

PET WORLD:

Algoritmos de recomendações aplicados no mundo animal

GARCIA, Alisson Andrey da Silva¹

REIS, André da Mata¹

SANTOS, Matheus Gallo de Siqueira¹

JÚNIOR, Maurício Moreira²

¹Estudante de Graduação – Universidade José do Rosário Vellano UNIFENAS

²Docente de Graduação – Universidade José do Rosário Vellano UNIFENAS

RESUMO

Objetivo: No atual contexto do Brasil, há mais de setenta mil animais em situação de abandono, ou seja, sem um tutor responsável. ONGs trabalham constantemente para mudar essa realidade, porém seus meios são simples e com um alcance limitado. A tecnologia é algo primordial para auxiliar na mudança desse panorama. O objetivo deste trabalho foi desenvolver um aplicativo que utiliza um sistema de recomendação, ajudando pessoas a escolherem animais mais próximos ao seu gosto pessoal. Será desenvolvido uma API utilizando o framework Slim, contendo o algoritmo de recomendação, que utilizará a técnica de distância euclidiana e um aplicativo para dispositivos móveis, desenvolvido com a tecnologia React Native. Todos os testes serão realizados a partir de dados mocados, que será uma simulação de perfis de usuários e seus

gostos. A partir daí, as recomendações serão geradas conforme a utilização de cada perfil no aplicativo. A aplicação é capaz de sugerir animais para os usuários, de acordo com seus perfis e gostos pessoais. O aplicativo tem as funcionalidades de achados, perdidos e doações. Estas são as principais funcionalidades de navegação e, como funcionalidades essenciais, foi implementado no aplicativo os cadastros e edição tanto dos animais quanto dos usuários. O aplicativo tem se mostrado eficiente em resultados e testes parciais nas recomendações aos usuários simulados.

Palavras-chave

Recomendação, Algoritmos, Animais, Aplicativos

ABSTRACT

Purpose: In the current context of Brazil, there are more than seventy thousand animals in a situation of abandonment, that is, without a responsible guardian. NGOs are constantly working to change this reality, but their means are simple and limited in scope. Technology is essential to help change this scenario. The objective of this work was to develop an application that uses a recommendation system, helping people to choose animals closer to their personal taste. An API will be developed using the Slim framework, containing the recommendation algorithm, which will use the Euclidean distance technique and an application for mobile devices, developed with React Native technology. All tests will be carried out based on modified data, which will be a simulation of user profiles and their tastes. From there, the recommendations will be generated according to the use of each profile in the application. The application is able to suggest animals to users, according to their personal profiles and tastes. The application has the features of lost and found, and donations, these are the main navigation features, and as essential features, the registration and editing of both animals and users was implemented in the application. The application has been shown to be efficient in results and partial tests in recommendations to simulated users.

Keywords

Recommendation, Animals, Algorithms, App

1 INTRODUÇÃO

Em um cenário onde as pessoas têm cada vez mais acesso a produtos oferecidos por meio de lojas online, atingir o gosto dos clientes, de forma com que os mesmos fiquem mais satisfeitos com o produto ou serviço adquirido, se torna um diferencial positivo para as empresas que conseguem esse feito de forma eficaz. Sistemas de recomendação estão sendo amplamente utilizados para esse tipo de comércio que trabalha através da internet. Segundo Souza e Mailidú (2011), as funções de um Sistema de Recomendação, são analisar dados e extrair informações dos mesmos para que possam ser feitas as futuras filtragens, que resultam em recomendações para os usuários. Para a implementação de um Sistema de Recomendação, podem ser utilizadas diversas técnicas diferentes, dentre elas, as três principais são as de filtragem colaborativa, filtragem por conteúdo e para aumentar a qualidade da filtragem, existe a filtragem híbrida.

