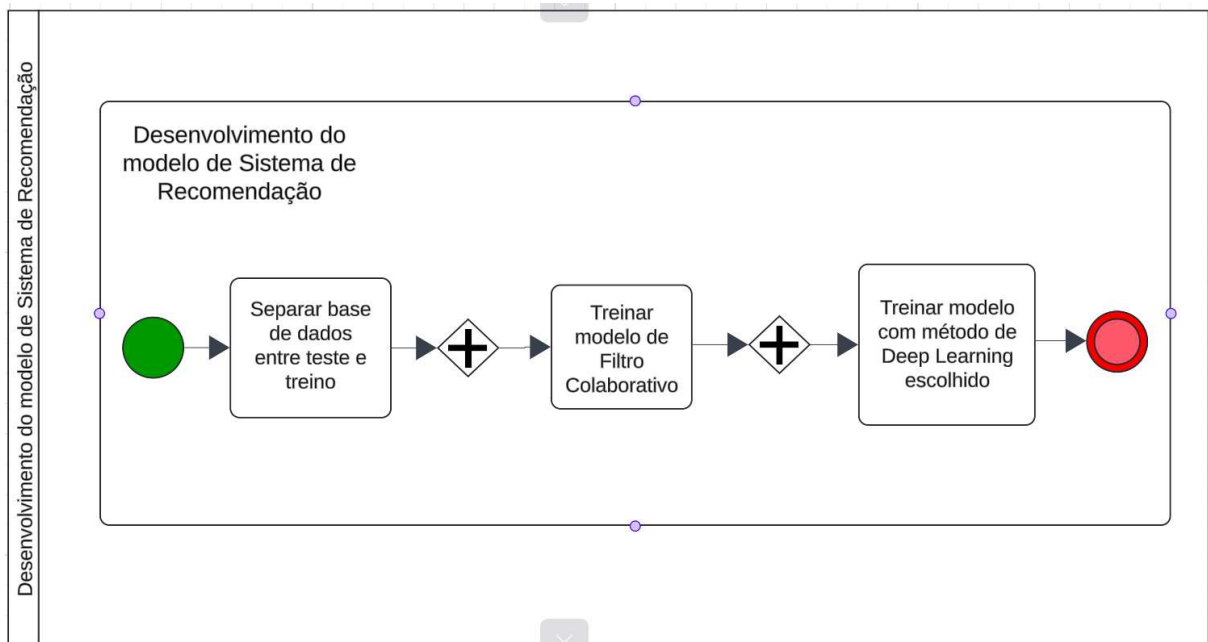
Figura 17 – Fluxo de Atividades da Segunda Etapa do Trabalho



Fonte: Autora

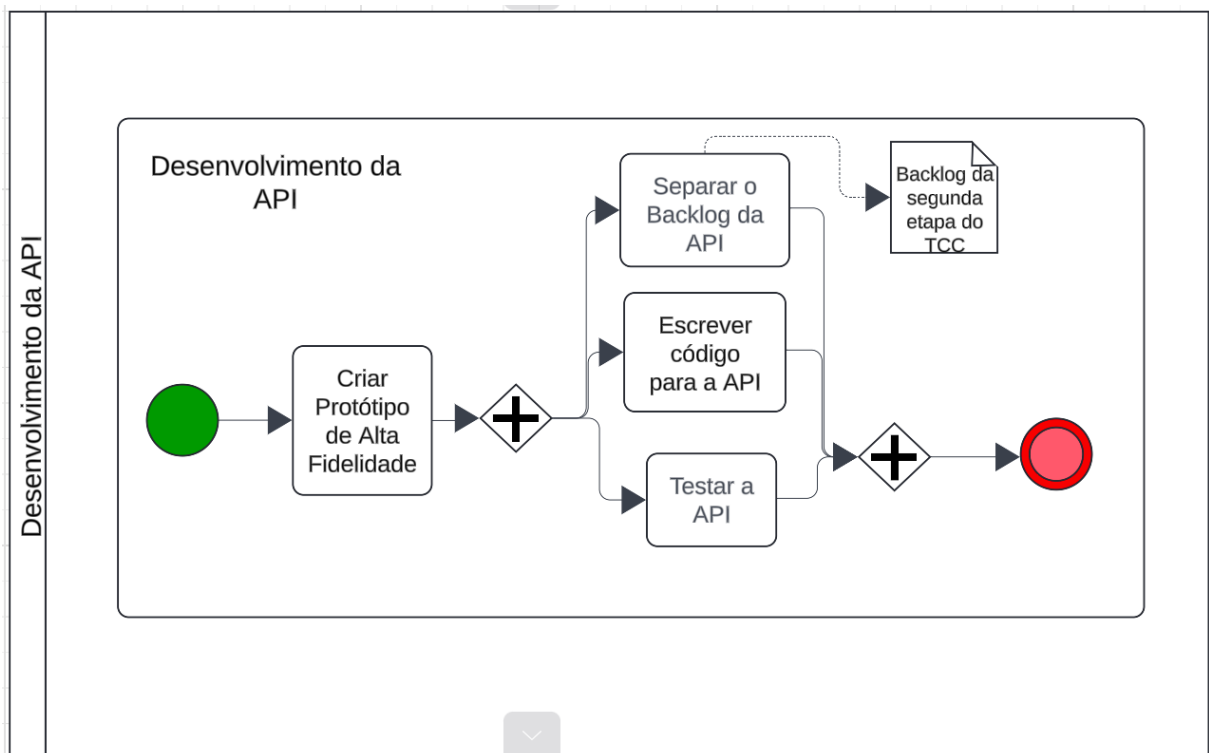
1. Aplicar Correções da Banca: aplicou-se, ainda na primeira parte do trabalho, as correções sugeridas pela banca. Status: Concluída;
2. Separar Base de Dados: escolheu-se e manipulou-se a base de dados usada nesse trabalho. Status: Concluída. Resultado: [Filtros Colaborativos e Aprendizado Profundo](#) (Cap. 5, seção 3.1);
3. Desenvolvimento do Modelo de Sistema de Recomendação: desenvolveu-se o modelo de Sistema de Recomendação, com base nas Provas de Conceito realizadas na primeira parte do trabalho, bem como orientou-se pelo [Método de Desenvolvimento](#) (Cap. 4, seção 3). Status: Concluída. Resultado: [Filtros Colaborativos e Aprendizado Profundo](#) (Cap. 5, seção 3.1);
4. Desenvolvimento da API: desenvolveu-se a API que permitiu interagir com o usuário e apresentar o Sistema de Recomendação desenvolvido, com maior foco no *frontend*

Figura 18 – Subprocesso - Modelo de Sistema de Recomendação



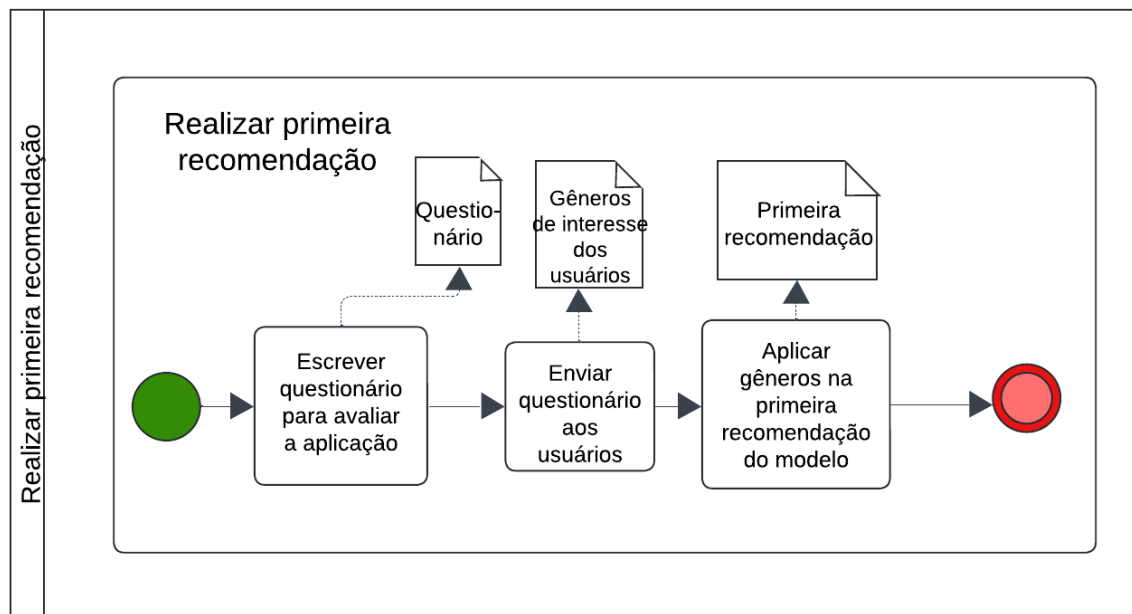
Fonte: Autora

Figura 19 – Subprocesso - Desenvolvimento da API



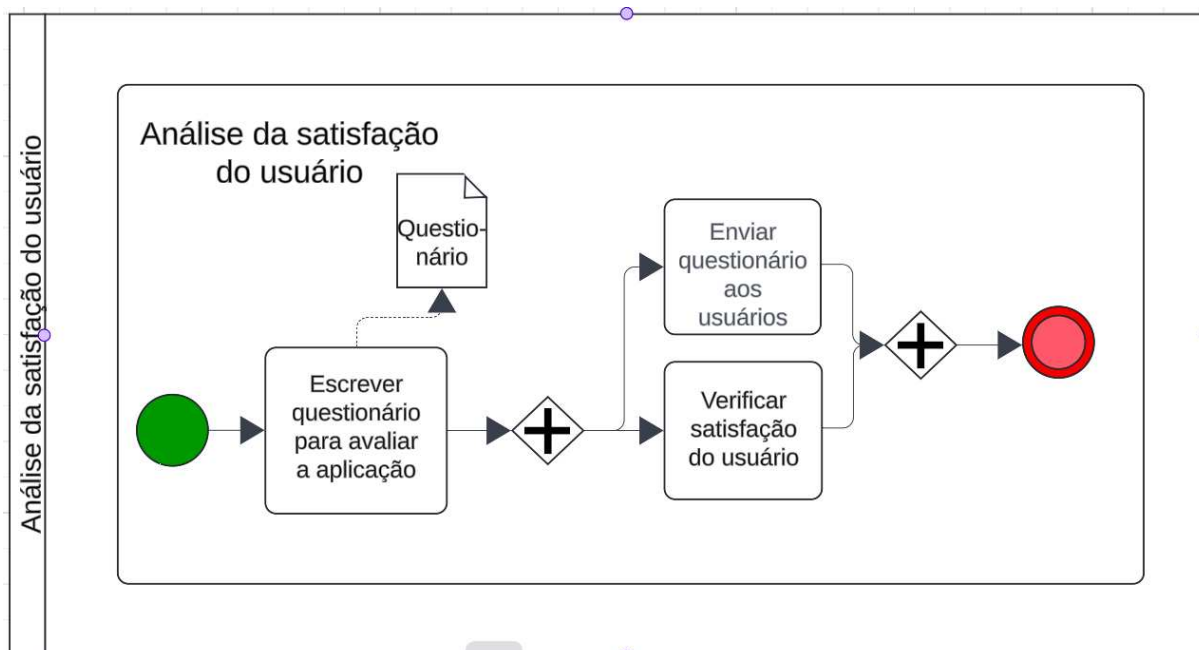
Fonte: Autora

Figura 20 – Subprocesso - Coleta de Dados do Usuário



Fonte: Autora

Figura 21 – Subprocesso - Análise da Satisfação do Usuário



Fonte: Autora

da aplicação. Por ser um subprocesso que envolveu desenvolvimento, foi guiado pelo [Método de Desenvolvimento](#) (Cap. 4, seção 3) estabelecido. Status: Concluída. Resultado: [API](#) (Cap. 5, seção 3.2);

5. Aplicar Modelo na API: aplicou-se o modelo de Sistema de Recomendação na aplicação desenvolvida. Status: Concluída. Resultado: [API](#) (Cap. 5, seção 3.2);
6. Análise da Satisfação do Usuário: capturou-se, através de formulários, a experiência do usuário com o Sistema de Recomendação desenvolvido. Orientou-se pelo [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4). Status: Concluída. Resultado: [Formulários](#) (Cap. 6, seção 1);
7. Aplicar Satisfação do Usuário para Incrementar o Modelo: separou-se os dados que poderiam ser aproveitados para incrementar o modelo, aplicando-os em um novo treinamento do modelo. Status: Concluída. Resultado: [Filtros Colaborativos e Aprendizado Profundo](#) (Cap. 5, seção 3.1) e [Formulários](#) (Cap. 6, seção 1);
8. Análise dos Novos Resultados: novamente, coletou-se a satisfação dos usuários e analisou-se a precisão do Sistema de Recomendação. Por ser um subprocesso que envolveu análise, orientou-se pelo [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4). Status: Concluída. Resultado: [Formulários](#) (Cap. 6, seção 1);
9. Refinar Monografia: revisou-se a parte escrita do trabalho, finalizando sua redação. Status: Concluída, e
10. Apresentar Trabalho: será apresentado o trabalho final à banca. Status: A Ser Realizada.

4.6 Cronogramas

O Quadro [11](#) apresenta os meses previstos para cada atividade e/ou subprocesso da primeira etapa do Trabalho.

O Quadro [12](#) apresenta os meses previstos para cada atividade e/ou subprocesso da segunda etapa do Trabalho.

4.7 Resumo do Capítulo

Neste capítulo, foi apresentada a Classificação de Pesquisa, quanto à Natureza (Pesquisa Aplicada); Abordagem (Pesquisa Híbrida, Quali e Quantitativa); Objetivos (Pesquisa Exploratória), e Procedimentos (Pesquisa Bibliográfica/ Pesquisa-ação). Adicionalmente, foram esclarecidos os métodos: Pesquisa Bibliográfica, para formação de

Quadro 11 – Cronograma de Atividades/Subprocessos da Primeira Etapa do TCC

Atividades	Março	Abril	Maio	Junho	Julho
Definir Tema	X				
Levantar Bibliografia	X				
Descrever Introdução	X				
Levantamento do Referencial Teórico		X			
Descrever Suporte Tecnológico		X			
Descrever Metodologia			X		
Desenvolvimento Prova de Conceito			X	X	
Descrever Proposta			X	X	
Descrever Status Atual do Trabalho				X	
Apresentar Monografia					X

Fonte: Autora

Quadro 12 – Cronograma de Atividades/Subprocessos da Segunda Etapa do TCC

Atividades	Setembro	Outubro	Novembro	Dezembro	Janeiro	Fevereiro
Aplicar Correções da Banca	X					
Separar Base de Dados	X					
Desenvolvimento Modelo de Sistema de Recomendação		X	X			
Desenvolvimento da API		X	X	X		
Aplicar Modelo na API		X	X	X		
Analisar Resultados					X	
Usar os resultados para incrementar o modelo					X	
Reaplicar novo modelo na API					X	
Análise dos novos resultados					X	
Refinar Monografia					X	
Apresentar Monografia						X

Fonte: Autora

uma base teórica sólida para o trabalho; Método de Desenvolvimento, sendo uma combinação de Scrum, e Método de Coleta e Análise dos Resultados, usando formulários, e analisando as respostas dos usuários com base em métricas (Precisão, Revocação, Erro Médio Absoluto e Erro Quadrático Médio da Raiz). Por fim, constam os fluxos de atividades/subprocessos e cronogramas, tanto para a primeira etapa, quanto para a segunda etapa do Trabalho.

5 Sistema de Recomendação com Aprendizado Profundo

Este Capítulo apresenta em detalhes sobre o Sistema de Recomendação desenvolvido. Aborda, primeiramente, a [Contextualização](#) (seção 1), que retoma as [Questões de Pesquisa e Desenvolvimento](#) (Cap. 1, seção 3) a serem respondidas e os desafios envolvendo Sistemas de Recomendação. Em seguida, apresenta a [Solução Inicial para o Sistema de Recomendação](#) (seção 2), no contexto da primeira etapa do trabalho, que aborda sobre os possíveis filtros e métodos de Aprendizado Profundo que podem ser usados em Sistemas de Recomendação, apresentando as métricas para cada filtro e métodos usados. Na sequência, apresenta-se a [Solução Evoluída para o Sistema de Recomendação](#) (seção 3), que com base nos resultados obtidos na [Solução Inicial para o Sistema de Recomendação](#) (seção 2), apresenta como o modelo foi evoluído. Por fim, tem-se o [Resumo do Capítulo](#) (seção 4).

5.1 Contextualização

A proposta desse trabalho vêm da experiência da autora com recomendações geradas em plataformas de *streaming*, percebendo que nem sempre as recomendações eram precisas ou de seu interesse. Dessa forma, voltando para um público mais geral, levantou-se a questão de como as recomendações influenciavam os usuários dessas plataformas.

Como [Acevedo et al. \(2020\)](#) e [Lourenço e Krakauer \(2021\)](#) apresentam, os avanços tecnológicos e a pandemia do COVID-19 impulsionaram o uso de serviços de *streaming* nos últimos anos. Nesse contexto, o uso de Sistemas de Recomendação nas plataformas de *streaming*, induzem o usuário a continuar usando as plataformas de *streaming* ([HASAN; JHA; LIU, 2018](#)), contribuindo para o aumento no uso de serviços de *streaming* no cotidiano da população.

Entretanto, como apresentado em [Jain et al. \(2015\)](#), os atuais Sistemas de Recomendação enfrentam alguns desafios, como *Cold Start*, que foram debatidos na seção de [Contextualização](#) (Cap. 1, seção 1). Esses problemas provavelmente justificam as recomendações imprecisas vivenciadas pela autora. Assim, levantou-se mais uma questão de como poderia melhorar a satisfação do usuário nos Sistemas de Recomendação. Ao cursar a disciplina FGA0083 - APRENDIZADO DE MÁQUINA, no campus do Gama da Universidade de Brasília, levantou-se uma possível questão. Seria possível utilizar Inteligência Artificial e Aprendizado de Máquina para incrementar um Sistema de Recomendação?

Dessa forma, nesse trabalho, a autora propõe-se a analisar a satisfação do usuá-

rio em um Sistema de Recomendação centrado em Inteligência Artificial. Sendo assim, desenvolveu-se um Sistema de Recomendação utilizando Filtros Colaborativos e Filtro com Rede Neural Convolucional de Aprendizado Profundo, levantados nas seções [Inteligência Artificial](#) (Cap. 2, seção 2) e [Filtros Baseados em Aprendizado Profundo](#) (Cap. 2, seção 2.2). Em seguida, o Sistema de Recomendação proposto foi avaliado por métricas e por usuários, como definido em [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4).

5.2 Solução Inicial para o Sistema de Recomendação

Existem diversas aplicações de Inteligência Artificial em Sistemas de Recomendação, como levantado nas seções [Inteligência Artificial](#) (Cap. 2, seção 2) e [Filtros Baseados em Aprendizado Profundo](#) (Cap. 2, seção 2.2). A fim de escolher um modelo para aplicar em um Sistema de Recomendação, foram desenvolvidas algumas provas de conceito com os filtros de Inteligência Artificial e as abordagens em Aprendizado Profundo. Adicionalmente, fez-se uso das métricas de avaliação do modelo. Em um primeiro momento, não ocorreu a validação junto ao usuário. As métricas constam mais bem explicadas em [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4). Com isso, escolheu-se o modelo a ser usado na proposta.

Para a escolha dos Filtros, embora a literatura aponte os Filtros Híbridos, aplicado com Filtro Colaborativo e Filtro de Conteúdo, como a melhor abordagem (vide seção [Filtros Híbridos](#) (Cap. 2, seção 2.1)), o modelo de filtragem híbrida, utilizando Filtro Colaborativo e Filtro de Conteúdo, durante o seu treinamento no Colab ([GOOGLE, 2024](#)), apresentou problemas de memória no Filtro de Conteúdo, encerrando a execução antes de finalizar o treinamento. Esforços foram dispendidos pela autora para contornar essa situação. Entretanto, os testes não foram bem sucedidos. Por essa razão, o filtro utilizado foi o Filtro Colaborativo. Já para a abordagem de Aprendizado Profundo, utilizada no filtro escolhido, e orientando-se pelas provas de conceito descritas nas próximas seções, foi escolhida aquela que obteve as melhores métricas, ou seja: a abordagem de Rede Neural Convolucional. Dessa forma, obtendo-se um Filtro Híbrido orientando-se por Filtro Colaborativo e Rede Neural Convolucional.

Para a elaboração das abordagens de Aprendizado Profundo, foram realizados alguns testes com Máquina Restrita de Boltzmann. Entretanto, não foi possível chegar a um modelo treinado que tivesse resultados satisfatórios. Por isso, para a escolha dos Filtros Baseados em Aprendizado Profundo, levou-se em consideração apenas Rede Neural Convolucional, Rede Neural Recorrente, e *Autoencoder*.