Segundo o Instituto Pet Brasil (2019), o país tem em torno de quarenta milhões de animais de estimação, aproximadamente cento e setenta e dois mil desses animais se encontram em situação de abandono, ou seja, não possuem um tutor definido. Grande parte dos animais abandonados acabam sendo acolhidos e tutelados por Organizações não Governamentais. No país, existem mais de trezentos e setenta ONGS que fazem o cuidado de animais abandonados, sendo quarenta e seis por cento dessas localizadas na região Sudeste, dezoito por cento na região Sul,

dezessete por cento na região Nordeste, doze por cento na região Norte e sete por cento na região Centro-Oeste. Um grande obstáculo na atuação das ONGs é a capacidade máxima de acolhimento. Dados levantados pelo Instituto Pet Brasil, indicam que as de pequeno porte comportam até cem animais, as de médio porte comportam de cento e um até quinhentos e as de grande porte acolhem mais de quinhentos e um animais.

É muito importante que haja um controle sobre os animais de estimação, tanto para cuidar da saúde deles quanto da população. Um dos principais objetivos das ONGs, além do acolhimento dos animais, é encaminhá-los para um adotante responsável para que o mesmo possa tutelar o animal. Esse encaminhamento muitas das vezes é divulgado através de redes sociais, e de formas locais, como por exemplo: carros de som, cartazes e eventos. As ONGs são mantidas por doações e trabalhos voluntários. Ferramentas tecnológicas, como os aplicativos móveis, podem ser uma solução para auxiliar na dificuldade de fazer o contato de um animal que esteja disponível para doação com um tutor para ser seu responsável.

Este trabalho tem como objetivo desenvolver um aplicativo para dispositivos móveis que poderá auxiliar a população geral a realizar buscas de animais de estimação que estejam para doação, utilizando de algoritmos de recomendação para a exibição de animais de acordo com o perfil do usuário.

2 MATERIAL E MÉTODOS

Neste tópico serão apresentadas todas as ferramentas, tecnologias e o método que foi utilizado na pesquisa.

2.1 Tecnologias e ferramentas

1. Segundo o site do React Native (2020), o próprio é um framework baseado no React (framework JavaScript para desenvolvimento web) que possibilita o desenvolvimento de aplicações mobile, tanto para Android, como para iOS, utilizando apenas Javascript. Com ele é possível criar aplicativos fluídos, com carregamentos rápidos, fácil integração com recursos nativos, desenvolvimento de aplicações para Android e iOS utilizando código único, entre outras vantagens.
2. “O PHP é uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML” (PHP, 2020). Com ele, é possível construir sites totalmente dinâmicos, como também APIs que podem ser consumidas por outras aplicações. O que distingue o PHP de algo como o JavaScript no lado do cliente é que o código é executado no servidor, gerando o HTML que é então enviado para o navegador.
3. Segundo Macário e Baldo (2005), banco de dados é um local onde dados são armazenados e gerenciados. A palavra "relacional" significa

que os dados armazenados estão organizados em tabelas. Quanto a sua definição, MySQL é um banco de dados relacional (RDBMS – *Relational Database Management Systems*) com um modelo de cliente-servidor, um software de código aberto ou serviço usado na criação e gerenciamento de bancos de dados baseados no modelo relacional. O serviço utiliza a linguagem SQL (*Structure Query Language* – Linguagem de Consulta Estruturada), que é a linguagem utilizada para inserir, acessar e gerenciar o conteúdo armazenado em um banco de dados.

4. De acordo com o site oficial do Firebase (2020), o mesmo é uma plataforma móvel do Google utilizada para facilitar o desenvolvimento de aplicativos web ou móveis, de uma forma efetiva, rápida e simples. Conta com um banco de dados não relacional, e com uma tecnologia de atualização em tempo real, ótimo para integração em aplicações que necessitam efetuar notificações ou procedimentos similares.
5. Conforme descrito por Diana e Gerosa (2010), um banco de dados não relacional é um banco de dados que não usa o esquema de tabela de linhas e colunas encontrado na maioria dos sistemas de banco de dados tradicionais. Em vez disso, os bancos de dados não relacionais usam um modelo de armazenamento otimizado para os requisitos específicos do tipo de dados que está sendo armazenado. Por

exemplo, os dados podem ser armazenados como pares chave/valor simples.