Ainda no contexto da aplicação das abordagens de Aprendizado Profundo, o método utilizado foi usar o Filtro Colaborativo para refinar os dados, e na sequência aplicar as

abordagens de Aprendizado Profundo nesses dados refinados, classificando-se como uma técnica de Cascata, abordada na seção [Classificação de Filtros Híbridos](#) (Cap. 2, seção 2.1.1). Ao usar esse método, percebeu-se que, conforme [Possíveis Implementações](#) (Cap. 2, seção 2.1.2), foram implementadas características do primeiro filtro (Filtro Colaborativo) no segundo filtro (Abordagens de Aprendizado Profundo). Para todos os modelos treinados nessa seção utilizou-se como base as arquiteturas apresentadas na seção [Kaggle](#) (Cap. 3, seção 1.1), utilizando os mesmos parâmetros e valores que os autores utilizaram nos modelos de referência.

Para treinar os modelos de Sistema de Recomendação, foi escolhida a base de dados do [Parashar \(2023\)](#) de filmes, como abordado na seção do [Kaggle](#) (Cap. 3, seção 1.1). Essa base possui em sua estrutura um identificador único para usuários (*userId*) e para filmes (*movieId*); a avaliação que o usuário deu para o filme (*ratings*); o horário que ocorreu a avaliação do filme *timestamp*; o título do filme (*title*), e os gêneros do filme (*genres*). Tais dados constam demonstrados na Figura 22. Em resumo, tem-se, por exemplo, para o *userId* 1, o *movieId* 296, cujo *ratings* é 5.0. Isso ocorre para o caso do filme intitulado "*Pulp Fiction* (1994)", sendo esse um filme nos gêneros: *Comedy/Crime/Drama/Thriller*. Já para o *userId* 4, o *ratings* conferido, para esse mesmo filme, foi 4.0.

Figura 22 – Base de Dados

	userId	movieId	rating	timestamp	title	genres
0	1	296	5.0	1147880044	Pulp Fiction (1994)	Comedy Crime Drama Thriller
1	3	296	5.0	1439474476	Pulp Fiction (1994)	Comedy Crime Drama Thriller
2	4	296	4.0	1573938898	Pulp Fiction (1994)	Comedy Crime Drama Thriller
3	5	296	4.0	830786155	Pulp Fiction (1994)	Comedy Crime Drama Thriller
4	7	296	4.0	835444730	Pulp Fiction (1994)	Comedy Crime Drama Thriller
5	8	296	5.0	890489713	Pulp Fiction (1994)	Comedy Crime Drama Thriller
6	10	296	4.5	1227571308	Pulp Fiction (1994)	Comedy Crime Drama Thriller
7	12	296	5.0	1119354604	Pulp Fiction (1994)	Comedy Crime Drama Thriller
8	13	296	5.0	1238029599	Pulp Fiction (1994)	Comedy Crime Drama Thriller
9	14	296	5.0	1506208897	Pulp Fiction (1994)	Comedy Crime Drama Thriller

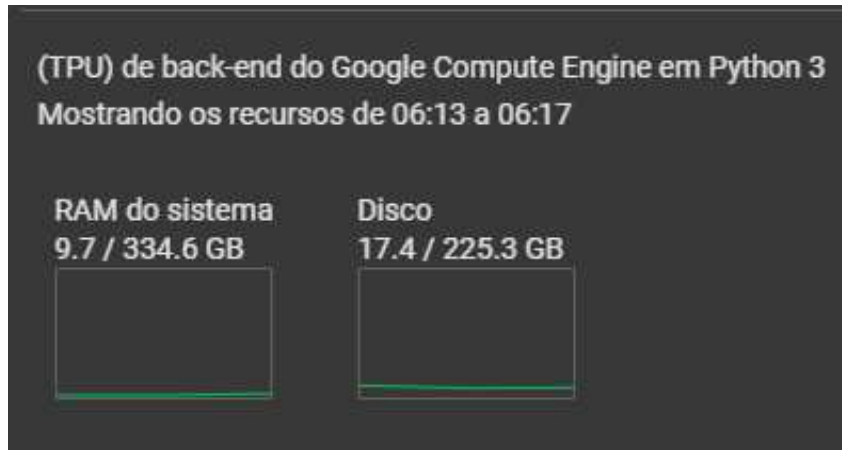
Fonte: Autora

5.2.1 Modelo com Filtros Colaborativos

Antes de treinar o modelo com as abordagens de Aprendizado Profundo, no contexto desse trabalho, é preciso ter uma base com um filtro de Inteligência Artificial que se aplica a Sistemas de Recomendação. Para isso, foi escolhido o Filtro Colaborativo, por consumir menos memória durante a execução. A Figura 23 apresenta o consumo de memória do Filtro Colaborativo. Em comparação com o Filtro por Conteúdo que não che-

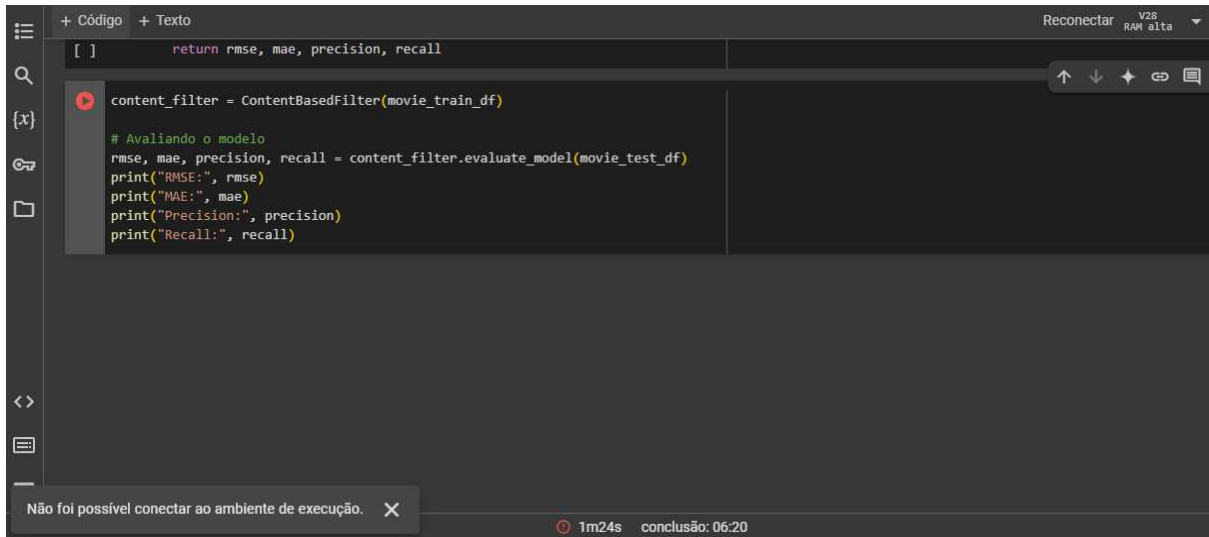
gou a finalizar a execução, desconectando do ambiente após um pico no uso de memória, apresentado nas Figuras 24 e 25.

Figura 23 – Consumo de Memória - Filtro Colaborativo



Fonte: Autora

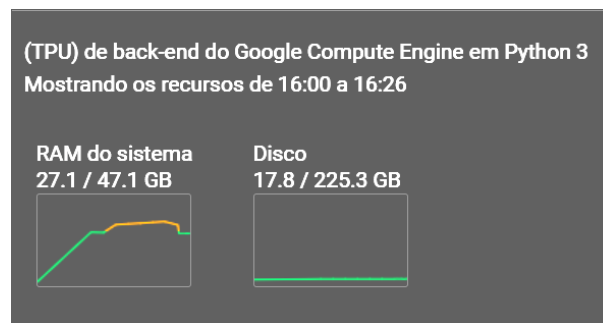
Figura 24 – Falha na Conexão - Filtro de Conteúdo



Fonte: Autora

Utilizando a linguagem python, foi criada uma classe para os Filtros Colaborativos, separada em três funções principais (funções que executam todas as lógicas principais do modelo, sendo: treinar, avaliar e recomendar) e três funções auxiliares (funções que, para quesito de organização, foram separadas do código principal, mas apresentam alguma lógica importante na elaboração do código e, por isso, merecem destaque). As principais funções constam a seguir apresentadas:

Figura 25 – Consumo de Memória - Filtro de Conteúdo



Fonte: Autora

- Preparar Dados: Função auxiliar, responsável por carregar os dados da base de dados e separar as colunas a serem utilizadas. Na Figura 26, é apresentado um trecho de código dessa função auxiliar;

Figura 26 – Preparar Dados - Filtro Colaborativo

```
def prepare_data(self):  
    # Definir o esquema dos dados de entrada  
    reader = Reader(rating_scale=(self.train_data['rating'].min(), self.train_data['rating'].max()))  
    data = Dataset.load_from_df(self.train_data[['userId', 'movieId', 'rating']], reader)  
    return data
```

Fonte: Autora

- Treinar Modelo: Função principal, responsável por treinar o modelo. Na Figura 27, é apresentado um trecho de código dessa função principal;

Figura 27 – Treinar Modelo - Filtro Colaborativo

```
def train(self):  
    data = self.prepare_data()  
    trainset, testset = train_test_split(data, test_size=0.2, random_state=42)  
  
    # Treinar o algoritmo SVD  
    self.algo = SVD()  
    self.algo.fit(trainset)  
  
    # Avaliar o modelo  
    self.evaluate(testset)
```

Fonte: Autora

- Avaliar Modelo: Função principal, responsável por avaliar o modelo com base nas métricas definidas na seção [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4) [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4) (Precisão, Revocação, Erro Médio Absoluto e Erro Quadrático Médio da Raiz). Na Figura 28, é apresentado um trecho de código dessa função principal;

Figura 28 – Avaliar Modelo - Filtro Colaborativo

```
def evaluate(self, testset):
    # Avaliar o modelo no conjunto de teste
    predictions = self.algo.test(testset)

    # RMSE e MAE
    rmse = accuracy.rmse(predictions)
    mae = accuracy.mae(predictions)

    # Precision e Recall
    precisions, recalls = self.precision_recall_at_k(predictions, k=5, threshold=3.5)

    # Média de precisão e recall
    mean_precision = sum(prec for prec in precisions.values()) / len(precisions)
    mean_recall = sum(rec for rec in recalls.values()) / len(recalls)

    print(f'EQMR: {rmse}')
    print(f'EMA: {mae}')
    print(f'Precisão: {mean_precision}')
    print(f'Revocação: {mean_recall}')
```

Fonte: Autora

- Calcular Precisão e Revocação: Função auxiliar, responsável por calcular a Precisão e a Revocação do modelo. Na Figura 29, é apresentado um trecho de código dessa função auxiliar;

Figura 29 – Avaliar Modelo - Filtro Colaborativo

```
def precision_recall_at_k(self, predictions, k=10, threshold=3.5):
    # Função para calcular precisão e recall em k
    user_est_true = defaultdict(list)
    for uid, _, true_r, est, _ in predictions:
        user_est_true[uid].append((est, true_r))

    precisions = {}
    recalls = {}

    for uid, user_ratings in user_est_true.items():
        # Ordenar as previsões do usuário
        user_ratings.sort(key=lambda x: x[0], reverse=True)
        # Considerar as top-k previsões
        n_rel = sum((true_r >= threshold) for (_, true_r) in user_ratings)
        n_rec_k = sum((est >= threshold) for (est, _) in user_ratings[:k])
        n_rel_and_rec_k = sum(((true_r >= threshold) and (est >= threshold)) for (est, true_r) in user_ratings[:k])

        precisions[uid] = n_rel_and_rec_k / n_rec_k if n_rec_k != 0 else 0
        recalls[uid] = n_rel_and_rec_k / n_rel if n_rel != 0 else 0

    return precisions, recalls
```

Fonte: Autora

- Prever Avaliação: Função auxiliar, responsável por prever a avaliação do usuário para os filmes. Na Figura 30, é apresentado um trecho de código dessa função auxiliar, e
- Obter Recomendação: Função principal, responsável por recomendar ao usuário um filme. Na Figura 31, é apresentado um trecho de código dessa função principal.

Figura 30 – Prever Avaliação - Filtro Colaborativo

```
def predict(self, userId, movieId):  
    # Prever a avaliação de um usuário para um filme específico  
    return self.algo.predict(userId, movieId).est
```

Fonte: Autora

Figura 31 – Obter Recomendação - Filtro Colaborativo

```
def recommend(self, userId, n=10):  
    # Obter recomendações para um usuário  
    user_movies = self.train_data[self.train_data['userId'] == userId]['movieId'].tolist()  
    all_movies = self.train_data['movieId'].unique()  
    predictions = [(movieId, self.predict(userId, movieId)) for movieId in all_movies if movieId not in user_movies]  
    predictions.sort(key=lambda x: x[1], reverse=True)  
    return predictions[:n]
```

Fonte: Autora

Com essa classe definida, foram realizados alguns testes alterando o tamanho da base de dados, para entender as diferenças que uma base bem definida tem no modelo. Foram obtidos os seguintes resultados:

- Utilizando 5% da Base, conforme mostrado na Figura 32. Percebe-se que os valores se mantêm abaixo de 1 e acima de 0, como esperado para essas métricas. Porém, o valor do Erro Quadrático Médio da Raiz está próximo de 1, e para essa métrica significa que as previsões do modelo não estão boas. Já para as outras métricas, é desejado um Erro Médio absoluto menor, mas o valor atual, próximo de 0,7 é aceitável. Assim como, espera-se que a Precisão e a Revocação sejam o mais próximas de 1, já que indicam a relevância das recomendações. Entretanto, Revocação e Precisão próximas de 0.6, para uma base bem reduzida, pode-se considerar um bom resultado.

Figura 32 – 5% da Base - Filtro Colaborativo

```
EQMR: 0.9096804172934235  
EMA: 0.6991094071694247  
Precisão: 0.6115944497300353  
Revocação: 0.5887105374729223
```

Fonte: Autora

- Utilizando 40% da Base, conforme mostrado na Figura 33. Percebe-se que o Erro Quadrático Médio da Raiz e o Erro Médio absoluto diminuíram, indicando uma melhora nas previsões. Porém, ainda é um valor considerado alto para o Erro Quadrático Médio da Raiz, visto que continua próximo de 1. Já para a Precisão, houve um aumento de 0,12, indicando uma melhora na relevância das recomendações do

modelo. Todavia, a Revocação teve uma redução, embora não muito significativa, em seu valor, apresentando um comportamento anormal, considerando que a Precisão está alta.

Figura 33 – 40% da Base - Filtro Colaborativo

```
EQMR: 0.8329121546363293
EMA: 0.632719620393796
Precisão: 0.7375318877553673
Revocação: 0.5502866333979213
```

Fonte: Autora

- Utilizando 100% da Base, conforme mostrado na Figura 34. Percebe-se que, novamente, houve uma melhora na Precisão, no Erro Quadrático Médio da Raiz e no Erro Médio absoluto. Portanto, a Precisão e o Erro Médio absoluto apresentaram resultados satisfatórios para um modelo apenas com Filtros Colaborativos. Entretanto, a Revocação diminuiu mais, ainda apresentando comportamento anormal.

Figura 34 – 100% da Base - Filtro Colaborativo

```
cf_recommender = CollaborativeFilteringRecommender(movie_dataframe)
cf_recommender.train()
recommendations = cf_recommender.recommend(userId=1, n=10)
print(recommendations)

RMSE: 0.7777
MAE: 0.5869
EQMR: 0.777675454662334
EMA: 0.5869260887527686
Precisão: 0.8231424852547558
Revocação: 0.42404595409565854
```

Fonte: Autora

Pode-se observar que, ao aumentar a base, os parâmetros de Erro Médio Absoluto e Erro Quadrático Médio da Raiz diminuem, e o parâmetro de Precisão aumenta. Isso indica que quanto maior a quantidade de dados disponíveis para treinar o modelo, melhor será seu desempenho. Porém, cabe colocar que deve ser analisado o comportamento anormal da Revocação.

5.2.2 Modelo com Redes Neurais Convolucionais

Usando como base o modelo de Filtros Colaborativos, por questão de memória do servidor do Colab, optou-se pela versão que utiliza 5% da base de dados.

Similar ao que foi feito para os Filtros Colaborativos, foi criada uma classe de Redes Neurais Convolucionais, com as mesmas três funções principais (treinar, avaliar e recomendar), porém com apenas duas funções auxiliares, sendo elas:

- Preparar Dados: Função auxiliar, responsável por carregar os dados da base de dados e separar as colunas a serem utilizadas. Na Figura 35, é apresentado um trecho de código dessa função auxiliar;

Figura 35 – Preparar Dados - Rede Neural Convolucional

```
def prepare_data(self):
    # Prepare os dados usando o modelo de filtro colaborativo
    data = self.cf_model.prepare_data()
    trainset, testset = train_test_split(data, test_size=0.2, random_state=42)

    user_ids = self.cf_model.train_data['userId'].unique()
    movie_ids = self.cf_model.train_data['movieId'].unique()

    n_users = len(user_ids)
    n_movies = len(movie_ids)

    self.user_map = {user_id: idx for idx, user_id in enumerate(user_ids)}
    self.movie_map = {movie_id: idx for idx, movie_id in enumerate(movie_ids)}

    X_train = []
    y_train = []

    for _, row in self.cf_model.train_data.iterrows():
        user_idx = self.user_map[row['userId']]
        movie_idx = self.movie_map[row['movieId']]
        rating = row['rating']
        X_train.append([user_idx, movie_idx])
        y_train.append(rating)

    X_train = np.array(X_train)
    y_train = np.array(y_train)

    X_test = []
    y_test = []

    for (uid, mid, true_r) in testset:
        if uid in self.user_map and mid in self.movie_map:
            user_idx = self.user_map[uid]
            movie_idx = self.movie_map[mid]
            X_test.append([user_idx, movie_idx])
            y_test.append(true_r)

    X_test = np.array(X_test)
    y_test = np.array(y_test)

    return X_train, y_train, X_test, y_test
```

- Treinar Modelo: Função principal, responsável por treinar o modelo. Na Figura 36, é apresentado um trecho de código dessa função principal;

Figura 36 – Treinar Modelo - Rede Neural Convolutacional

```
def train(self):
    X_train, y_train, X_test, y_test = self.prepare_data()

    # Construir o modelo CNN
    n_users = X_train[:, 0].max() + 1
    n_movies = X_train[:, 1].max() + 1

    self.cnn_model = Sequential()
    self.cnn_model.add(Embedding(input_dim=n_users, output_dim=50, input_length=2))
    self.cnn_model.add(ZeroPadding1D(padding=1))
    self.cnn_model.add(Conv1D(32, 3, activation='relu'))
    self.cnn_model.add(MaxPooling1D(2))
    self.cnn_model.add(Flatten())
    self.cnn_model.add(Dense(64, activation='relu'))
    self.cnn_model.add(Dense(1))

    self.cnn_model.compile(optimizer='adam', loss='mean_squared_error')

    self.cnn_model.fit(X_train, y_train, epochs=3, batch_size=32)

    self.evaluate(X_test, y_test)
```

Fonte: Autora

- Avaliar Modelo: Função principal, responsável por avaliar o modelo com base nas métricas definidas na seção [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4) (Precisão, Revocação, Erro Médio Absoluto e Erro Quadrático Médio da Raiz). Na Figura 37, é apresentado um trecho de código dessa função principal;

Figura 37 – Avaliar Modelo - Rede Neural Convolutacional

```
def evaluate(self, X_test, y_test):
    # Fazendo previsões
    y_pred = self.cnn_model.predict(X_test)

    # Calcular RMSE e MAE
    rmse = np.sqrt(np.mean((y_test - y_pred.flatten()) ** 2))
    mae = np.mean(np.abs(y_test - y_pred.flatten()))

    print(f'EQMR: {rmse}')
    print(f'EMA: {mae}')

    # Calculando Precision e Recall
    y_pred_binary = [1 if pred >= 3.5 else 0 for pred in y_pred.flatten()]
    y_true_binary = [1 if true >= 3.5 else 0 for true in y_test]

    precision = sum([1 for yt, yp in zip(y_true_binary, y_pred_binary) if yt == 1 and yp == 1]) / sum(y_pred_binary)
    recall = sum([1 for yt, yp in zip(y_true_binary, y_pred_binary) if yt == 1 and yp == 1]) / sum(y_true_binary)

    print(f'Precisão: {precision}')
    print(f'Revocação: {recall}')
```

Fonte: Autora

- Prever Avaliação: Função auxiliar, responsável por prever a avaliação do usuário para os filmes. Na Figura 38, é apresentado um trecho de código dessa função auxiliar, e

Figura 38 – Prever Avaliação - Rede Neural Convolutacional

```
def predict(self, userId, movieId):  
    user_idx = self.user_map[userId]  
    movie_idx = self.movie_map[movieId]  
    return self.cnn_model.predict([[user_idx, movie_idx]]).flatten()[0]
```

Fonte: Autora

- Obter Recomendação: Função principal, responsável por recomendar ao usuário um filme. Na Figura 39, é apresentado um trecho de código dessa função principal.