6. Segundo o site oficial Visual Studio Code (2020), a plataforma é um editor de código-fonte multiplataforma desenvolvido pela Microsoft, podendo ser utilizado a partir do Windows, MacOS ou Linux. O mesmo proporcionará um ambiente de trabalho que possibilita edição de todas as linguagens de programação presentes neste artigo.
7. De acordo com o site da Apache Software Foundation (2020), o Apache é um servidor de código aberto mantido pela mesma, ele permite que os sites mostrem e mantenham seus conteúdos na internet. Para chegar nesse objetivo, ele age como um mediador entre o servidor e as máquinas dos clientes. O mesmo puxa conteúdo de um servidor em cada pedido do cliente e realiza essa entrega na internet.
8. “Slim é um framework PHP que auxilia na criação de aplicações web e APIs simples, mas poderosas” (SLIM, 2020). Com ele é permitido criar aplicações seguras e robustas de maneira mais simples e com uma baixa complexidade se comparado a demais frameworks. Também possui características importantes, dentre elas facilidade na integração de aplicações externas ao projeto e suporte nativo a rotas HTTP.
9. Segundo o site oficial Lucidchart (2020), a plataforma é um espaço de trabalho visual que mescla diagramação, visualização de dados e colaboração para acelerar a compreensão. Com ele é possível a criação do Modelo

Entidade Relacionamento (MER), possibilitando a criação de um modelo de banco de dados que atenda às necessidades do projeto.

2.2 A Plataforma Desenvolvida

A plataforma consiste em um ecossistema de aplicações que trabalham interligadas, possibilitando a construção de um aplicativo móvel e de uma API.

Para a aplicação móvel, será utilizada a tecnologia React-Native, onde o usuário poderá se cadastrar e utilizar de acordo com sua necessidade. Para o desenvolvimento dos algoritmos de recomendação, utilizaremos a técnica de distância euclidiana, através da linguagem PHP. O aplicativo fará armazenamento de dados através de um banco de dados relacional, o mesmo será desenvolvido utilizando o Sistema de Gerenciamento de Banco de Dados MySQL, e utilizará um banco de dados não relacional para fazer envios de notificações em tempo real através da tecnologia Firebase.

As APIs permitem que a conectividade se torne fácil, rápida e conveniente e os processos cada vez mais automatizados. Com isso a aplicação utilizará dos seus benefícios, trazendo agilidade e eficácia nas integrações, permitindo que a troca de informações aconteça de forma mais segura e fácil, tanto para quem disponibiliza os dados quanto para quem utiliza. Será utilizada para a API a linguagem de programação PHP, fazendo uso do framework Slim.

2.3 Diagrama de Dados

O Modelo Entidade Relacionamento, criado utilizando a ferramenta Lucidchart, para que atendessem com base nas necessidades do projeto está referenciado na FIG. 1

2.4 Algoritmo de Recomendação

Considerando que o sistema fará uma recomendação híbrida, que segundo Burke (2002) seria a junção da recomendação baseada em conteúdo com a recomendação colaborativa, o mesmo utilizará a técnica de distância euclidiana onde são realizados cálculos para encontrar a similaridade entre as pessoas e/ou os animais.

2.5 Distância Euclidiana

A distância euclidiana é a distância entre dois pontos em um plano cartesiano. Quando há necessidade de encontrar a distância entre um ponto $P = (P_1, P_2, \dots, P_n)$ e $Q = (Q_1, Q_2, \dots, Q_n)$, em um espaço euclidiano n -dimensional. Como mostra Júnior et al. (2009), esse método consiste em provar que, quanto maior a distância entre dois pontos em um plano cartesiano, menor a possibilidade deles pertencerem a mesma referência. A fórmula está descrita na FIG. 2.

$$\sqrt{\sum_{i=1}^n (P_i - Q_i)^2}$$

Figura 2: Fórmula da Distância Euclidiana.

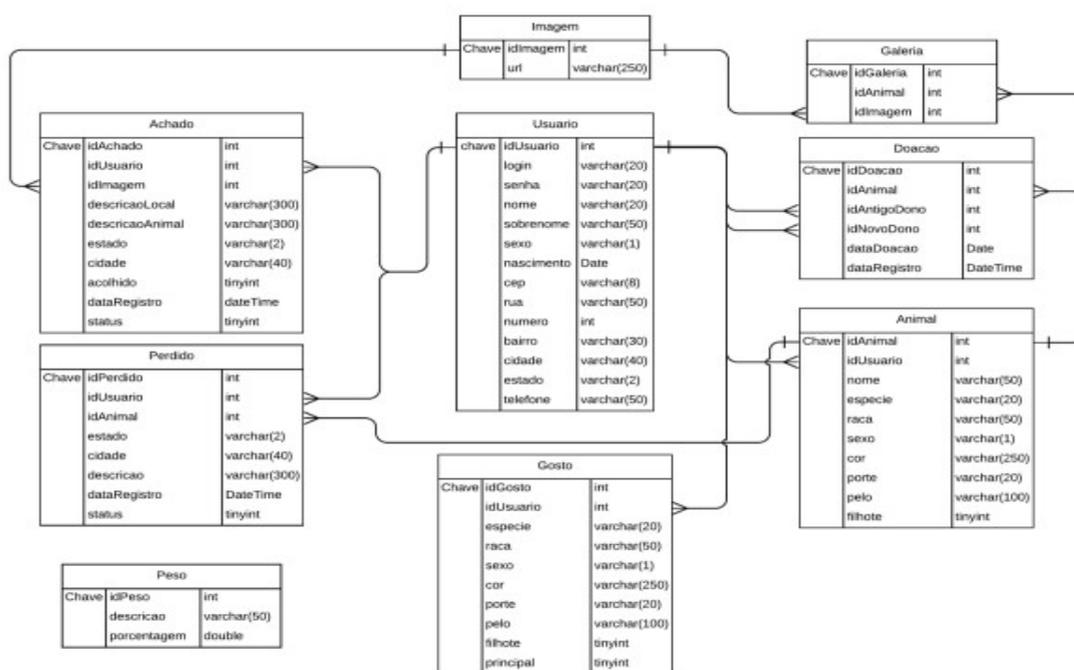


Figura 1: Diagrama de Dados

2.5.1 Distância Euclidiana Entre Pessoas

Inicialmente, foram definidos alguns parâmetros onde define-se o gosto relacionado a animais para que seja possível comparar duas pessoas, são eles: espécie, raça, sexo, cor, porte, tipo de pelo e filhote. Para cada parâmetro foi definido um peso, pois existem características que são mais relevantes que outras, portanto os pesos foram definidos com base na importância das características no momento de escolha de um animal, em detrimento de outras. As definições dos pesos estão referenciadas na Tabela 1. Estas características e seus respectivos foram definidos de acordo com que o algoritmo conseguisse

realizar uma boa recomendação. Assim, espécie e raça foram de maior valor.

Tabela 1: Definição dos Pesos

Características	Peso
Espécie	30%
Raça	30%
Sexo	6%
Cor	4%
Porte	20%
Pelo	4%
Filhote	6%

2.5.2 Distância Euclidiana Entre Animais

Seguindo a mesma lógica do item 2.5.1, para encontrarmos a distância entre dois animais, se faz necessário realizar a comparação de suas

características com outros animais cadastrados, também resultando em uma nota que define a similaridade entre eles.

2.6 Recomendação Colaborativa

Conforme mostra Takahashi (2015), esse modelo de filtragem vem sendo amplamente utilizado nos algoritmos de sites de comércio eletrônico em geral. Sua lógica consiste no seguinte modelo: existem dois perfis semelhantes de usuários, por exemplo, o usuário A interagiu com um certo item e o usuário B com outro item, a partir daí o algoritmo gera uma nota, partindo das semelhanças dos itens e dos usuários, recomendando assim os itens que ambos interagiram a partir das notas geradas pelo algoritmo, sempre recomendando para o usuário, os itens que possuem as maiores notas, e que o mesmo ainda não tenha interagido.