Figura 39 – Obter Recomendação - Rede Neural Convolutacional

```
def recommend(self, userId, n=10):  
    user_idx = self.user_map[userId]  
    all_movies = self.cf_model.train_data['movieId'].unique()  
    predictions = [  
        (movie_id, self.predict(user_idx, movie_id))  
        for movie_id in all_movies  
        if movie_id not in self.cf_model.train_data[  
            self.cf_model.train_data['userId'] == userId  
        ][ 'movieId' ].tolist()  
    ]  
    predictions.sort(key=lambda x: x[1], reverse=True)  
    return predictions[:n]
```

Fonte: Autora

Os resultados do modelo treinado por Redes Neurais Convolutacionais são ilustrados na Figura 40. Percebe-se que a Precisão e a Revocação estão bem próximas de 1, sendo assim, um valor alto e satisfatório para um Sistema de Recomendação. Também percebe-se, que apesar de ainda alto, o Erro Quadrático Médio da Raiz está bem mais baixo, se comparado ao modelo que usa apenas Filtros Colaborativos. Já o Erro Médio Absoluto está bem próximo de 0,5, sendo também um valor satisfatório. Ao comparar o resultado do modelo de Rede Neural Convolutacional (aplicado no modelo de Filtro Colaborativo) com o modelo de apenas Filtro Colaborativo, percebe-se uma melhora significativa.

5.2.3 Modelo com Redes Neurais Recorrentes

Usando como base o modelo de Filtros Colaborativos, por questão de memória do servidor do Colab, optou-se pela versão que utiliza 5% da base de dados.

Figura 40 – Resultados - Rede Neural Convolutacional

```
EQMR: 0.7463394946731559
EMA: 0.5449784967545062
Precisão: 0.8727521123832435
Revocação: 0.7689600355878871
```

Fonte: Autora

Como nos outros modelos, no modelo com Redes Neurais Recorrentes, tem-se cinco funções, sendo três principais (treinar, avaliar e recomendar) e duas auxiliares (preparar base de dados e prever). Seguem detalhes das funções.

- Preparar Dados: Função auxiliar, responsável por carregar os dados da base de dados. Na Figura 41, é apresentado um trecho de código dessa função auxiliar;

Figura 41 – Preparar Dados - Rede Neural Recorrente

```
def prepare_data(self):
    # Prepare os dados usando o modelo de filtro colaborativo
    data = self.cf_model.prepare_data()
    trainset, testset = train_test_split(data, test_size=0.2, random_state=42)
    return trainset, testset
```

Fonte: Autora

- Treinar Modelo: Função principal, responsável por treinar o modelo. Na Figura 42, é apresentado um trecho de código dessa função principal;
- Avaliar Modelo: Função principal, responsável por avaliar o modelo com base nas métricas definidas na seção [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4) (Precisão, Revocação, Erro Médio Absoluto e Erro Quadrático Médio da Raiz). Na Figura 43, é apresentado um trecho de código dessa função principal;
- Prever Avaliação: Função auxiliar, responsável por prever a avaliação do usuário para os filmes. Na Figura 44, é apresentado um trecho de código dessa função auxiliar, e
- Obter Recomendação: Função principal, responsável por recomendar ao usuário um filme. Na Figura 45, é apresentado um trecho de código dessa função principal.

Os resultados do modelo treinado por Redes Neurais Recorrentes são ilustrados na Figura 46:

Com esses resultados, percebe-se novamente uma precisão e uma revocação altas, próximas de 1, indicando alta relevância das recomendações do modelo. O valor do Erro Médio Absoluto também é satisfatório. Já o valor do Erro Quadrático Médio da Raiz,

Figura 42 – Treinar Modelo - Rede Neural Recorrente

```
def train(self):
    trainset, testset = self.prepare_data()

    user_ids = self.cf_model.train_data['userId'].unique()
    movie_ids = self.cf_model.train_data['movieId'].unique()

    n_users = len(user_ids)
    n_movies = len(movie_ids)

    self.user_map = {user_id: idx for idx, user_id in enumerate(user_ids)}
    self.movie_map = {movie_id: idx for idx, movie_id in enumerate(movie_ids)}

    X_train = []
    y_train = []

    for _, row in self.cf_model.train_data.iterrows():
        user_idx = self.user_map[row['userId']]
        movie_idx = self.movie_map[row['movieId']]
        rating = row['rating']
        X_train.append([user_idx, movie_idx])
        y_train.append(rating)

    X_train = np.array(X_train)
    y_train = np.array(y_train)

    # Construir o modelo RNN
    self.rnn_model = Sequential()
    self.rnn_model.add(Embedding(input_dim=n_users, output_dim=50, input_length=2))
    self.rnn_model.add(LSTM(50, activation='relu'))
    self.rnn_model.add(Dense(1))

    self.rnn_model.compile(optimizer='adam', loss='mean_squared_error')
    self.rnn_model.fit(X_train, y_train, epochs=3, batch_size=32)

    self.evaluate(testset)
```

Fonte: Autora

como no modelo de Rede Neural Convolutacional, ainda pode ser considerado alto, mas apresenta um valor melhor do que do modelo de Filtro Colaborativo. Ao comparar os resultados com os modelos anteriores, tem-se que apresenta uma melhora significativa em comparação ao modelo de Filtro Colaborativo. Entretanto, apesar das métricas estarem com valores próximos, o modelo de Rede Neural Convolutacional tem um desempenho mais satisfatório.

5.2.4 Modelo com *Autoencoder*

Usando como base o modelo de Filtros Colaborativos, por questão de memória do servidor do Colab, optou-se pela versão que utiliza 5% da base de dados.

Como nos outros modelos, no modelo com *Autoencoder*, tem-se cinco funções,

Figura 43 – Avaliar Modelo - Rede Neural Recorrente

```
def evaluate(self, testset):
    X_test = []
    y_true = []

    for (uid, mid, true_r) in testset:
        if uid in self.user_map and mid in self.movie_map:
            user_idx = self.user_map[uid]
            movie_idx = self.movie_map[mid]
            X_test.append([user_idx, movie_idx])
            y_true.append(true_r)

    X_test = np.array(X_test)
    y_true = np.array(y_true)

    y_pred = self.rnn_model.predict(X_test)

    # Calcular RMSE e MAE
    rmse = np.sqrt(np.mean((y_true - y_pred.flatten()) ** 2))
    mae = np.mean(np.abs(y_true - y_pred.flatten()))

    print(f'EQMR: {rmse}')
    print(f'EMA: {mae}')

    # Calculando Precision e Recall
    y_pred_binary = [1 if pred >= 3.5 else 0 for pred in y_pred.flatten()]
    y_true_binary = [1 if true >= 3.5 else 0 for true in y_true]

    precision = sum([1 for yt, yp in zip(y_true_binary, y_pred_binary) if yt == 1 and yp == 1]) / sum(y_pred_binary)
    recall = sum([1 for yt, yp in zip(y_true_binary, y_pred_binary) if yt == 1 and yp == 1]) / sum(y_true_binary)

    print(f'Precisão: {precision}')
    print(f'Revocação: {recall}')
```

Fonte: Autora

Figura 44 – Prever Avaliação - Rede Neural Recorrente

```
def predict(self, userId, movieId):
    user_idx = self.user_map[userId]
    movie_idx = self.movie_map[movieId]
    return self.rnn_model.predict([[user_idx, movie_idx]]).flatten()[0]
```

Fonte: Autora

Figura 45 – Obter Recomendação - Rede Neural Recorrente

```
def recommend(self, userId, n=10):
    user_idx = self.user_map[userId]
    all_movies = self.cf_model.train_data['movieId'].unique()
    predictions = [
        (movie_id, self.predict(user_idx, movie_id))
        for movie_id in all_movies
        if movie_id not in self.cf_model.train_data[
            self.cf_model.train_data['userId'] == userId
        ]['movieId'].tolist()
    ]
    predictions.sort(key=lambda x: x[1], reverse=True)
    return predictions[:n]
```

Fonte: Autora

Figura 46 – Resultados - Rede Neural Recorrente

```
EQMR: 0.7831389733204843
EMA: 0.5739887072731775
Precisão: 0.8396803867966296
Revocação: 0.7797537084048475
```

Fonte: Autora

sendo três principais (treinar, avaliar e recomendar) e duas auxiliares (preparar base de dados e predizer). Seguem as mesmas:

- Preparar Dados: Função auxiliar, responsável por carregar e mapear os dados da base de dados. Na Figura 47, é apresentado um trecho de código dessa função auxiliar;

Figura 47 – Preparar Dados - *Autoencoder*

```
def prepare_data(self):
    # Prepare os dados usando o modelo de filtro colaborativo
    train_data = self.cf_model.train_data

    user_ids = train_data['userId'].unique()
    movie_ids = train_data['movieId'].unique()

    self.user_map = {user_id: idx for idx, user_id in enumerate(user_ids)}
    self.movie_map = {movie_id: idx for idx, movie_id in enumerate(movie_ids)}

    n_users = len(user_ids)
    n_movies = len(movie_ids)

    X = np.zeros((n_users, n_movies))

    for _, row in train_data.iterrows():
        user_idx = self.user_map[row['userId']]
        movie_idx = self.movie_map[row['movieId']]
        X[user_idx, movie_idx] = row['rating']

    trainset, testset = sk_train_test_split(X, test_size=0.2, random_state=42)
    return trainset, testset
```

Fonte: Autora

- Treinar Modelo: Função principal, responsável por treinar o modelo. Na Figura 48, é apresentado um trecho de código dessa função principal;
- Avaliar Modelo: Função principal, responsável por avaliar o modelo com base nas métricas definidas na seção [Método de Coleta e Análise dos Resultados](#) (Cap. 4, se-

Figura 48 – Treinar Modelo - *Autoencoder*

```
def train(self):
    X_train, X_test = self.prepare_data()

    # Construir o modelo do Autoencoder
    input_dim = X_train.shape[1]

    input_layer = Input(shape=(input_dim,))
    encoded = Dense(64, activation='relu')(input_layer)
    decoded = Dense(input_dim, activation='sigmoid')(encoded)

    self.autoencoder_model = Model(input_layer, decoded)

    self.autoencoder_model.compile(optimizer='adam', loss='mean_squared_error')

    self.autoencoder_model.fit(X_train, X_train, epochs=3, batch_size=32)

    self.evaluate(X_test)
```

Fonte: Autora

ção 4) (Precisão, Revocação, Erro Médio Absoluto e Erro Quadrático Médio da Raiz). Na Figura 49, é apresentado um trecho de código dessa função principal;

- Prever Avaliação: Função auxiliar, responsável por prever a avaliação do usuário para os filmes. Na Figura 50, é apresentado um trecho de código dessa função auxiliar, e
- Obter Recomendação: Função principal, responsável por recomendar ao usuário um filme. Na Figura 51, é apresentado um trecho de código dessa função principal.

Os resultados do modelo treinado por Redes Neurais Recorrentes são ilustrados na Figura 52:

Pode-se perceber que, para o *Autoencoder*, a Previsão e a Revocação são iguais a zero. Isso ocorre, porque, como média de avaliação, foi definido o valor 3.5 (de 0 a 5). Acima de 3.5, é considerado positivo, sendo o item interessante para o usuário. Já abaixo de 3.5, seriam itens que o usuário teria pouco ou nenhum interesse. Com ambos os valores zerados, indica que nenhuma das previsões foi considerada positiva (≥ 3.5). Além disso, os valores de EQMR e EMA divergem bastante dos valores dos outros modelos (que estavam em 0,7 e 0,5, respectivamente). Dessa forma, a solução aplicando *Autoencoder* nos Filtros Colaborativos não pode ser considerada satisfatória.

5.3 Solução Evoluída para o Sistema de Recomendação

Comparando os modelos desenvolvidos e testados anteriormente, com base nas métricas e interpretações definidas na seção [Método de Coleta e Análise dos Resultados](#)

Figura 49 – Avaliar Modelo - *Autoencoder*

```
def evaluate(self, X_test):
    # Fazendo previsões usando o modelo do Autoencoder
    encoded_data = self.autoencoder_model.predict(X_test)

    # Recuperando os ratings originais
    y_true = X_test[X_test.nonzero()]
    y_pred = encoded_data[X_test.nonzero()]

    # Calcular RMSE e MAE
    rmse = np.sqrt(np.mean((y_true - y_pred) ** 2))
    mae = np.mean(np.abs(y_true - y_pred))

    print(f'RMSE: {rmse}')
    print(f'MAE: {mae}')

    # Calculando Precision e Recall
    y_pred_binary = [1 if pred >= 3.5 else 0 for pred in y_pred]
    y_true_binary = [1 if true >= 3.5 else 0 for true in y_true]

    tp = sum([1 for yt, yp in zip(y_true_binary, y_pred_binary) if yt == 1 and yp == 1])
    predicted_positives = sum(y_pred_binary)
    actual_positives = sum(y_true_binary)

    precision = tp / predicted_positives if predicted_positives > 0 else 0
    recall = tp / actual_positives if actual_positives > 0 else 0

    print(f'Precision: {precision}')
    print(f'Recall: {recall}')
```

Fonte: Autora

Figura 50 – Prever Avaliação - *Autoencoder*

```
def predict(self, userId, movieId):
    user_idx = self.user_map[userId]
    movie_idx = self.movie_map[movieId]
    prediction = self.autoencoder_model.predict(np.array([[user_idx]]))
    return prediction[0][movie_idx]
```

Fonte: Autora

Figura 51 – Obter Recomendação - *Autoencoder*

```
def recommend(self, userId, n=10):
    user_idx = self.user_map[userId]
    all_movies = self.cf_model.train_data['movieId'].unique()
    predictions = [
        (movie_id, self.predict(userId, movie_id))
        for movie_id in all_movies
        if movie_id not in self.cf_model.train_data[
            self.cf_model.train_data['userId'] == userId
        ]['movieId'].tolist()
    ]
    predictions.sort(key=lambda x: x[1], reverse=True)
    return predictions[:n]
```

Fonte: Autora

Figura 52 – Resultados - *Autoencoder*

```
RMSE: 3.6510350536058667
MAE: 3.49822478955724
Precision: 0
Recall: 0.0
```

Fonte: Autora

(Cap. 4, seção 4), o modelo que apresentou as melhores métricas ao ser aplicado nos Filtros Colaborativos foi o modelo de Rede Neural Convolutacional. Portanto, sendo o modelo desenvolvido ao longo desse trabalho. Esse modelo, na sequência, foi aplicado em uma *API*, e apresentado aos usuários para validação. Adiante, teve-se a coleta de dados através de questionário, como apresentado na seção [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4). Ressalta-se que as respostas desse questionário foram usadas para incrementar a base de dados e retreinar o modelo. Por fim, apresentou-se novamente ao usuário, reaplicando o questionário, no intuito de saber se ocorreram ou não melhorias nas recomendações para aqueles usuários.

5.3.1 Filtros Colaborativos e Aprendizado Profundo

Na solução evoluída para o Sistema de Recomendação, foi utilizada a mesma base de dados proposta durante as Provas de Conceito, porém agora usando 100% dela, ainda abordando a temática de recomendações voltadas para filmes. Como foram utilizados Filtros Colaborativos e Rede Neural Convolutacional, ainda ocorrerá o problema de "partida a frio". Nesse contexto, foi acrescentado ao modelo já desenvolvido durante as provas de conceito as funções `recommend_movies_for_new_user` e `calculate_score` apresentadas

nas Figuras 53 e 54, para lidar com esse problema, visto que os usuários da aplicação ainda não possuem dados na base.

- Calcular Previsão: avalia a similaridade entre os gêneros de um filme e os gêneros preferidos por um novo usuário.

Figura 53 – Calcular Previsão - RNC

```
def calculate_score(self, genres, preferred_genres):  
    # Calcula o match entre os generos do novo usuário e os generos existentes  
    movie_genres = set(genres.split('|'))  
    common_genres = movie_genres.intersection(preferred_genres)  
    return len(common_genres)
```

Fonte: Autora

- Recomendação Novo Usuário: gera recomendações de filmes para um novo usuário com base nos gêneros preferidos. Aplicou-se um filtro por gênero nessa primeira recomendação.

Figura 54 – Recomendação Novo Usuário - RNC

```
def recommend_movies_for_new_user(self, preferred_genres, n=10):  
    # Acessar dados do modelo colaborativo  
    data = self.cf_model.train_data  
  
    # Adicionar coluna de pontuação com base nos gêneros  
    data['score'] = data['genres'].apply(lambda x: self.calculate_score(x, set(preferred_genres)))  
  
    # Filtrar filmes pelos generos que o novo usuário gosta  
    filtered_movies = data[data['score'] > 0]  
  
    # Agrupar filmes e gerar média de avaliações  
    average_ratings = (  
        filtered_movies.groupby(['movieId', 'title', 'score'])['rating']  
        .mean()  
        .reset_index()  
    )  
  
    sorted_movies = average_ratings.sort_values(by=['score', 'rating'], ascending=[False, False])  
  
    # Obter recomendações para um novo usuário  
    top_n_movies = sorted_movies.head(n)  
  
    # Retornar tuplas (título, ID, média de avaliação)  
    recommendations = [(row['title'], row['movieId'], row['rating']) for _, row in top_n_movies.iterrows()]  
    return recommendations
```

Fonte: Autora

Os resultados do modelo treinado por Filtro Colaborativo e Rede Neural Convocional são ilustrados na Figura 55:

Figura 55 – Resultados - RNC Modelo Final

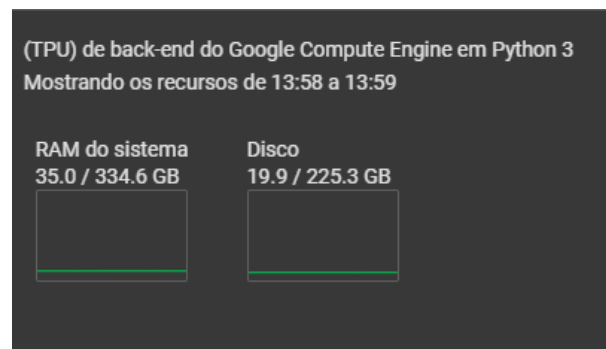
```
EQMR: 0.8028059243872688
EMA: 0.5994469822836684
Precisão: 0.8219960565119335
Revocação: 0.7838866930929272
```

Fonte: Autora

Com esses resultados, percebe-se novamente uma precisão e uma revocação muito altas, próximas de 1, indicando alta relevância das recomendações do modelo, principalmente a precisão. O valor do Erro Médio Absoluto também é satisfatório, mantendo-se baixo, próximo a 0,5. Já o valor do Erro Quadrático Médio da Raiz, apresentou uma divergência em comparação com os outros modelos testados, inclusive o próprio modelo de RNC com 5% da base, mostrando-se muito alto, o que pode não ser indicativo. Ressalta-se que é possível esse valor diminuir, ao aumentar a quantidade de épocas.

Além das métricas, o consumo de memória e consumo de disco do sistema também foram aferidos, conforme ilustrado na Figura 56.

Figura 56 – Resultados - Consumos de Memória e Disco RNC



Fonte: Autora

Os consumos de memória e disco do Sistema de Recomendação mantiveram-se baixos, comparados aos limites (334,6GB, de memória, e 225,3GB, de disco). Dessa forma, apresentando um bom desempenho de custos computacionais, apesar das 24 horas necessárias para o treinamento completo do modelo.

5.3.2 API

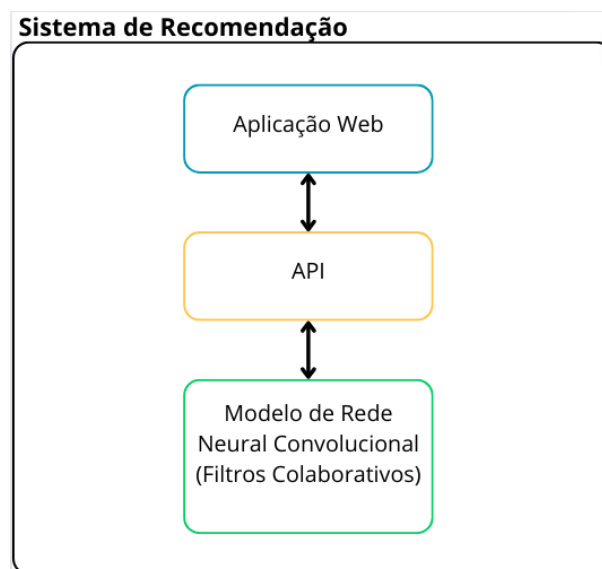
Ressalta-se, por fim, que foi desenvolvida uma aplicação *web*, com base em um protótipo de alta fidelidade, que se integra a uma *API* com o modelo de Rede Neural Convolutiva com Filtros Colaborativos aplicados. Essa aplicação retorna aos usuários recomendações de filmes com base no seu perfil, e considerando o Sistema de Recomendação centrado em Aprendizado Profundo.

Entende-se por API (*Application Programming Interface*) como um software intermediário que permite a integração facilitada entre aplicações. Ao usar APIs, tem-se maior conforto e padronização para, por exemplo, compartilhar dados e informações entre interessados. Nesse sentido, optou-se por essa forma padronizada de comunicação para integrar mais facilmente a aplicação Web, que é a camada da solução mais próxima do usuário final, com a parte *core* do Sistema de Recomendação Centrado em IA (ou seja, o Modelo de Rede Neural Convolucional usando Filtros Colaborativos).

Na Figura 57, consta um esboço da arquitetura da solução proposta, em macro nível de abstração, com destaque para as camadas:

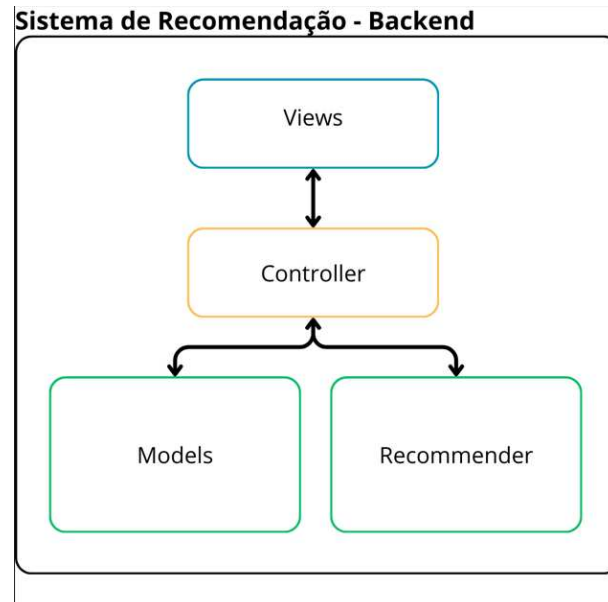
- Aplicação Web: camada arquitetural mais próxima do usuário final, contendo uma interface para facilitar o uso por parte dos interessados;
- API: camada arquitetural intermediária, que permite a integração facilitada entre a camada anterior e a próxima camada, e
- Modelo de Rede Neural Convolucional (Filtros Colaborativos): Camada mais interna, que contém a parte *core* do Sistema de Recomendação Centrado em IA.

Figura 57 – Esboço da Arquitetura da Solução Proposta (Macro Nível de Abstração)

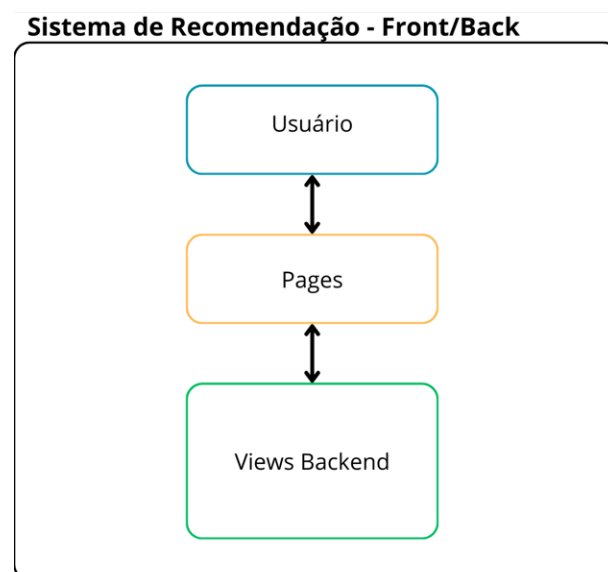


Fonte: Autora

Nas Figuras 58 e 59, tem-se a arquitetura organizada em *backend* e *frontend*. *Backend* compreendendo um modelo em 3 camadas: *view*, *controller* e *model*/lógica do Sistema de Recomendação. *Frontend* compreendendo também uma distribuição em camadas, com usuário na camada de mais alto nível de abstração; seguido da camada que contém as pages, e revelando na última camada as *Views Backend*.

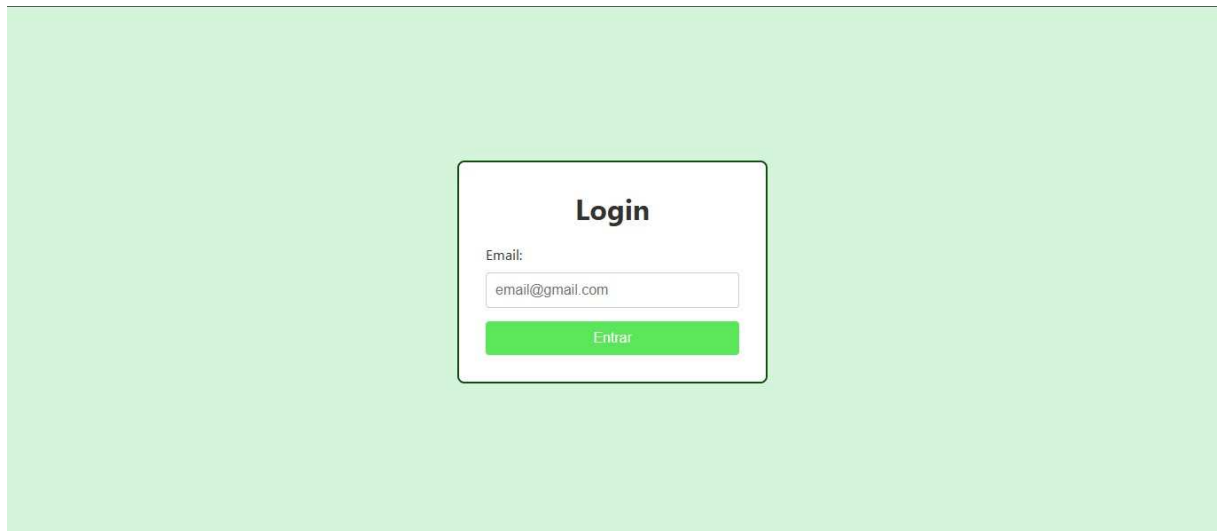
Figura 58 – Esboço da Arquitetura do *Backend* (Macro Nível de Abstração)

Fonte: Autora

Figura 59 – Esboço da Arquitetura do *Frontend* (Macro Nível de Abstração)

Fonte: Autora

Para o *frontend*, foram desenvolvidas três telas principais. A tela de login, ilustrada na Figura 60, que serve para identificar se o usuário já existe na base ou se é um novo usuário e redirecionar para a recomendação correta. A tela de gêneros, ilustrada na Figura 61, que coleta os dados do usuário, ou seja, quais gêneros ele tem interesse. Por fim, a tela de recomendação, ilustrada na Figura 62, que apresenta ao usuário as recomendações.

Figura 60 – Tela de *Login*A screenshot of a login screen. It features a light green background. In the center, there is a white rectangular box with a thin black border. Inside this box, the word "Login" is displayed in bold black text at the top. Below it, the label "Email:" is followed by a text input field containing the placeholder text "email@gmail.com". At the bottom of the box is a solid green button with the white text "Entrar".

Fonte: Autora

Figura 61 – Tela de Obter Gêneros

A screenshot of a genre selection screen. It has a light green background. A white rectangular box with a thin black border is centered on the screen. The title "Escolha seus gêneros favoritos" is at the top in bold black text. Below the title, the instruction "Selecione entre 3 a 5 gêneros:" is displayed. Underneath, there is a grid of 16 checkboxes, each followed by a genre name: Comédia, Crime, Drama, Suspense, Guerra, Musical, Romance, Aventura, Film-Noir, Ficção Científica, Faroeste, Fantasia, Mistério, Infantil, Ação, and Documentário. At the bottom of the box is a solid green button with the white text "Finalizar".

Fonte: Autora

O Fluxo do usuário na aplicação é entrar ou cadastrar (usuário existente e usuário novo, respectivamente). Para novos usuários, eles são redirecionados para a tela de gêneros, onde obtem-se os gêneros que têm interesse, para em seguida obter-se a primeira recomendação. Caso seja um usuário já existente, ele é direcionado imediatamente para a

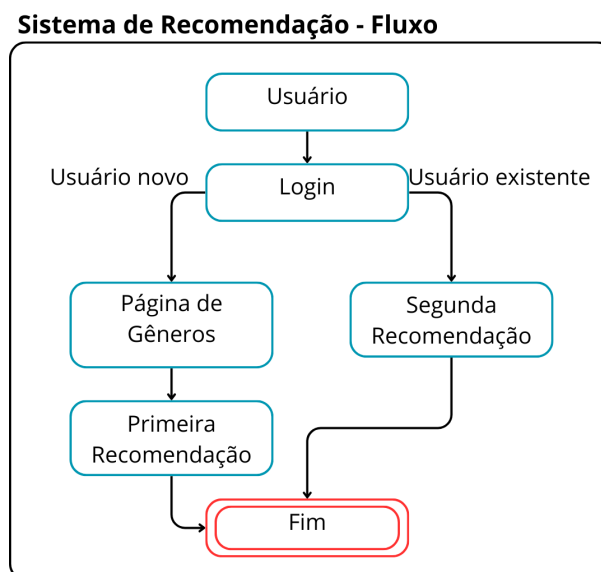
Figura 62 – Tela de Recomendação



Fonte: Autora

tela de recomendação, na qual ele obtém a segunda recomendação. A Figura 63 representa o fluxo realizado pelo usuário.

Figura 63 – Fluxo do Usuário



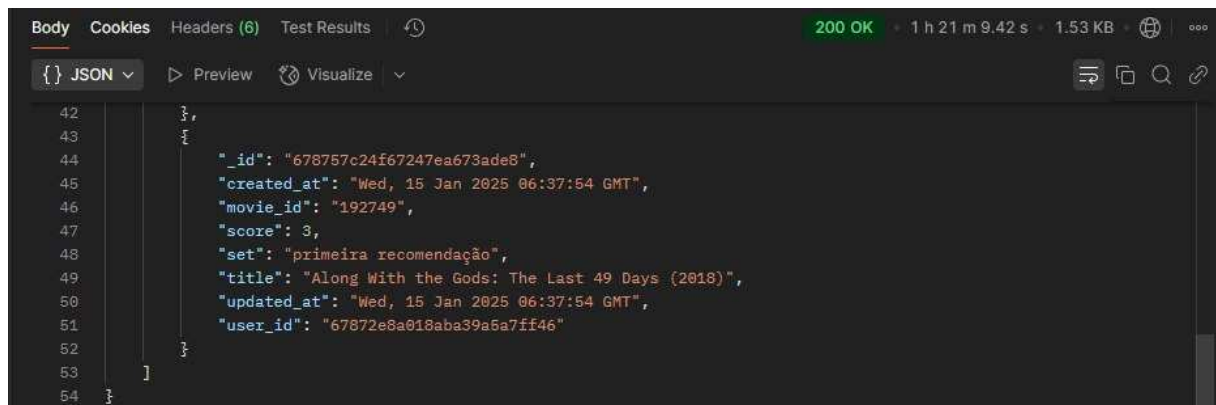
Fonte: Autora

No entanto, ao tratar-se do *backend*, apesar de conectar com sucesso com o modelo RNC criado, a aplicação apresentou problemas de desempenho, demorando mais de uma hora na resposta das requisições que realizavam as recomendações, como demonstrado nas Figuras 64 e 65. Nesse contexto, para evitar que essa demora no recebimento das recomendações influenciasse a experiência do usuário, impactando - desnecessariamente - a avaliação do Sistema de Recomendação, optou-se por elaborar mais um formulário para coletar os dados e gêneros de preferência dos usuários. Em seguida, diretamente

no modelo, recebe-se a recomendação para cada usuário; elabora-se o formulário de primeira recomendação. Com as respostas desse formulário, adiciona-se à base do modelo os usuários, treinando novamente o modelo. Assim, conclui-se com a segunda e última recomendação, a qual foi avaliada novamente pelos usuários em um terceiro formulário.

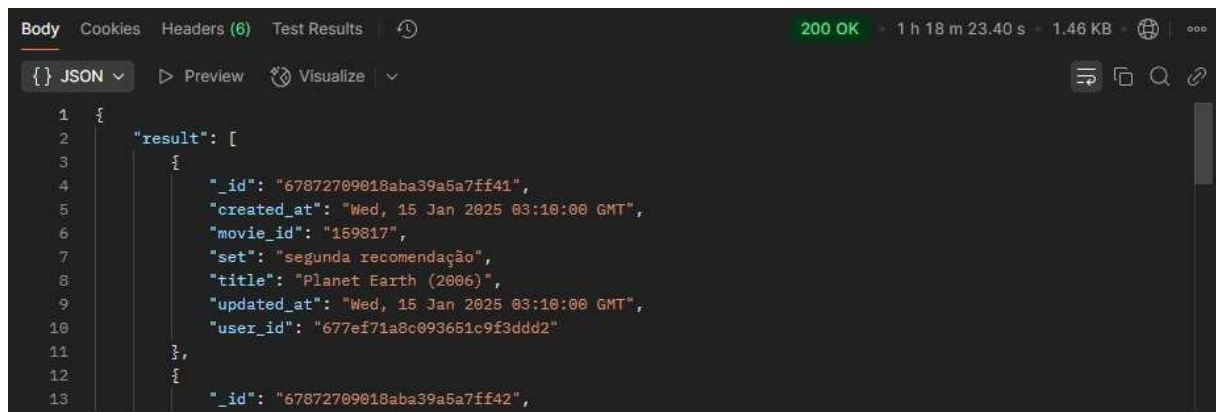
O Termo de Consentimento Livre e Esclarecido acordado aos participantes da pesquisa, encontra-se no Apêndice A. Os templates de cada formulário usado encontram-se nos Apêndices B, C e D.

Figura 64 – Resposta da Requisição de Primeira Recomendação



Fonte: Autora

Figura 65 – Resposta da Requisição de Segunda Recomendação



Fonte: Autora

5.4 Resumo do Capítulo

Este Capítulo aborda o contexto com base no qual esse trabalho surgiu, bem como a solução inicial para o caso do Sistema de Recomendação Orientado ao Aprendizado Profundo. Essa solução preliminar do sistema consistiu em um conjunto de testes com diferentes abordagens de Filtros e seus resultados, escolhendo o modelo que teve resultados mais aderentes às métricas de interesse. O modelo escolhido foi o de Rede Neural

Convolutacional com Filtros Colaborativos. Apresentou-se ainda a diferença no consumo de memória entre o Filtro Colaborativo e o de Conteúdo.

Além disso, considerando os testes realizados, originou-se a Solução Atual (já evoluída), ou seja: uma aplicação capaz de gerar recomendações de filmes com base no perfil do usuário, e que utiliza o modelo de Aprendizado Profundo escolhido. Vale ressaltar ainda, que apesar da base escolhida ter sido de filmes, o modelo pode ser utilizado para outros temas, à medida que tenham colunas similares às colunas da base utilizada (conjunto de usuários com avaliações para cada item).

Conferiu-se ainda um esboço da arquitetura da solução, sendo essa estruturada em três macro camadas: Aplicação, API e Modelo de Rede Neural. Revelou-se ainda um nível mais interno dessa arquitetura, com foco na Aplicação. Nesse caso, organizado em *frontend* e do *backend* da aplicação. Por fim, apresentou-se sobre os problemas ocorridos com as requisições no *backend*, e quais decisões foram tomadas para mitigação.

6 Análise de Resultados

Este Capítulo apresenta a análise sobre os resultados obtidos neste trabalho. A Análise de Resultado foi conduzida com base no [Método de Coleta e Análise dos Resultados](#) (Cap. 4, seção 4), apresentado anteriormente. Para isso, comenta-se, primeiramente sobre o [Perfil dos Participantes](#) (seção 1), e como ocorreram as seleções dos participantes, considerando esse perfil. Depois, são abordados os [Formulários](#) (seção 2), nos quais consta como ocorreram as coletas das avaliações realizadas, bem como quantas respostas foram obtidas. Para em seguida, apresentar [Divulgação de Resultados](#) (seção 3), que divulga as avaliações dos usuários e analisa o modelo com base nelas. Por fim, tem-se o [Resumo do Capítulo](#) (seção 4).

6.1 Perfil dos Participantes

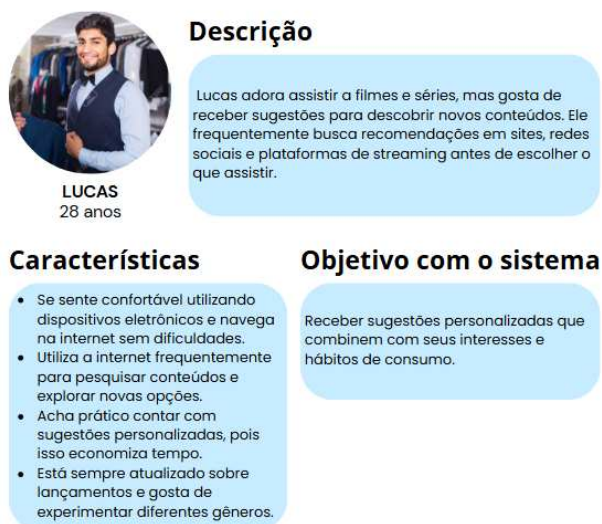
Visando estabelecer o perfil dos participantes que se enquadrariam na presente pesquisa, foram especificadas personas e antipersona, conforme apresentadas nas Figuras [66](#) a [68](#).

A primeira persona, chamada Lucas, 28 anos, representa o entusiasta de recomendações. Lucas é um usuário confortável com tecnologia, que utiliza dispositivos eletrônicos diariamente e acessa frequentemente a internet para diversas atividades, incluindo a busca por sugestões de filmes. Ele gosta de explorar novos conteúdos e confia em recomendações para otimizar seu tempo e encontrar opções alinhadas com seus gostos. Seu interesse está na personalização das sugestões, pois acredita que um sistema de recomendação pode melhorar sua experiência ao assistir filmes.

A segunda persona, Mariana, 30 anos, representa o perfil do indeciso na escolha de filmes. Mariana consome filmes e séries regularmente, mas enfrenta dificuldades na hora de decidir o que assistir. Ela sente que há muitas opções disponíveis e, por isso, acaba perdendo muito tempo navegando sem chegar a uma escolha satisfatória. Para minimizar essa indecisão, ela busca recomendações em sites especializados e redes sociais. Mariana deseja um sistema que ofereça sugestões precisas e de acordo com suas preferências, tornando sua experiência mais prática e agradável.

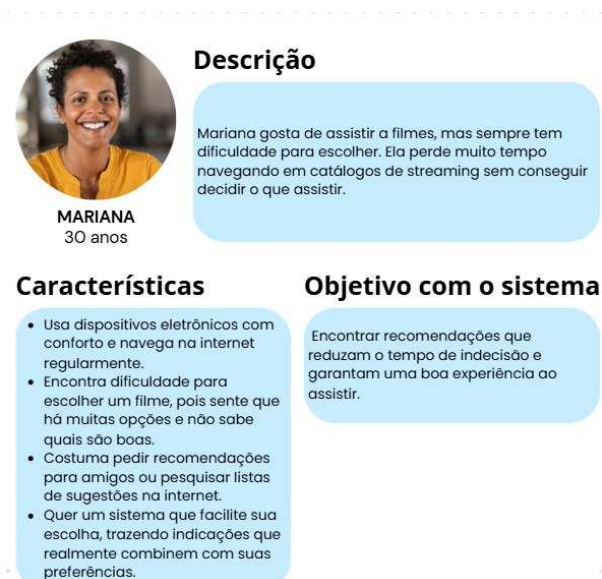
Por outro lado, a antipersona, Carlos, 65 anos, representa um usuário que não se encaixa no público-alvo do sistema. Carlos raramente utiliza dispositivos eletrônicos e tem pouca familiaridade com navegação na internet. Seu consumo de filmes é esporádico e ele não vê necessidade de receber recomendações, pois prefere assistir ao que já conhece ou ao que está passando na TV. Além disso, ele pode ter dificuldades para preencher formulários

Figura 66 – Persona 1



Fonte: Autora

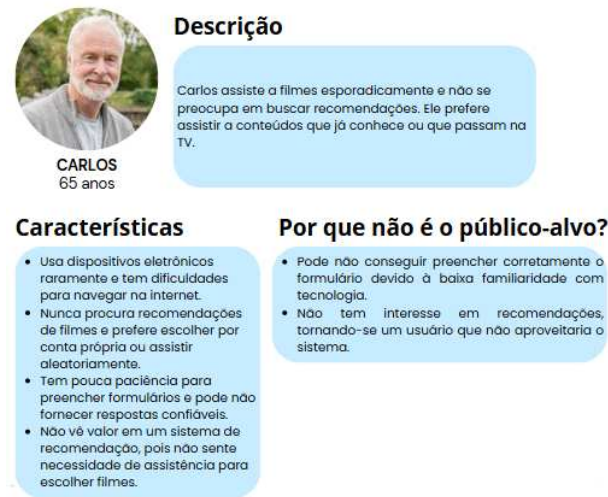
Figura 67 – Persona 2



Fonte: Autora

ou interagir com um sistema digital, tornando-se um usuário que não se beneficiaria da ferramenta proposta.

Figura 68 – Antipersona



Fonte: Autora

6.2 Formulários

Foram realizados três formulários para coleta de dados e avaliação do modelo de Sistema de Recomendação.

Para o primeiro formulário, que tinha como objetivo coletar as informações dos usuários e garantir que ele esteja dentro de um perfil desejado para a avaliação do Sistema de Recomendação, foram realizadas sete perguntas para limitar o perfil:

- Qual é o seu nível de conforto ao utilizar dispositivos eletrônicos, como computadores, smartphones ou tablets? (Valores: Muito Confortável, Confortável, Pouco Confortável, Desconfortável),
- Com que frequência você utiliza dispositivos eletrônicos no seu dia a dia? (Valores: Diariamente, Algumas vezes por semana, Raramente, Nunca),
- Você utiliza a internet para realizar atividades como compras *online*, acessar redes sociais, ou buscar informações? (Valores: Sim, frequentemente; Sim, ocasionalmente; Não, raramente; Não, nunca),
- Você consegue localizar e utilizar recursos em um site (como preencher formulários ou navegar entre páginas) sem instruções detalhadas? (Valores: Sim, facilmente; Sim, com algum esforço; Não, com dificuldade; Não sem ajuda),

- Como você classificaria sua dificuldade para escolher um filme para assistir? (Valores: Facilmente, Com algum esforço, Com dificuldade, Com muita dificuldade),
- Com que frequência você busca recomendações de filmes antes de assistir? (Valores: Frequentemente, Ocasionalmente, Raramente, Nunca), e
- Qual é o seu nome?

Para a avaliação do Sistema de Recomendação, é ideal que quem responda o questionário tenha interesse em recomendações de filmes ou tenha dificuldade para escolher um filme por conta própria, e que consiga preencher sem dificuldades os questionários. Além disso, foi perguntado o nome para identificar quais recomendações pertencem a quais usuários.

Adicionalmente, foi realizada uma pergunta para coletar os gêneros preferidos do usuário. Pergunta: Escolha de 3 a 5 gêneros de sua preferência. Valores (todos os gêneros contidos na base de dados): Comédia, Crime, Drama, Suspense, Guerra, Musical, Romance, Aventura, Filme Noir, Ficção Científica, Faroeste, Fantasia, Mistério, Infantil, Ação, Documentário, Animação, Terror, IMAX.

Para este primeiro formulário, foram obtidas dezessete respostas, das quais duas foram desconsideradas por não se enquadrarem perfil (visto anteriormente com maiores detalhes). Nesse caso, uma das pessoas não preencheu o formulário corretamente; e a outra não tinha interesse em recomendações e nem dificuldade para escolher filmes.

Já para o segundo formulário, tendo as primeiras recomendações geradas para as quinze pessoas que se encaixavam no perfil, foram apresentados a cada usuário dez filmes recomendados, com o título, sinopse e trailer (caso existesse). As perguntas foram:

- Sobre o filme <Título do Filme>. Já o assistiu? Se **sim**, qual a nota que você dá ao <Título do Filme>? (Escala de 0 a 5, sendo 0 muito ruim e 5 muito bom), e
- Se **não**, qual é o seu interesse em ver o filme <Título do Filme>? (Escala de 0 a 5, sendo 0 muito ruim e 5 muito bom).

A escala foi escolhida de 0 a 5, pois é a escala utilizada na base para as avaliações dos filmes, assim sendo possível inserir essas avaliações para os usuários que responderam ao formulário.

Para a segunda etapa, treze pessoas responderam, das quinze que avançaram para a segunda etapa. Dessa forma, as treze foram adicionadas na base, e o modelo foi treinado novamente para gerar a segunda recomendação.

Por fim, para o terceiro formulário, tendo as últimas recomendações geradas para as treze pessoas que responderam ao segundo formulário, foram novamente apresentados

a cada usuário cinco filmes recomendados, com o título, sinopse e trailer (caso existisse), e as mesmas perguntas objetivas. Além destas, foram também introduzidas três perguntas subjetivas, visando avaliar de forma qualitativa o modelo:

- Qual é a sua opinião geral sobre as recomendações que você recebeu até agora? Elas atenderam às suas expectativas? Explique;
- Numa escala de 0 a 10, qual é a nota que você atribui para sua satisfação geral com as recomendações? (Escala de 0 a 10), e
- Quais melhorias você sugeriria para que as recomendações se tornem mais precisas ou mais personalizadas para o seu perfil?

Para esse formulário, foram obtidas 12 respostas.

6.3 Divulgação de Resultados

Nessa seção, têm-se a divulgação e a interpretação das respostas dos formulários aplicados.

Para o segundo formulário, as avaliações das recomendações deram-se da seguinte forma:

- Participante 1:
 - Persona de Aderência: Persona 2

Considerando as 10 recomendações realizadas, bem como as avaliações coletadas de 0 a 5, sendo 0, a nota mais baixa, e 5 a nota mais alta, tem-se que o primeiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 3;
- Recomendação 3: nota 4;
- Recomendação 4: nota 2;
- Recomendação 5: nota 5;
- Recomendação 6: nota 0;
- Recomendação 7: nota 0;
- Recomendação 8: nota 0;
- Recomendação 9: nota 0, e

- Recomendação 10: nota 2.

Ao final, obteve-se, como média das notas das recomendações conferidas ao primeiro participante: 2,1;

- Participante 2:

- Persona de Aderência: Persona 2

Considerando as 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o segundo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 0;
- Recomendação 2: nota 0;
- Recomendação 3: nota 0;
- Recomendação 4: nota 0;
- Recomendação 5: nota 0;
- Recomendação 6: nota 5;
- Recomendação 7: nota 0;
- Recomendação 8: nota 0;
- Recomendação 9: nota 0, e
- Recomendação 10: nota 0.

Ao final, obteve-se, como média das notas das recomendações conferidas ao segundo participante: 0,5;

- Participante 3:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o terceiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 3;
- Recomendação 3: nota 3;
- Recomendação 4: nota 0;
- Recomendação 5: nota 5;
- Recomendação 6: nota 4;

- Recomendação 7: nota 1;
- Recomendação 8: nota 4;
- Recomendação 9: nota 5, e
- Recomendação 10: nota 2;

Ao final, obteve-se, como média das notas das recomendações conferidas ao terceiro participante: 3,1;

- Participante 4:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o quarto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 4;
- Recomendação 3: nota 3;
- Recomendação 4: nota 1;
- Recomendação 5: nota 5;
- Recomendação 6: nota 5;
- Recomendação 7: nota 1;
- Recomendação 8: nota 3;
- Recomendação 9: nota 2, e
- Recomendação 10: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao quarto participante: 3,0;

- Participante 5:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o quinto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 3;
- Recomendação 3: nota 3;

- Recomendação 4: nota 2;
- Recomendação 5: nota 3;
- Recomendação 6: nota 4;
- Recomendação 7: nota 3;
- Recomendação 8: nota 2;
- Recomendação 9: nota 3, e
- Recomendação 10: nota 2.

Ao final, obteve-se, como média das notas das recomendações conferidas ao quinto participante: 2,8;

- Participante 6:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o sexto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 4;
- Recomendação 3: nota 3;
- Recomendação 4: nota 2;
- Recomendação 5: nota 2;
- Recomendação 6: nota 5;
- Recomendação 7: nota 4;
- Recomendação 8: nota 5;
- Recomendação 9: nota 5, e
- Recomendação 10: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao sexto participante: 3,6;

- Participante 7:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o sétimo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 3;
- Recomendação 3: nota 3;
- Recomendação 4: nota 2;
- Recomendação 5: nota 1;
- Recomendação 6: nota 2;
- Recomendação 7: nota 0;
- Recomendação 8: nota 2;
- Recomendação 9: nota 0, e
- Recomendação 10: nota 1.

Ao final, obteve-se, como média das notas das recomendações conferidas ao sétimo participante: 1,8;

- Participante 8:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o oitavo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 2;
- Recomendação 3: nota 3;
- Recomendação 4: nota 4;
- Recomendação 5: nota 3;
- Recomendação 6: nota 2;
- Recomendação 7: nota 3;
- Recomendação 8: nota 2;
- Recomendação 9: nota 3, e
- Recomendação 10: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao oitavo participante: 2,9;

- Participante 9:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o nono participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 4;
- Recomendação 3: nota 5;
- Recomendação 4: nota 1;
- Recomendação 5: nota 2;
- Recomendação 6: nota 2;
- Recomendação 7: nota 5;
- Recomendação 8: nota 3;
- Recomendação 9: nota 5, e
- Recomendação 10: nota 4.

Ao final, obteve-se, como média das notas das recomendações conferidas ao nono participante: 3,4;

- Participante 10:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 1;
- Recomendação 3: nota 3;
- Recomendação 4: nota 1;
- Recomendação 5: nota 3;
- Recomendação 6: nota 5;
- Recomendação 7: nota 0;
- Recomendação 8: nota 1;
- Recomendação 9: nota 4, e
- Recomendação 10: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo participante: 2,6;

- Participante 11:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo primeiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 4;
- Recomendação 3: nota 0;
- Recomendação 4: nota 2;
- Recomendação 5: nota 0;
- Recomendação 6: nota 5;
- Recomendação 7: nota 1;
- Recomendação 8: nota 4;
- Recomendação 9: nota 0, e
- Recomendação 10: nota 5.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo primeiro participante: 2,4;

- Participante 12:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo segundo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 5;
- Recomendação 3: nota 3;
- Recomendação 4: nota 4;
- Recomendação 5: nota 5;
- Recomendação 6: nota 5;
- Recomendação 7: nota 1;
- Recomendação 8: nota 4;
- Recomendação 9: nota 3, e

- Recomendação 10: nota 1.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo segundo participante: 3,6, e

- Participante 13:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo terceiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 1;
- Recomendação 2: nota 2;
- Recomendação 3: nota 1;
- Recomendação 4: nota 1;
- Recomendação 5: nota 0;
- Recomendação 6: nota 3;
- Recomendação 7: nota 1;
- Recomendação 8: nota 3;
- Recomendação 9: nota 0, e
- Recomendação 10: nota 1.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo terceiro participante: 1,3.

O Sistema de Recomendação, nessa primeira etapa, apresentou resultados abaixo do esperado, recomendando filmes que os usuários não tinham interesse. Dessa forma, o método de recomendação para novos usuários precisou ser revisado e melhorado.

Para o terceiro e último formulário, as avaliações das recomendações deram-se da seguinte forma:

- Participante 1:

- Persona de Aderência: Persona 2

Considerando as 5 recomendações realizadas, bem como as avaliações coletadas de 0 a 5, sendo 0, a nota mais baixa, e 5 a nota mais alta, tem-se que o primeiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 0;
- Recomendação 2: nota 0;
- Recomendação 3: nota 2;
- Recomendação 4: nota 0, e
- Recomendação 5: nota 0.

Ao final, obteve-se, como média das notas das recomendações conferidas ao primeiro participante: 0,4; nota geral do Sistema de Recomendação: 2 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: sem comentários;

- Participante 2:

- Persona de Aderência: Persona 2

Considerando as 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o segundo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 3;
- Recomendação 3: nota 4;
- Recomendação 4: nota 3, e
- Recomendação 5: nota 4.

Ao final, obteve-se, como média das notas das recomendações conferidas ao segundo participante: 3,8; nota geral do Sistema de Recomendação: 6 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: trazer mais recomendações, apenas 5 não são suficientes para avaliar. Recomendações baseadas nas notas de sites famosos (IMDB, RottenTomatoes);

- Participante 3:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o terceiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 4;
- Recomendação 3: nota 5;
- Recomendação 4: nota 4, e

- Recomendação 5: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao terceiro participante: 4,2; nota geral do Sistema de Recomendação: 8 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: o primeiro formulário misturava muitos gêneros num filme só, ficando muito longe do seu interesse, mas o segundo trouxe recomendações mais certas;

- Participante 4:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o quarto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 2;
- Recomendação 3: nota 4;
- Recomendação 4: nota 4, e
- Recomendação 5: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao quarto participante: 3,2; nota geral do Sistema de Recomendação: 9 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: pegar mais informações sobre os usuários para as recomendações serem mais interessantes;

- Participante 5:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o quinto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 3;
- Recomendação 3: nota 4;
- Recomendação 4: nota 3, e
- Recomendação 5: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao quinto participante: 3,4; nota geral do Sistema de Recomendação: 7 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: somente um filme despertou real interesse, os outros ficou mais neutra, tendo a possibilidade de assistir;

- Participante 6:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o sexto participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 3;
- Recomendação 3: nota 2;
- Recomendação 4: nota 0, e
- Recomendação 5: nota 0.

Ao final, obteve-se, como média das notas das recomendações conferidas ao sexto participante: 1,8; nota geral do Sistema de Recomendação: 3 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: as primeiras recomendações foram mais precisas. Mais perguntas para definir o perfil de usuário;

- Participante 7:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o sétimo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 5;
- Recomendação 3: nota 3;
- Recomendação 4: nota 4, e
- Recomendação 5: nota 4.

Ao final, obteve-se, como média das notas das recomendações conferidas ao sétimo participante: 4,2; nota geral do Sistema de Recomendação: 9 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: recomendações baseadas nas notas de sites famosos (IMDB, RottenTomatoes) e em mais gêneros de interesse;

- Participante 8:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o oitavo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 4;
- Recomendação 2: nota 5;
- Recomendação 3: nota 4;
- Recomendação 4: nota 3, e
- Recomendação 5: nota 3.

Ao final, obteve-se, como média das notas das recomendações conferidas ao oitavo participante: 3,8; nota geral do Sistema de Recomendação: 8 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: coleta de preferências diretas e *feedback* contínuo;

- Participante 9:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o nono participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 3;
- Recomendação 2: nota 5;
- Recomendação 3: nota 1;
- Recomendação 4: nota 4, e
- Recomendação 5: nota 4.

Ao final, obteve-se, como média das notas das recomendações conferidas ao nono participante: 3,4; nota geral do Sistema de Recomendação: 8 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: definir melhor perfil de usuário;

- Participante 10:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 5;
- Recomendação 2: nota 1;
- Recomendação 3: nota 0;
- Recomendação 4: nota 5, e
- Recomendação 5: nota 4.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo participante: 3,0; nota geral do Sistema de Recomendação: 7 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: formulário mais detalhado para formar perfil de usuário;

- Participante 11:

- Persona de Aderência: Persona 1

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo primeiro participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 0;
- Recomendação 2: nota 4;
- Recomendação 3: nota 5;
- Recomendação 4: nota 5, e
- Recomendação 5: nota 2.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo participante: 3,2; nota geral do Sistema de Recomendação: 6 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: captar mais dados relacionados aos estilos do filmes e cruzar informações mais pessoais da pessoa que procura a recomendação, e

- Participante 12:

- Persona de Aderência: Persona 2

Tendo em vista a mesma dinâmica, com 10 recomendações realizadas, bem como a mesma escala de notas, tem-se que o décimo segundo participante conferiu as seguintes notas por recomendação:

- Recomendação 1: nota 1;
- Recomendação 2: nota 3;
- Recomendação 3: nota 5;
- Recomendação 4: nota 1, e
- Recomendação 5: nota 1.

Ao final, obteve-se, como média das notas das recomendações conferidas ao décimo participante: 2,2; nota geral do Sistema de Recomendação: 7 (de 0 a 10, sendo 0, a nota mais baixa, e 10 a nota mais alta), e como comentários: recomendações mais voltadas pro gosto do usuário.

Com base nas respostas do último formulário, considerando apenas as notas das recomendações, pode-se inferir que para três pessoas as recomendações foram ruins e não eram de seu interesse. Já para outras duas pessoas as recomendações foram bastante interessantes e tiveram notas mais altas. Porém, para a maioria dos usuários, as recomendações foram medianas, neutras.

No entanto, ao serem adicionadas as respostas das perguntas subjetivas à análise, a maioria dos usuários conferiu uma nota alta ao Sistema de Recomendação, com algumas ressalvas nos comentários; quatro deram notas medianas, e dois ainda julgaram que o Sistema foi ruim e não lhes atendeu.

Nos comentários, repete-se muito que seria interessante uma análise mais detalhada do perfil dos usuários, para que as recomendações atendessem melhor aos seus gostos.

Ao comparar o resultado da segunda recomendação com a primeira, tem-se uma melhora significativa no interesse dos usuários. O Sistema de Recomendação conferiu recomendações mais válidas, e mais aderentes aos gostos dos usuários. Dessa forma, avaliando o Sistema de Recomendação como um todo, é necessário melhorar as recomendações iniciais aos usuários novos. Entretanto, ressalta-se que, em se tratando de usuários já existentes, mesmo com poucos dados disponíveis para esses usuários, o Sistema apresentou um comportamento mediano a satisfatório ao trazer algumas recomendações interessantes aos usuários.

6.4 Resumo do Capítulo

Este Capítulo concentrou-se na análise dos resultados obtidos ao avaliar o Sistema de Recomendação orientado a Aprendizado Profundo. Nesse contexto, abordou-se sobre o perfil dos participantes, bem como sobre como foram coletados os dados para análise, via formulários. Além disso, revelou-se sobre quais foram os resultados provenientes desses dados e suas respectivas avaliações, conferidas pelos participantes.

Conclui-se que o Sistema de Recomendação apresentou um resultado insatisfatório para as primeiras recomendações geradas, as quais apenas filtravam por gênero, não trazendo os benefícios da Rede Neural Convolucional. Já para a segunda recomendação, mesmo com um acréscimo pequeno (apenas 10 avaliações de filmes), foram percebidas melhoras nas avaliações dos usuários. Coloca-se ainda, por fim, que o Sistema de Recomendação, em geral, foi bem avaliado pelos participantes. Mas, ainda cabem refinamentos na solução para uma melhor adequação das recomendações aos gostos dos usuários do sistema.

7 Conclusão

Neste Capítulo, são apresentadas as considerações finais Trabalho, bem como os resultados obtidos até então. Divide-se em seções, iniciando com [Atividades Realizadas](#) (seção 1), que apresenta as atividades que foram cumpridas, com base no que foi especificado na seção [Fluxo das Atividades/Subprocessos](#) (Cap. 4, seção 5). Em seguida, apresenta-se o [Status dos Objetivos Específicos](#) (seção 2), que aborda atendimento aos Objetivos Específicos definidos em [Objetivos Específicos](#) (Cap. 1, seção 4.2), bem como procura-se responder sobre as Questões de Pesquisa e Desenvolvimento. Na sequência, têm-se as [Contribuições e Limitações](#) (seção 3), que aborda as contribuições e limitações do trabalho. Depois, são acordadas as [Impressões da Autora em Relação ao Trabalho](#) (seção 4), que confere reflexões da autora sobre o trabalho desenvolvido. Para em seguida, sugerir possíveis melhorias ao trabalho no [Trabalhos Futuros](#) (seção 5). Por último, tem-se o [Resumo do Capítulo](#) (seção 6), que aborda sobre os principais tópicos apresentados no Capítulo.

7.1 Atividades Realizadas

A primeira etapa do trabalho teve como foco atividades que fundamentaram e basearam a proposta deste trabalho, além de pesquisas, para melhor construção do conhecimento sobre os tópicos de interesse do projeto, e testes práticos, para experienciar diferentes modelos de Inteligência Artificial. Com base no [Fluxo das Atividades/Subprocessos](#) (Cap. 4, seção 5), o Quadro 13 apresenta o estado de atividades e subprocessos definidos para a primeira etapa do trabalho.

Ressalta-se que, ao longo da realização da primeira etapa do TCC, houve um crescimento de conhecimento bastante razoável por parte da autora, em especial sobre:

- (i) Inteligência Artificial, incluindo filtros específicos, ora colaborativos, ora baseados em conteúdo, ora híbridos, bem como algoritmos de Aprendizado Profundo usando Redes Neurais, e
- (ii) Sistemas de Recomendação, incluindo o ciclo de atividades inerentes a esses sistemas, da coleta à avaliação.

Adicionalmente, a autora experimentou na prática vários algoritmos, bem como suportes tecnológicos, o que conferiu uma visão bem mais concreta e coerente sobre a proposta em si. Antes, quando tudo ainda estava no contexto das ideias, a autora tinha uma visão mais teórica sobre a proposta. Depois, vivenciando a prática, a autora compreendeu de forma mais clara sobre: o que era alcançável; o que ainda era embrionário, mesmo

orientando-se pela literatura especializada, e o que vislumbrar como possíveis extensões da proposta (ou seja, trabalhos futuros desejados para dar continuidade aos esforços da autora).

Quadro 13 – Andamento de Atividades e Subprocessos da Primeira Etapa do TCC

Atividades e Subprocessos	Andamento
Definir Tema	Concluída
Levantar Bibliografia	Concluída
Descrever Introdução	Concluída
Levantamento do Referencial Teórico	Concluída
Descrever Suporte Tecnológico	Concluída
Descrever Metodologia	Concluída
Desenvolvimento Prova de Conceito	Concluída
Descrever Proposta	Concluída
Descrever Conclusões Iniciais	Concluída
Apresentar Monografia	Concluída

Fonte: Autora

Na segunda etapa deste trabalho, o foco esteve em desenvolver a proposta apresentada no [Sistema de Recomendação com Aprendizado Profundo](#) (Cap. 5), aplicando a teoria apresentada no [Referencial Teórico](#) (Cap. 2), com o apoio das tecnologias mostradas no [Suporte Tecnológico](#) (Cap.3).

O Quadro 14 apresenta o andamento de atividades e subprocessos referentes à segunda etapa do trabalho.

Quadro 14 – Andamento de Atividades e Subprocessos da Segunda Etapa do TCC

Atividades e Subprocessos	Andamento
Aplicar Correções da Banca	Concluída
Separar Base de Dados	Concluída
Desenvolvimento do Modelo de Sistema de Recomendação	Concluída
Desenvolvimento da API	Concluída
Aplicar Modelo na API	Concluída, com ressalvas
Analisar Resultados	Concluída
Usar Resultados para Incrementar o Modelo	Concluída
Reaplicar Novo Modelo na API	Concluída
Análise dos Novos Resultados	Concluída
Refinar Monografia	Concluída
Apresentar Monografia	Em Andamento

Fonte: Autora

7.2 Status dos Objetivos Específicos

Na seção **Objetivos Específicos** (Cap. 1, seção 4.2), foram definidos os Objetivos Específicos deste trabalho, os quais já foram atendidos. Segue uma breve análise sobre esses objetivos, com indicação dos insumos gerados:

- Levantamento sobre Sistemas de Recomendação: foi apresentado no **Referencial Teórico** (Cap. 2). Esse objetivo foi alcançado pela autora, que realizou pesquisa bibliográfica, conforme consta no **Método de Pesquisa Bibliográfica** (Cap. 4, seção 2). A partir de *strings* de busca, critérios de seleção (inclusão e exclusão), bem como leituras de artigos, foi possível obter um levantamento sobre Sistemas de Recomendação, com destaque para o estudo do ciclo de atividades inerentes ao processo utilizado em um Sistema de Recomendação, desde a coleta à avaliação, perpassando por: armazenamento, filtragem, análise e geração.
- Levantamento, no contexto da IA, de algoritmos de Aprendizado Profundo que mais atendam à proposta desse trabalho: foi apresentado no **Referencial Teórico** (Cap. 2) e no **Sistema de Recomendação com Aprendizado Profundo** (Cap. 5). Nesse caso, a autora procurou não apenas revelar uma visão orientada à literatura especializada, constante no Referencial Teórico que embasa o trabalho como um todo; como também procurou apresentar provas de conceitos, que conferem, desde à primeira etapa do TCC, uma visão de cunho aplicado, com insumos implementados e disponíveis em repositórios dedicados para consulta. Caso seja do interesse, os repositórios podem ser acessados via os *links*:
 - Prova de Conceito de Filtro Colaborativo (5% da Base de Dados):
<https://github.com/CibeleG/TCC/tree/main/poc-fc-5>
 - Prova de Conceito de Filtro Colaborativo (40% da Base de Dados):
<https://github.com/CibeleG/TCC/tree/main/poc-fc-40>
 - Prova de Conceito de Filtro Colaborativo (100% da Base de Dados):
<https://github.com/CibeleG/TCC/tree/main/poc-fc-100>
 - Prova de Conceito de Autoencoder:
<https://github.com/CibeleG/TCC/tree/main/poc-autoencoder>
 - Prova de Conceito de Rede Neural Convolutacional:
<https://github.com/CibeleG/TCC/tree/main/poc-cnn>
 - Prova de Conceito de Rede Neural Recorrente:
<https://github.com/CibeleG/TCC/tree/main/poc-rnn>
 - Sistema de Recomendação Rede Neural Convolutacional + Filtro Colaborativo:
<https://github.com/CibeleG/TCC/tree/main/CNN-SR>

- *Backend* da Aplicação:
<https://github.com/CibeleG/TCC/tree/main/back>
- *Frontend* da Aplicação:
<https://github.com/CibeleG/TCC/tree/main/front>
- Treinamento da base de dados orientando-se pelos algoritmos de Aprendizado Profundo levantados: foi apresentado no [Sistema de Recomendação com Aprendizado Profundo](#) (Cap. 5). Novamente, o intuito da autora foi conferir insumos mais concretos, para que a proposta fosse algo alcançável, e não uma proposta teórica, sem uma análise mais prática e aplicada de fato. Nesse sentido, com as provas de conceito, a autora experienciou vários algoritmos de Aprendizado Profundo, fazendo uso de filtros, de forma individual, ou combinada (Filtros Híbridos). Selecionou-se o modelo de Rede Neural Convolutacional com Filtros Colaborativos.
- Desenvolvimento de um Sistema de Recomendação Centrado em IA: foi criado o modelo de Aprendizado Profundo apresentado na seção [Filtros Colaborativos e Aprendizado Profundo](#) (Cap. 5, seção 3.1).
- Análise da Satisfação do Usuário considerando esse Sistema de Recomendação: foi desenvolvido após a criação do modelo, e discutido no Capítulo [Análise de Resultados](#) (Cap. 6).

Diante do exposto, é possível responder às questões de pesquisa e desenvolvimento, definidas do Cap. 1.

Questão de Desenvolvimento: Como um Sistema de Recomendação centrado em Aprendizado Profundo pode ser desenvolvido?

A resposta encontra-se tratada ao longo do Cap. 5, no qual são apresentados vários detalhes sobre o desenvolvimento do Sistema de Recomendação orientando-se por um modelo de IA, mais especificamente, Filtros Colaborativos e Aprendizado Profundo, com estudos exploratórios usando Redes Neurais Convolucionais, Redes Neurais Recorrentes e *Autoencoder*.

Questão de Pesquisa: Esse Sistema de Recomendação centrado em Aprendizado Profundo, de fato, proporciona maior satisfação aos seus usuários?

A resposta encontra-se tratada ao longo do Cap. 6, no qual constam vários insumos sobre os resultados obtidos, usando como base validações junto ao público alvo, sendo esse definido a partir da especificação de personas, e testado usando questionários e ciclos de pesquisa centrados em métricas específicas.

7.3 Contribuições e Limitações

Esse trabalho contribui apresentando um modelo básico de Sistema de Recomendação por Aprendizado Profundo que utiliza RNC e Filtros Colaborativos, combinando SVD (*Singular Value Decomposition*), um método clássico de filtragem colaborativa, com uma rede neural convolucional (RNC) para melhorar as recomendações. Isso representa um modelo híbrido, onde:

- O SVD capta padrões latentes de usuários e filmes com base em avaliações passadas;
- A RNC aprende representações numéricas a partir das interações entre usuários e filmes, o que pode permitir uma melhor generalização e refinamento das previsões, e
- A RNC usa *embeddings* de usuários e filmes e os processa com camadas convolucionais, o que pode ajudar a modelar padrões mais complexos nos dados.

Apresenta-se um modelo diferente para um Sistema de Recomendação, visto que a maioria dos Sistemas de Recomendação usa RNCs para análise de conteúdo (ex.: imagens de produtos), e não para filtragem colaborativa.

Outra contribuição importante é que o modelo converge em apenas 2 épocas, o que indica eficiência computacional, especialmente considerando a grande base de dados (25 milhões de registros). O sistema usa *embeddings* para representar usuários e filmes, reduzindo a dimensionalidade e permitindo um treinamento mais rápido. A recomendação ocorre de maneira otimizada, utilizando *batch prediction* ao invés de previsões individuais.

Ressalta-se ainda que o Sistema de Recomendação apresenta excelentes valores para as métricas de precisão e revocação, como apresentado na seção [Filtros Colaborativos e Aprendizado Profundo](#) (Cap. 5, seção 3.1).

No entanto, este trabalho possui algumas limitações, sendo elas:

- Necessita de um ambiente de execução com alta RAM, ambientes os quais só se encontram disponíveis em versões pagas;
- Ainda apresenta o problema de "partida a frio", visto que mesmo tendo uma função para primeiras recomendações, essas não tiveram uma boa avaliação pelos os usuários, vide seção [Análise de Resultados](#) (Cap. 6);
- A integração da aplicação (parte próxima ao usuário) com o Sistema de Recomendação em si possui incorrências que ainda demandam atenção, uma vez que foi observado um tempo maior para responder as requisições oriundas da interface gráfica, e

- Escalabilidade continua sendo um problema, visto que a aplicação necessita de ambiente com alta RAM para sua execução, principalmente devido ao tamanho da base de dados (25 milhões de registros), que a medida que à medida que novos usuários são adicionados, tende-se a aumentar o tempo de execução, prejudicando a escalabilidade.

7.4 Impressões da Autora em Relação ao Trabalho

No presente Trabalho de Conclusão de Curso, ambas as etapas foram de grande aprendizado para a autora, que possuía pouco conhecimento na área de IA e de Aprendizado Profundo. Construiu-se uma base para o compreender os conceitos e definir qual caminho dentro da área de Inteligência Artificial seria seguido durante o [Referencial Teórico](#) (Cap. 2). Durante a [Solução Inicial para o Sistema de Recomendação](#) (Cap. 5, seção 2), foi possível evoluir os conhecimentos técnicos necessários para o desenvolvimento do Sistema de Recomendação. A autora agora considera ter uma base mais sólida na área, principalmente nos principais modelos utilizados no desenvolvimento do Sistema de Recomendação (RNC e Filtros Colaborativos).

Reporta-se satisfação com o aprendizado adquirido e com as recomendações finais, considerando a avaliação positiva da maioria dos usuários avaliados. Entretanto, em se tratando da plenitude dos resultados obtidos, ainda há insatisfação com algumas etapas do trabalho, tanto pelo ambiente de execução, a necessitar de mais memória para conseguir treinar o Sistema de Recomendação com mais épocas; quanto alguns problemas comuns aos Sistemas de Recomendação que não foram completamente resolvidos. Sendo estes, a partida a frio, que apesar de ter uma solução implementada, apresentou recomendações desinteressantes aos usuários avaliados, e a escalabilidade, que também pode apresentar um problema à medida que a base de dados aumenta.

Os principais desafios enfrentados pela autora durante a primeira etapa do trabalho foram:

- Memória RAM insuficiente no ambiente de execução, necessitando de uma evolução no Colab para TPU, e
- Implementar eficientemente Máquina Restrita de Boltzmann e Filtros Híbridos (Filtro Colaborativo + Filtro por Conteúdo).

Já para a segunda etapa do trabalho, os desafios foram:

- Fazer o modelo convergir agora com a base total. A memória novamente apresentou-se como um problema, com o ambiente de execução constantemente desconectando;

- Fazer o modelo integrado na API ter um tempo de resposta de requisições útil. O tempo médio alcançado foi de uma hora e vinte minutos;
- Melhorar o tempo de treinamento. O modelo demorando cerca de 24 horas para finalizar o treinamento, e
- Evitar pequenos contratempos. Por exemplo, ao utilizar toda a base, o treinamento da RNC ignorava o primeiro usuário. A solução para tal problema demandou muito tempo para ser descoberta.

7.5 Trabalhos Futuros

Para Trabalhos Futuros, considerando as limitações mencionadas na [Contribuições e Limitações](#) (Cap. 7, seção 3), sugere-se:

- Mudar a forma de integração do modelo com a API, a fim de diminuir o tempo de resposta das requisições;
- Aumentar a memória e o disco do ambiente de execução, permitindo aumentar ainda o número de épocas durante o treinamento do modelo, para assim verificar a precisão das recomendações;
- Acrescentar na aplicação a avaliação das recomendações obtidas, não necessitando realizar vários formulários, considerando a desistência de alguns usuários de um formulário para outro;
- Trabalhar com um conjunto maior de usuários para validação do modelo;
- Aplicar um filtro de conteúdo em conjunto com o filtro colaborativo tanto para as primeiras recomendações, para evitar o problema de "partida a frio", quanto para as recomendações de usuários existentes, assim tendo mais conhecimento sobre os gostos dos usuários, e
- Realizar um levantamento inicial com novos usuários de forma a conhecer seus interesses e melhorar o método de filtro por gênero na primeira recomendação.

Esses pontos listados apresentam oportunidades de melhoria e podem ser abordados em trabalhos futuros relacionados a esse projeto.

7.6 Resumo do Capítulo

Na primeira etapa deste trabalho, foi levantada a bibliografia necessária para embasar teoricamente este trabalho, e também foram realizadas algumas provas de conceito

para testar a teoria e definir o melhor modelo a ser utilizado no trabalho. Dessa forma, todas as atividades previstas para a primeira parte do trabalho, foram concluídas. Já para a segunda etapa, foi realizado o desenvolvimento do que foi proposto na primeira etapa e a análise da satisfação do usuário. Assim, concluindo os **Objetivos Específicos** (Cap. 1, seção 4.2) e o **Objetivo Geral** (Cap. 1, seção 4.1), sendo esse último: "estudo sobre Sistemas de Recomendação centrados em Aprendizado Profundo, revelando como desenvolvê-los e analisando se os mesmos melhoram a satisfação dos usuários."

Referências

- ACEVEDO, C. R. et al. As plataformas de streaming e seu impacto no comportamento do consumidor. *REA-Revista Eletrônica de Administração*, v. 19, n. 2, p. 287–300, 2020. Citado na página 73.
- AHMED, I. *Getting Started with a Movie Recommendation System*. 2024. On Kaggle. Disponível em: <<https://www.kaggle.com/code/ibtesama/getting-started-with-a-movie-recommendation-system#Content-Based-Filtering>>. Citado na página 51.
- ALLAM, A.; DAHLAN, H. M. User experience: challenges and opportunities. *Journal of Information Systems Research and Innovation*, v. 3, n. 1, 2013. Citado na página 47.
- ANALYTICS, E. *Cold Start Problem*. 2021. Disponível em: <<https://www.expressanalytics.com/blog/cold-start-problem/>>. Citado na página 27.
- APPLEGATE, R. Models of user satisfaction: Understanding false positives. *RQ*, v. 32, n. 4, 1993. Summer. Citado na página 47.
- BAZDYREV, A. *Neural Network Baseline for Recommender Systems*. 2019. On Kaggle. Disponível em: <<https://www.kaggle.com/code/abazdyrev/neural-network-baseline-for-recommender-systems>>. Citado na página 51.
- BURKE, R. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, Springer, v. 12, 2002. Citado 8 vezes nas páginas 35, 36, 38, 39, 40, 41, 42 e 58.
- CANVA. *Canva*. 2024. Disponível em: <<https://www.canva.com/>>. Citado 2 vezes nas páginas 53 e 54.
- CHAI, T.; DRAXLER, R. R. et al. Root mean square error (rmse) or mean absolute error (mae). *Geoscientific model development discussions*, v. 7, n. 1, p. 1525–1534, 2014. Citado na página 62.
- CHEW, L.-J.; HAW, S.-C.; SUBRAMANIAM, S. Recommender system for retail domain: An insight on techniques and evaluations. In: *Proceedings of the 12th International Conference on Computer Modeling and Simulation*. New York, NY, USA: Association for Computing Machinery, 2020. (ICCMS '20). ISBN 9781450377034. Disponível em: <<https://doi.org/10.1145/3408066.3408101>>. Citado na página 25.
- DATAI. *Convolutional Neural Network (CNN) Tutorial*. 2020. On Kaggle. Disponível em: <<https://www.kaggle.com/code/kanncaa1/convolutional-neural-network-cnn-tutorial>>. Citado na página 51.
- DATASCIENCE, R. S. *Understanding Auto-encoders*. 2024. On Kaggle. Disponível em: <<https://www.kaggle.com/code/rockystats/understanding-auto-encoders>>. Citado na página 51.
- DOCUMENTATION, S. *Surprise Documentation*. 2024. Disponível em: <<https://surprise.readthedocs.io/en/latest/index.html>>. Citado na página 54.

- DOCUMENTATION, T. *Tensorflow Documentation*. 2024. Disponível em: <<https://www.tensorflow.org/?hl=pt-br>>. Citado na página 54.
- DOCUMENTATION, W. *Wurlitzer Documentation*. 2024. Disponível em: <<https://pypi.org/project/wurlitzer/>>. Citado na página 54.
- ENGEL, G. I. Pesquisa-ação. *Educar em Revista*, SciELO Brasil, p. 181–191, 2000. Citado 2 vezes nas páginas 56 e 59.
- ESPERIDIAO, M. Avaliação de satisfação de usuários. *Cadernos de Saúde Pública*, v. 21, n. Suplemento 1, 2005. Disponível em: <<https://www.scielo.org/pdf/csc/2005.v10suppl0/303-312/pt>>. Citado 2 vezes nas páginas 29 e 30.
- FACE, H. *Hugging Face - AI & NLP Community*. 2024. Disponível em: <<https://huggingface.co/>>. Citado 2 vezes nas páginas 52 e 54.
- FORMS, G. *Google Forms*. 2024. Disponível em: <<https://workspace.google.com/intl/pt-BR/products/forms/>>. Citado na página 54.
- FOUNDATION, P. S. *Python Programming Language*. 2024. Disponível em: <<https://www.python.org/>>. Citado 2 vezes nas páginas 52 e 54.
- GERHARDT, T. E.; SILVEIRA, D. T. *Métodos de pesquisa*. [S.l.]: Plageder, 2009. Citado na página 55.
- GIL, A. C. *Como Elaborar Projetos de Pesquisa*. 5. ed. [S.l.]: Editora Atlas, 2010. Citado na página 55.
- GIT. *Git*. 2024. Disponível em: <<https://git-scm.com/>>. Citado na página 54.
- GITHUB, I. *GitHub: Where the world builds software*. 2024. Disponível em: <<https://github.com/>>. Citado 2 vezes nas páginas 53 e 54.
- GOOGLE. *Google Colaboratory*. 2024. Disponível em: <<https://colab.research.google.com/>>. Citado 3 vezes nas páginas 52, 54 e 74.
- GRIFFITHS, J. R.; JOHNSON, F.; HARTLEY, R. J. User satisfaction as a measure of system performance. *Journal of Librarianship and Information Science*, v. 39, n. 3, p. 142–152, 2007. Disponível em: <<https://doi.org/10.1177/0961000607080417>>. Citado na página 47.
- GUNAWARDANA, A.; SHANI, G.; YOGEV, S. Evaluating recommender systems. In: _____. *Recommender Systems Handbook*. New York, NY: Springer US, 2022. ISBN 978-1-0716-2197-4. Disponível em: <https://doi.org/10.1007/978-1-0716-2197-4_15>. Citado 2 vezes nas páginas 25 e 28.
- HASAN, M. R.; JHA, A. K.; LIU, Y. Excessive use of online video streaming services: Impact of recommender system use, psychological factors, and motives. *Computers in Human Behavior*, v. 80, p. 220–228, 2018. ISSN 0747-5632. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0747563217306581>>. Citado na página 73.
- HEIDARI, N.; MORADI, P.; KOOCHARI, A. An attention-based deep learning method for solving the cold-start and sparsity issues of recommender systems. *Knowledge-Based Systems*, v. 256, 2022. ISSN 0950-7051. Disponível em: <<https://www.sciencedirect.com/science/article/pii/S0950705122009339>>. Citado na página 27.

- ISO. *ISO FDIS 9241-210 (2009) Human-centred design process for interactive systems*. 2009. Standard. Citado na página 47.
- JACOBSON, K. et al. Music personalization at spotify. In: *Proceedings of the 10th ACM Conference on Recommender Systems*. New York, NY, USA: Association for Computing Machinery, 2016. (RecSys '16). ISBN 9781450340359. Disponível em: <<https://doi.org/10.1145/2959100.2959120>>. Citado na página 34.
- JAIN, S. et al. Trends, problems and solutions of recommender system. In: *International Conference on Computing, Communication & Automation*. [S.l.: s.n.], 2015. p. 955–958. Citado na página 73.
- JUNG, K.-Y.; PARK, D.-H.; LEE, J.-H. Hybrid collaborative filtering and content-based filtering for improved recommender system. In: BUBAK, M. et al. (Ed.). *Computational Science - ICCS 2004*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004. p. 295–302. ISBN 978-3-540-24685-5. Citado na página 36.
- KAGGLE. *Kaggle: Your Machine Learning and Data Science Community*. 2024. Disponível em: <<https://www.kaggle.com/>>. Citado 2 vezes nas páginas 51 e 54.
- KARATZOGLOU, A.; HIDASI, B. Deep learning for recommender systems. In: *Proceedings of the Eleventh ACM Conference on Recommender Systems*. New York, NY, USA: Association for Computing Machinery, 2017. (RecSys '17). ISBN 9781450346528. Disponível em: <<https://doi.org/10.1145/3109859.3109933>>. Citado na página 28.
- KHUSRO, S.; ALI, Z.; ULLAH, I. Recommender systems: Issues, challenges, and research opportunities. In: KIM, K. J.; JOUKOV, N. (Ed.). *Information Science and Applications (ICISA) 2016*. Singapore: Springer Singapore, 2016. ISBN 978-981-10-0557-2. Citado 3 vezes nas páginas 25, 29 e 58.
- KINGMA, D. P. et al. Improving variational inference with inverse autoregressive flow. In: *Advances in Neural Information Processing Systems*. [S.l.: s.n.], 2016. Citado na página 28.
- KOMPAN, M. et al. Exploring customer price preference and product profit role in recommender systems. *IEEE Intelligent Systems*, v. 37, n. 1, p. 89–98, 2022. Citado na página 28.
- KONSTAN, J. A.; RIEDL, J. *DECONSTRUCTING RECOMMENDER SYSTEMS*. 2012. Disponível em: <<https://spectrum.ieee.org/deconstructing-recommender-systems>>. Citado na página 28.
- LAKEHOUSE, D. *Perspectiva de dados + IA 2023*. 2023. Disponível em: <https://www.databricks.com/br/resources/ebook/state-of-data-ai?scid=7018Y000001Fi0cQAC&utm_medium=paid+search&utm_source=google&utm_campaign=20613856686&utm_adgroup=152642604685&utm_content=ebook&utm_offer=state-of-data-ai&utm_ad=676095019567&utm_term=ia%20databricks&gad_source=1&gclid=CjwKCAiAi6uvBhADEiwAWiyRdhu4vCcRtEYsvjz_adMw9KeOINlakog84AuYZQL8vwEI0NozB1M97hoC2boQAvD_BwE>. Citado 2 vezes nas páginas 26 e 29.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *Nature*, v. 521, n. 7553, May 2015. ISSN 1476-4687. Disponível em: <<https://doi.org/10.1038/nature14539>>. Citado na página 43.

LOMONACO, V. Continual learning with deep architectures. alma, 2019. Citado na página 28.

LOURENÇO, S. S.; KRAKAUER, P. V. de C. O uso de plataformas de streaming durante a pandemia pelo covid-19. *Revista Fatec Sebrae em debate-gestão, tecnologias e negócios*, v. 8, n. 15, p. 01–01, 2021. Citado na página 73.

LUCIDCHART. *Lucidchart*. 2024. Disponível em: <<https://www.lucidchart.com/pages/?>> Citado na página 54.

MAHMOOD, M. A. et al. Variables affecting information technology end-user satisfaction: A meta-analysis of the empirical literature. *International Journal of Human-Computer Studies*, v. 52, n. 5, 2000. Citado 4 vezes nas páginas 47, 48, 58 e 59.

MINAYO, M. C. d. S.; SANCHES, O. Quantitativo - qualitativo: Oposição ou complementaridade? *Cadernos de Saúde Pública*, v. 9, n. 3, 1993. Citado na página 30.

MISHRA, N. et al. Research problems in recommender systems. *Journal of Physics: Conference Series*, IOP Publishing, v. 1717, n. 1, jan 2021. Disponível em: <<https://dx.doi.org/10.1088/1742-6596/1717/1/012002>>. Citado na página 28.

MUSIB, L. Q. *Restricted Boltzmann Machines (RBMs)*. 2023. On Kaggle. Disponível em: <<https://www.kaggle.com/code/leilahasan/restricted-boltzmann-machines-rbms>>. Citado na página 51.

NASCIMENTO, F. P. d.; SOUSA, F. L. L. Classificação da pesquisa. natureza, método ou abordagem metodológica, objetivos e procedimentos. *Metodologia da Pesquisa Científica: teoria e prática—como elaborar TCC*. Brasília: Thesaurus, 2016. Citado na página 56.

NVIDIA. *Recommendation System*. 2024. Website. Retrieved from <<https://www.nvidia.com/en-us/glossary/recommendation-system/>>. Citado na página 36.

PAJORSKA, Z. *Guide to Recommendation System*. 2023. Stratoflow. Disponível em: <<https://stratoflow.com/guide-to-recommendation-system/>>. Citado 3 vezes nas páginas 34, 35 e 58.

PARASHAR, M. *Movie Recommendation System*. 2023. On Kaggle. Disponível em: <<https://www.kaggle.com/datasets/parasharmanas/movie-recommendation-system>>. Citado 2 vezes nas páginas 52 e 75.

PATHAK, K. *Movie Recommendation System for Deployment*. 2020. On Kaggle. Disponível em: <<https://www.kaggle.com/code/terminate9298/movie-recommendation-system-for-deployment#Content-Based-Recommandation-System>>. Citado na página 51.

PEREIRA, P.; TORREÃO, P.; MARÇAL, A. S. Entendendo scrum para gerenciar projetos de forma ágil. *Mundo PM*, v. 1, n. 14, p. 64–71, 2007. Citado na página 57.

PHAM, K. *What are Recommendation Systems?* 2019. Disponível em: <<https://medium.com/@khang.pham.exxact/what-are-recommendation-systems-6bb5036042db>>. Citado 3 vezes nas páginas 33, 34 e 58.

RODRIGUES, V. Conhecendo a visão do computador: Redes neurais convolucionais. *Medium*, 2019. Acesso em: 29 jan. 2025. Disponível em: <<https://vitorborbarodrigues.medium.com/conhecendo-a-vis%C3%A3o-do-computador-redes-neurais-convolucionais-e1c2b14bf426>>. Citado 2 vezes nas páginas 45 e 46.

SHEETS, G. *Google Sheets*. 2024. Disponível em: <<https://workspace.google.com/intl/pt-BR/products/sheets/>>. Citado na página 54.

SISI, F. N. E. et al. A review for recommender system models and deep learning. *International Journal of Computer and Information*, Faculty of Computers and Information, Menoufia University, v. 14, n. 1, 2020. Citado 3 vezes nas páginas 43, 44 e 58.

SULEIMENOV, I. E. et al. Artificial intelligence: what is it? In: *Proceedings of the 2020 6th International Conference on Computer and Technology Applications*. New York, NY, USA: Association for Computing Machinery, 2020. (ICCTA '20). ISBN 9781450377492. Disponível em: <<https://doi.org/10.1145/3397125.3397141>>. Citado 2 vezes nas páginas 26 e 34.

TEAM, T. L. P. *LaTeX - A document preparation system*. 2024. Disponível em: <<https://www.latex-project.org/>>. Citado 2 vezes nas páginas 53 e 54.

THORAT, P. B.; GOUDAR, R. M.; BARVE, S. Survey on collaborative filtering, content-based filtering and hybrid recommendation system. *International Journal of Computer Applications*, Foundation of Computer Science, v. 110, n. 4, 2015. Citado 5 vezes nas páginas 36, 41, 42, 43 e 58.

WANG, W.; LU, Y. Analysis of the mean absolute error (mae) and the root mean square error (rmse) in assessing rounding model. In: IOP PUBLISHING. *IOP conference series: materials science and engineering*. [S.l.], 2018. v. 324, p. 012049. Citado na página 62.

YAN, F. et al. Performance modeling and scalability optimization of distributed deep learning systems. In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. New York, NY, USA: Association for Computing Machinery, 2015. (KDD '15). ISBN 9781450336642. Disponível em: <<https://doi.org/10.1145/2783258.2783270>>. Citado na página 27.

ZHANG, G.; LIU, Y.; JIN, X. A survey of autoencoder-based recommender systems. *Frontiers of Computer Science*, v. 14, n. 3, April 2020. Received 12 February 2018, Accepted 09 November 2018, Published 30 August 2019. Citado na página 45.

ZHANG, S. et al. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv.*, Association for Computing Machinery, New York, NY, USA, v. 52, n. 1, feb 2019. ISSN 0360-0300. Disponível em: <<https://doi.org/10.1145/3285029>>. Citado 2 vezes nas páginas 25 e 29.

ZVIRAN, M.; ERLICH, Z. Measuring is user satisfaction: review and implications. *Communications of the Association for Information Systems*, v. 12, n. 1, 2003. Citado 2 vezes nas páginas 47 e 58.

Apêndices

APÊNDICE A – Termo de Consentimento Livre e Esclarecido

Figura 69 – Termo de Consentimento Livre e Esclarecido

Satisfação de Usuário em um Sistema de Recomendação Centrado em Deep Learning

Este formulário busca analisar as recomendações geradas pelo modelo de Sistema de Recomendação centrado em Deep Learning. Serão realizadas três etapas: preferência de filmes, avaliação da primeira recomendação e avaliação da recomendação final. Sendo essa a primeira etapa, a qual é dividida em duas seções: Dados Pessoais e Preferência de Gênero de Filmes. Sua participação é fundamental para entendermos o comportamento do modelo e a satisfação dos usuários que o utilizaram.

* Indica uma pergunta obrigatória

Você autoriza o uso das suas respostas para pesquisa acadêmica, com a garantia de que as informações serão tratadas de forma confidencial e usadas apenas para fins acadêmicos e de pesquisa? *

☐ Sim

☐ Não

Fonte: Autora

APÊNDICE B – Primeiro Formulário - Coleta de Dados do Usuário

Figura 70 – Coleta de Dados - Exemplo - Página 1

Seção 1 de 3

Satisfação de Usuário em um Sistema de Recomendação Centrado em Deep Learning

Este formulário busca analisar as recomendações geradas pelo modelo de Sistema de Recomendação centrado em Deep Learning. Serão realizadas três etapas: preferência de filmes, avaliação da primeira recomendação e avaliação da recomendação final.

Sendo essa a primeira etapa, a qual é dividida em duas seções: Dados Pessoais e Preferência de Gênero de Filmes. Sua participação é fundamental para entendermos o comportamento do modelo e a satisfação dos usuários que o utilizaram.

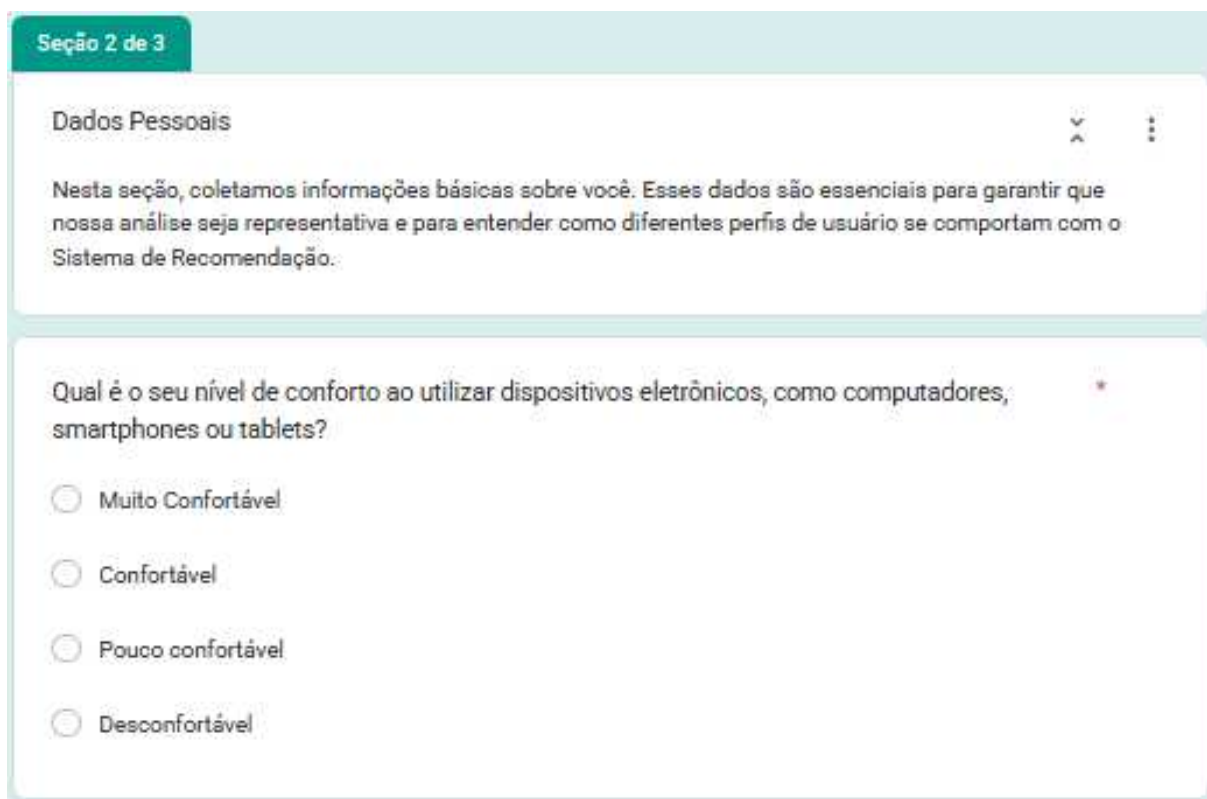
Você autoriza o uso das suas respostas para pesquisa acadêmica, com a garantia de que as informações serão tratadas de forma confidencial e usadas apenas para fins acadêmicos e de pesquisa? *

☐ Sim

☐ Não

Fonte: Autora

Figura 71 – Coleta de Dados - Exemplo - Página 2



Seção 2 de 3

Dados Pessoais

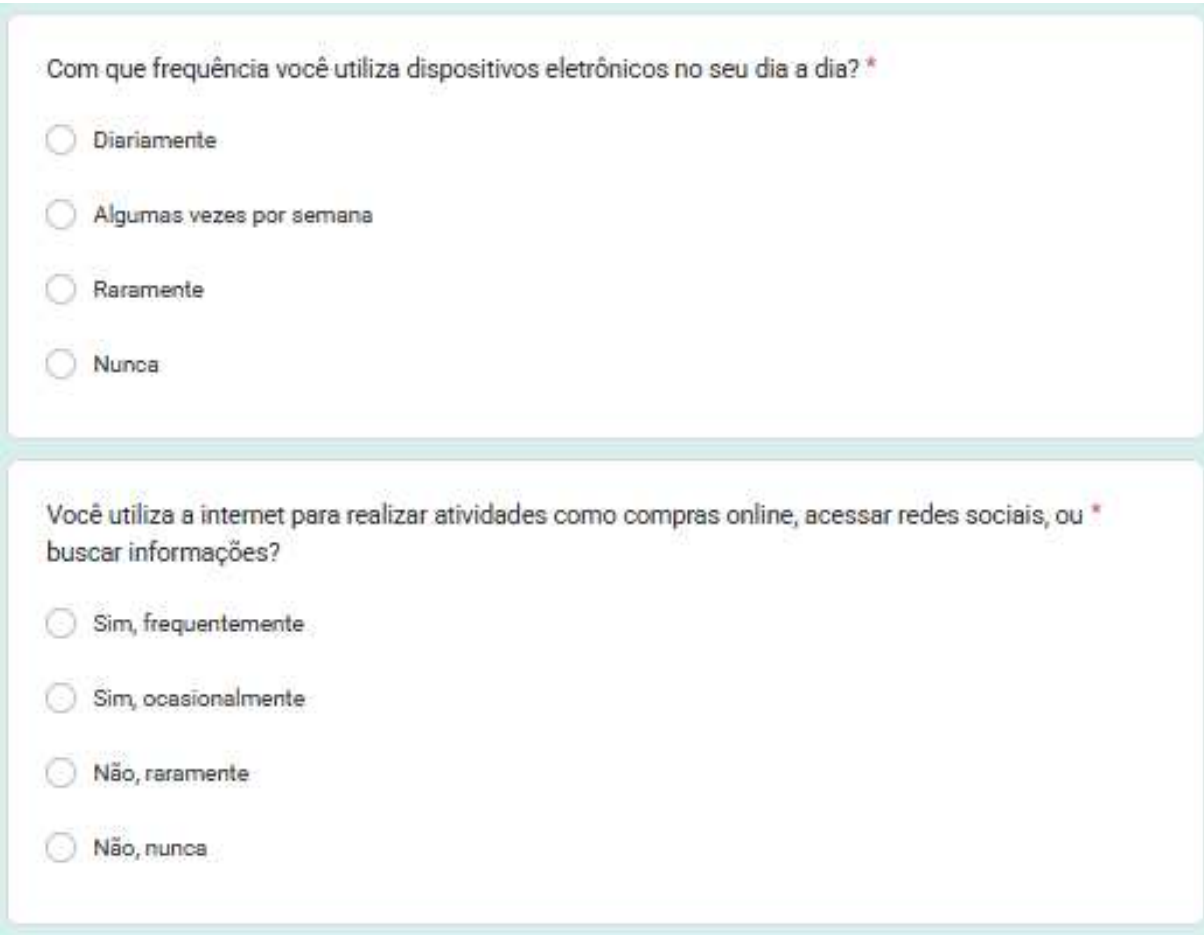
Nesta seção, coletamos informações básicas sobre você. Esses dados são essenciais para garantir que nossa análise seja representativa e para entender como diferentes perfis de usuário se comportam com o Sistema de Recomendação.

Qual é o seu nível de conforto ao utilizar dispositivos eletrônicos, como computadores, smartphones ou tablets?

- ☐ Muito Confortável
- ☐ Confortável
- ☐ Pouco confortável
- ☐ Desconfortável

Fonte: Autora

Figura 72 – Coleta de Dados - Exemplo - Página 3



Com que frequência você utiliza dispositivos eletrônicos no seu dia a dia? *

- ☐ Diariamente
- ☐ Algumas vezes por semana
- ☐ Raramente
- ☐ Nunca

Você utiliza a internet para realizar atividades como compras online, acessar redes sociais, ou buscar informações? *

- ☐ Sim, frequentemente
- ☐ Sim, ocasionalmente
- ☐ Não, raramente
- ☐ Não, nunca

Fonte: Autora

Figura 73 – Coleta de Dados - Exemplo - Página 4

Você consegue localizar e utilizar recursos em um site (como preencher formulários ou navegar entre páginas) sem instruções detalhadas? *

☐ Sim, facilmente

☐ Sim, com algum esforço

☐ Não, com dificuldade

☐ Não sem ajuda

Como você classificaria sua dificuldade para escolher um filme para assistir? *

☐ Facilmente

☐ Com algum esforço

☐ Com dificuldade

☐ Com muita dificuldade

Fonte: Autora

Figura 74 – Coleta de Dados - Exemplo - Página 5

Com que frequência você busca recomendações de filmes antes de assistir?

☐ Frequentemente

☐ Ocasionalmente

☐ Raramente

☐ Nunca

Qual seu nome? *

Texto de resposta curta

Fonte: Autora

Figura 75 – Coleta de Dados - Exemplo - Página 6

Seção 3 de 3

Preferência de Gêneros de Filmes

⌵

⋮

Nessa seção, coletaremos informações sobre quais são seus gêneros preferidos de filmes. Esse dado será utilizado para gerar as primeiras recomendações.

Escolha de 3 a 5 gêneros de sua preferência *

☐ Comédia

☐ Crime

☐ Drama

☐ Suspense

☐ Guerra

☐ Musical

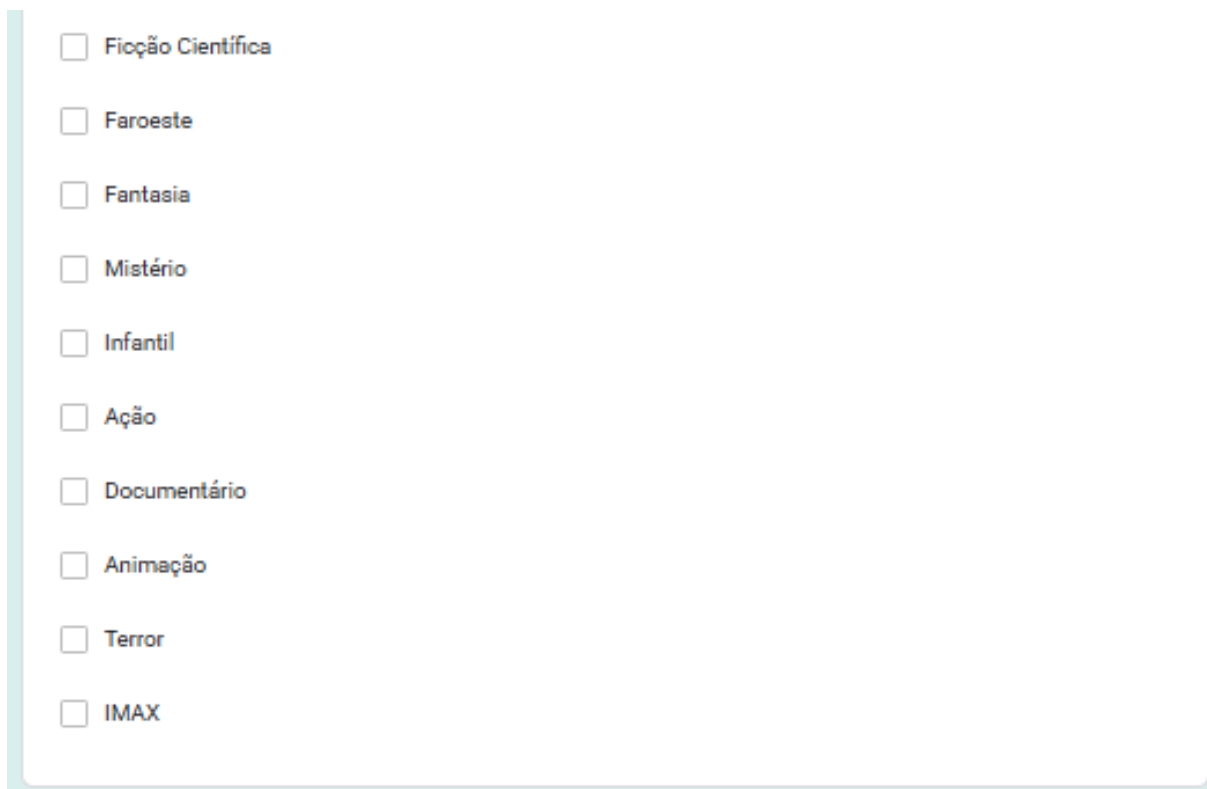
☐ Romance

☐ Aventura

☐ Filme Noir

Fonte: Autora

Figura 76 – Coleta de Dados - Exemplo - Página 7



Formulário de coleta de dados com uma lista de opções para seleção. Cada opção é precedida por uma caixa de seleção vazia.

- ☐ Ficção Científica
- ☐ Faroeste
- ☐ Fantasia
- ☐ Mistério
- ☐ Infantil
- ☐ Ação
- ☐ Documentário
- ☐ Animação
- ☐ Terror
- ☐ IMAX

Fonte: Autora

APÊNDICE C – Segundo Formulário - Avaliação da Primeira Recomendação

Figura 77 – Avaliação da Primeira Recomendação - Exemplo - Página 1

Satisfação de Usuário em um Sistema de Recomendação Centrado em Deep Learning

Este formulário busca analisar as recomendações geradas pelo modelo de Sistema de Recomendação centrado em Deep Learning. Serão realizadas três etapas: preferência de filmes, avaliação da primeira recomendação e avaliação da recomendação final. Sendo essa a segunda etapa, a qual é possui a seção de Avaliação Objetiva das Recomendações, com perguntas objetivas. Sua participação é fundamental para entendermos o comportamento do modelo e a satisfação dos usuários que o utilizaram. Nessa segunda etapa, as recomendações são mais imprecisas e servem principalmente pro sistema conhecer o usuário.

* Indica uma pergunta obrigatória

Você autoriza o uso das suas respostas para pesquisa acadêmica, com a garantia de que as informações serão tratadas de forma confidencial e usadas apenas para fins acadêmicos e de pesquisa? *

☐ Sim

☐ Não

Fonte: Autora

Figura 78 – Avaliação da Primeira Recomendação - Exemplo - Página 2

Qual seu nome? *

Escolher

Fonte: Autora

Figura 79 – Avaliação da Primeira Recomendação - Exemplo - Página 3


Nessa seção, coletaremos informações objetivas sobre as redomendações que foram sugeridas para você durante o uso da aplicação. Esses dados serão utilizados para verificar a precisão do modelo e treiná-lo novamente.

Fonte: Autora

Figura 80 – Avaliação da Primeira Recomendação - Exemplo - Página 4

Sobre o filme The Iron Giant (1999). Já assistiu?

Se sim, qual a nota que você dá ao filme The Iron Giant (1999)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 81 – Avaliação da Primeira Recomendação - Exemplo - Página 5

The Iron Giant (1999) - Trailer em inglês

Sinopse: Em 1957, um robô alienígena gigante aterrissa perto da pequena cidade de Rockwell, Maine. Hogarth, um garoto de nove anos que estava explorando a área, encontra o robô e os dois ficam amigos. Mas um agente do governo completamente obcecado surge com o objetivo de destruir o extraterrestre a qualquer custo.

A screenshot of a YouTube video player. The video title is 'The Iron Giant: Signatu...' and it features the Warner Bros. logo. The video thumbnail shows the head of the Iron Giant robot against a sunset background. A red play button is centered over the thumbnail.

Fonte: Autora

Figura 82 – Avaliação da Primeira Recomendação - Exemplo - Página 6

Se não, qual seu interesse em ver o filme The Iron Giant (1999)?



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 83 – Avaliação da Primeira Recomendação - Exemplo - Página 7

Sobre o filme Lost Highway (1997). Já assistiu?

Se sim, qual a nota que você dá ao filme Lost Highway (1997)?



0 1 2 3 4 5


Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 84 – Avaliação da Primeira Recomendação - Exemplo - Página 8

Lost Highway (1997) - Trailer em inglês

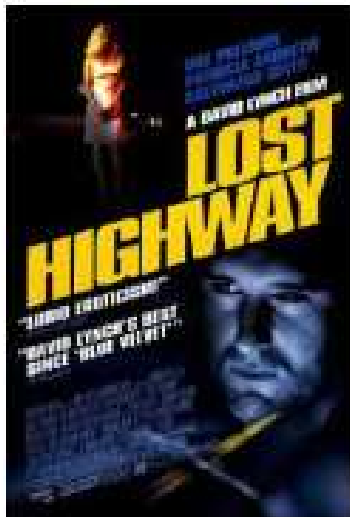
Sinopse: Um músico é acusado de assassinar sua esposa em casa. Ele sabe que não foi ele, mas ao ver as fitas da câmera de segurança, sua opinião muda, e não consegue entender o que aconteceu até descobrir a verdade.



Fonte: Autora

Figura 85 – Avaliação da Primeira Recomendação - Exemplo - Página 9

Se não, qual seu interesse em ver o filme *Lost Highway* (1997)?



0 1 2 3 4 5

Nenhum Interesse

☐☐☐☐☐☐


Muito Interessado

Fonte: Autora

Figura 86 – Avaliação da Primeira Recomendação - Exemplo - Página 10

Sobre o filme A Matter of Time (1976). Já assistiu?

Se sim, qual a nota que você dá ao filme A Matter of Time (1976)?



0 1 2 3 4 5


Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 87 – Avaliação da Primeira Recomendação - Exemplo - Página 11

Se não, qual seu interesse em ver o filme *A Matter of Time* (1976)?

Sinopse: A história fictícia é vagamente baseada nas façanhas reais da infame excêntrica italiana, a Marchesa Casati, a quem Druon conheceu durante seus anos finais em Londres, enquanto ele estava lá durante a Segunda Guerra Mundial.



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 88 – Avaliação da Primeira Recomendação - Exemplo - Página 12

Sobre o filme Ghost (2010). Já assistiu?

Se sim, qual a nota que você dá ao filme Ghost (2010)?



0 1 2 3 4 5


Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 89 – Avaliação da Primeira Recomendação - Exemplo - Página 13

Ghost (2010) - Trailer em inglês

Sinopse: A trama gira em torno de Sam Wheat, um bancário assassinado, cujo espírito permanece na terra para proteger sua namorada, Molly Jensen, das mesmas pessoas que o mataram - e para isso conta com a ajuda de Oda Mae Brown, uma excêntrica médium capaz de ouvir sua voz.



Fonte: Autora

Figura 90 – Avaliação da Primeira Recomendação - Exemplo - Página 14

Se não, qual seu interesse em ver o filme Ghost (2010)?



0 1 2 3 4 5

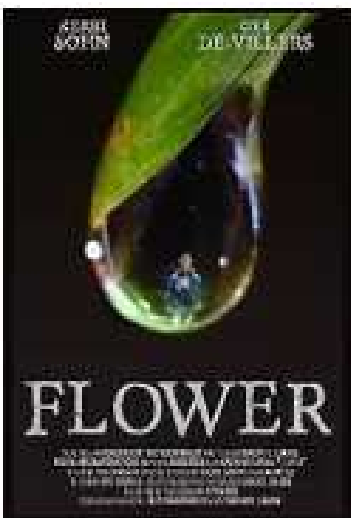
Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 91 – Avaliação da Primeira Recomendação - Exemplo - Página 15

Sobre o filme The Flower (2019). Já assistiu?

Se sim, qual a nota que você dá ao filme The Flower (2019)?



0 1 2 3 4 5

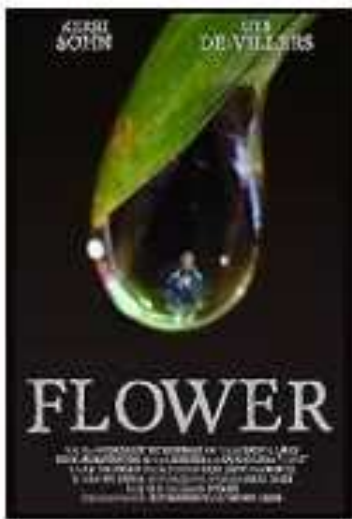
Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 92 – Avaliação da Primeira Recomendação - Exemplo - Página 16

Se não, qual seu interesse em ver o filme The Flower (2019)?

Sinopse: Uma zumbi encontra uma flor que a traz ao limiar da consciência, apenas para ser eliminada por um sobrevivente sem coração.



0 1 2 3 4 5

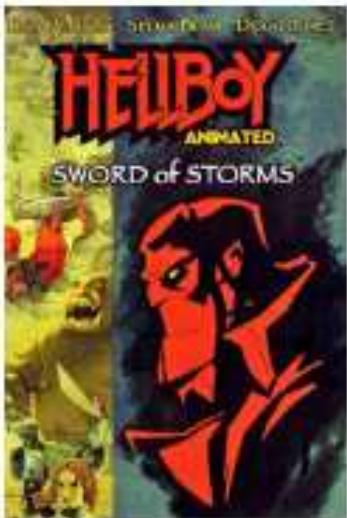
Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 93 – Avaliação da Primeira Recomendação - Exemplo - Página 17

Sobre o filme Hellboy Animated: Sword of Storms (2006). Já assistiu?

Se sim, qual a nota que você dá ao filme Hellboy Animated: Sword of Storms (2006)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora


Figura 94 – Avaliação da Primeira Recomendação - Exemplo - Página 18



Fonte: Autora

Figura 95 – Avaliação da Primeira Recomendação - Exemplo - Página 19

Se não, qual seu interesse em ver o filme Hellboy Animated: Sword of Storms (2006)?



0 1 2 3 4 5


Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 96 – Avaliação da Primeira Recomendação - Exemplo - Página 20

Sobre o filme Princess Mononoke (Mononoke-hime) (1997). Já assistiu?

Se sim, qual a nota que você dá ao filme Princess Mononoke (Mononoke-hime) (1997)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 97 – Avaliação da Primeira Recomendação - Exemplo - Página 21

Princess Mononoke (Mononoke-hime) (1997) - Trailer em inglês

Sinopse: Um príncipe, em busca de uma cura, luta em uma guerra entre a mata e uma colônia mineira. Nesta aventura ele conhece Mononoke.



Fonte: Autora

Figura 98 – Avaliação da Primeira Recomendação - Exemplo - Página 22

Se não, qual seu interesse em ver o filme Princess Mononoke (Mononoke-hime) (1997)?




0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 99 – Avaliação da Primeira Recomendação - Exemplo - Página 23

Sobre o filme *The Seven Deadly Sins: Prisoners of the Sky* (2018). Já assistiu?
Se sim, qual a nota que você dá ao filme *The Seven Deadly Sins: Prisoners of the Sky* (2018)?



0 1 2 3 4 5


Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 100 – Avaliação da Primeira Recomendação - Exemplo - Página 24

The Seven Deadly Sins: Prisoners of the Sky (2018) - Trailer em inglês


Sinopse: Os sete pecados capitais ajudam o povo do céu a enfrentar um poderoso grupo de demônios determinado a ressuscitar uma besta demoníaca que está adormecida há mais de 3.000 anos.



Fonte: Autora

Figura 101 – Avaliação da Primeira Recomendação - Exemplo - Página 25

Se não, qual seu interesse em ver o filme The Seven Deadly Sins: Prisoners of the Sky (2018)?



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 102 – Avaliação da Primeira Recomendação - Exemplo - Página 26

Sobre o filme Pulse (2006). Já assistiu?

Se sim, qual a nota que você dá ao filme Pulse (2006)?



0 1 2 3 4 5


Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 103 – Avaliação da Primeira Recomendação - Exemplo - Página 27

Pulse (2006) - Clipe em inglês


Sinopse: Um grupo de estudantes decide investigar o estranho suicídio de uma colega. Alguns dias após sua morte, ela entra em contato através de e-mails, solicitando ajuda. É o início de uma série de contatos através da tecnologia existente, incluindo celulares e internet, que permite acesso a um mundo desconhecido e perigoso.

A screenshot of a YouTube video player interface. At the top left, there is a small profile picture and the text 'Pulse (2006) | Trailer L...'. In the center is a large red play button. Below the play button, the text 'TRAILER LEGENDADO' is displayed in a stylized, glowing font. The background of the video frame shows a dark, abstract, and somewhat blurry image.

Fonte: Autora

Figura 104 – Avaliação da Primeira Recomendação - Exemplo - Página 28

Se não, qual seu interesse em ver o filme Pulse(2006)?



There are some frequencies we were never meant to find

PULSE

2017.14

0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 105 – Avaliação da Primeira Recomendação - Exemplo - Página 29

Sobre o filme *Prairie Dog* (2015). Já assistiu?

Se sim, qual a nota que você dá ao filme *Prairie Dog* (2015)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 106 – Avaliação da Primeira Recomendação - Exemplo - Página 30

Prairie Dog (2015) - Trailer em inglês

Sinopse: Um xerife solitário e uma consultora ambiental visitante correm para resolver o mistério que assombra os vales do Condado de Brada.



Fonte: Autora

Figura 107 – Avaliação da Primeira Recomendação - Exemplo - Página 31

Se não, qual seu interesse em ver o filme *Prairie Dog* (2015)?



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

APÊNDICE D – Terceiro Formulário - Avaliação da Segunda Recomendação

Figura 108 – Avaliação da Segunda Recomendação - Exemplo - Página 1

Satisfação de Usuário em um Sistema de Recomendação Centrado em Deep Learning

Este formulário busca analisar as recomendações geradas pelo modelo de Sistema de Recomendação centrado em Deep Learning. Serão realizadas três etapas: preferência de filmes, avaliação da primeira recomendação e avaliação da recomendação final. Sendo essa a terceira etapa, a qual é possui a seção de Avaliação Objetiva das Recomendações e Avaliação Subjetiva das Recomendações, com perguntas objetivas e abertas para capturar uma ampla variedade de competências. Sua participação é fundamental para entendermos o comportamento do modelo e a satisfação dos usuários que o utilizaram.

* Indica uma pergunta obrigatória

Você autoriza o uso das suas respostas para pesquisa acadêmica, com a garantia de que as informações serão tratadas de forma confidencial e usadas apenas para fins acadêmicos e de pesquisa? *

☐ Sim

☐ Não

Fonte: Autora

Figura 109 – Avaliação da Segunda Recomendação - Exemplo - Página 2

Qual seu nome? *

Escolher ▾

Fonte: Autora

Figura 110 – Avaliação da Segunda Recomendação - Exemplo - Página 3


Nessa seção, coletaremos informações objetivas sobre as redomendações que foram sugeridas para você durante o uso da aplicação. Esses dados serão utilizados para verificar a precisão do modelo e treiná-lo novamente.

Fonte: Autora

Figura 111 – Avaliação da Segunda Recomendação - Exemplo - Página 4

Sobre o filme Paperman (2012). Já assistiu?

Se sim, qual a nota que você dá ao filme Paperman (2012)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 112 – Avaliação da Segunda Recomendação - Exemplo - Página 5



Fonte: Autora

Figura 113 – Avaliação da Segunda Recomendação - Exemplo - Página 6

Se não, qual seu interesse em ver o Paperman (2012)?



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 114 – Avaliação da Segunda Recomendação - Exemplo - Página 7

Sobre a série Black Mirror. Já assistiu?

Se sim, qual a nota que você dá à Black Mirror?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 115 – Avaliação da Segunda Recomendação - Exemplo - Página 8



Fonte: Autora

Figura 116 – Avaliação da Segunda Recomendação - Exemplo - Página 9

Se não, qual seu interesse em ver a serie Black Mirror?



0 1 2 3 4 5

Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 117 – Avaliação da Segunda Recomendação - Exemplo - Página 10

Sobre o filme Over the Garden Wall (2013). Já assistiu?

Se sim, qual a nota que você dá ao filme Over the Garden Wall (2013)?



0

1

2

3

4

5

Muito Ruim

☐☐☐☐☐☐

Muito Bom

Fonte: Autora

Figura 118 – Avaliação da Segunda Recomendação - Exemplo - Página 11

Over the Garden Wall (2013) - Trailer em inglês

Sinopse: Wirt e seu irmão Gregory caminham na floresta com Beatrice, um pássaro falante. Os três estão indo para a cidade grande em busca de um livro que, segundo a lenda, contém tudo o que foi esquecido. Quando ficam cansados, Greg encontra um carro feito de legumes. Seu dono é John Crops, um cantor romântico também feito de legumes. Em troca do carro, Wirt promete encontrar a alma gêmea de Crops. Assim, a turma dá início a uma jornada de descobertas e aprendizado.



Fonte: Autora

Figura 119 – Avaliação da Segunda Recomendação - Exemplo - Página 12

Se não, qual seu interesse em ver o Over the Garden Wall (2013)?



0 1 2 3 4 5

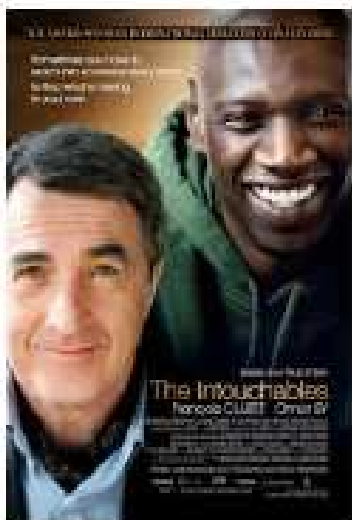
Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 120 – Avaliação da Segunda Recomendação - Exemplo - Página 13

Sobre o filme Intouchables (2011). Já assistiu?

Se sim, qual a nota que você dá ao filme Intouchables (2011)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora

Figura 121 – Avaliação da Segunda Recomendação - Exemplo - Página 14

Intouchables (2011) - Trailer

Sinopse: Um milionário tetraplégico contrata um homem da periferia para ser o seu acompanhante, apesar de sua aparente falta de preparo. No entanto, a relação que antes era profissional cresce e vira uma amizade que mudará a vida dos dois.



Fonte: Autora

Figura 122 – Avaliação da Segunda Recomendação - Exemplo - Página 15

Se não, qual seu interesse em ver o filme Intouchables (2011)?



0 1 2 3 4 5

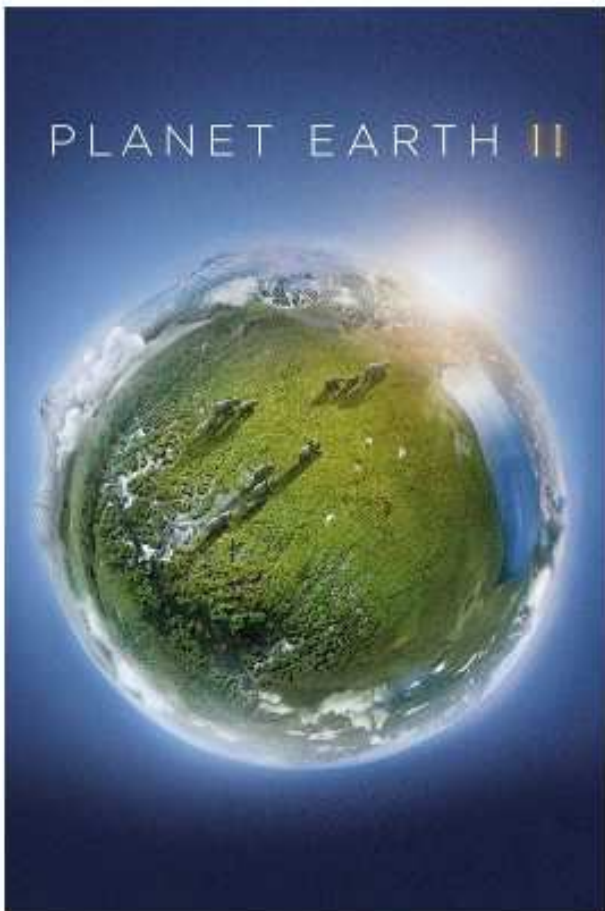
Nenhum Interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 123 – Avaliação da Segunda Recomendação - Exemplo - Página 16

Sobre o filme Planet Earth II (2016). Já assistiu?

Se sim, qual a nota que você dá ao filme Planet Earth II (2016)?



0 1 2 3 4 5

Muito Ruim ☐ ☐ ☐ ☐ ☐ ☐ Muito Bom

Fonte: Autora


Figura 124 – Avaliação da Segunda Recomendação - Exemplo - Página 17



Fonte: Autora

Figura 125 – Avaliação da Segunda Recomendação - Exemplo - Página 18

Se não, qual seu interesse em ver a série Planet Earth II (2016)?



PLANET EARTH II

0 1 2 3 4 5

Nenhum interesse ☐ ☐ ☐ ☐ ☐ ☐ Muito Interessado

Fonte: Autora

Figura 126 – Avaliação da Segunda Recomendação - Exemplo - Página 19

Avaliação Subjetiva do Sistema de Recomendação

Essa seção serve para coletar a sua opinião como usuário sobre as recomendações em geral. Esses dados serão utilizados para avaliar o desempenho do modelo e a satisfação do usuário com as recomendações.

Qual sua opinião geral sobre as recomendações que você recebeu até agora? *
Elas atenderam às suas expectativas? Explique.

Sua resposta

Fonte: Autora

Figura 127 – Avaliação da Segunda Recomendação - Exemplo - Página 20

Numa escala 0 a 10, qual nota você dá para sua satisfação geral com as recomendações? *

0 1 2 3 4 5 6 7 8 9 10

☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐ ☐

Quais melhorias você sugeriria para que as recomendações se tornem mais precisas ou mais personalizadas para o seu perfil? *

Sua resposta

Fonte: Autora