Com base na explicação dada anteriormente, o algoritmo de recomendação colaborativa pode ser definido seguindo os seguintes passos:

1. Selecionar uma pessoa no qual será efetuada a recomendação (Pessoa x);
2. Obter a similaridade desta pessoa com todas as outras cadastradas no banco de dados, ordenado de tal forma que a nota seja decrescente, ou seja, a lista de usuários ficará com a pessoa de maior nota em primeiro lugar;
3. Comparar o gosto da pessoa x com o gosto de cada pessoa da lista de similares obtido no

segundo passo, a fim de capturar as características que difere da pessoa x, armazenando o resultado em uma lista;

4. A partir da lista resultante do terceiro passo, é efetuado a distância euclidiana entre todos os animais (3.5.3), ordenando de tal forma que a nota seja decrescente, ou seja, a lista de animais ficará com o animal de maior nota em primeiro lugar;
5. Agora basta recomendar a lista resultante do quarto passo para a pessoa x, onde pode-se aplicar um filtro recomendando apenas os animais com nota maior que cinquenta por cento.

2.7 Recomendação Baseada em Conteúdo

Segundo Takahashi (2015) a filtragem baseada em conteúdo consiste em recomendar para o usuário itens que condizem com seu perfil cadastrado, a partir de itens já consumidos ou interagidos pelo mesmo. Essa lógica funciona da seguinte maneira, a primeiro momento é necessário que o usuário forneça o seu gosto pessoal, a partir dessas informações, o sistema irá recomendar itens semelhantes com os de seu gosto pessoal. Conforme o usuário utiliza o sistema, interagindo e avaliando os itens da plataforma, o algoritmo irá sempre recomendar itens semelhantes conforme suas avaliações e interações. Com base na explicação dada anteriormente, o algoritmo de recomendação baseado em conteúdo pode ser definido seguindo os seguintes passos:

1. Selecionar uma pessoa no qual será efetuada a recomendação;
2. Obter a lista de gostos dessa pessoa, no qual é obtida conforme ela vai interagindo com o sistema;
3. A partir das características obtidas das tabelas de gostos, é efetuado a distância euclidiana entre todos os animais (3.5.3), ordenando de tal forma que a nota seja decrescente, ou seja, a lista de animais ficará com o animal de maior nota em primeiro lugar;
4. Agora basta recomendar a lista resultante do terceiro passo, onde pode-se aplicar um filtro recomendando apenas os animais com nota maior que cinquenta por cento.

3 RESULTADOS E DISCUSSÃO

Foi desenvolvido um aplicativo para dispositivos móveis denominado PetWorld, que é capaz de efetuar recomendações para os usuários cadastrados no mesmo. Foi implementado em uma API os algoritmos de recomendação baseado no modelo híbrido, ou seja, utiliza-se mais de um método de recomendação, no caso deste, utiliza-se dois algoritmos distintos, um deles para fazer uma recomendação baseada em conteúdo, e o outro para fazer recomendações colaborativas. Além das recomendações, o aplicativo possui as seguintes funcionalidades: Achados, perdidos,

cadastro e edições de animais próprios, cadastro de usuários, edição de perfil, visualização dos animais com maiores detalhes, e permite efetuar adoções.

O usuário durante a realização do cadastro no aplicativo, em uma das quatro etapas necessárias, deverá preencher um formulário de gostos, de acordo com as características dos animais, anteriormente citadas na Tabela 1, conforme mostra a FIG. 3. Após o cadastro finalizado, os valores fornecidos pelo usuário nesta etapa, preencherão a tabela de gostos dele no banco de dados. A FIG. 3 está dividida em 3 telas para facilitar a identificação do cadastro.

Após a tabela de gostos ser preenchida, o aplicativo executará as recomendações utilizando os algoritmos de recomendação baseado em conteúdo e colaborativo, utilizando a fórmula de distância euclidiana com embasamento nos pesos de cada característica obtida, citados no tópico material e métodos.

A FIG. 4 mostra o resultado das recomendações para o usuário que acabou de preencher o formulário de gostos do aplicativo. A parte com o título “Recomendados para você”, se trata do resultado da execução do algoritmo de recomendação baseado em conteúdo, “Outras pessoas gostaram” se trata do método colaborativo.

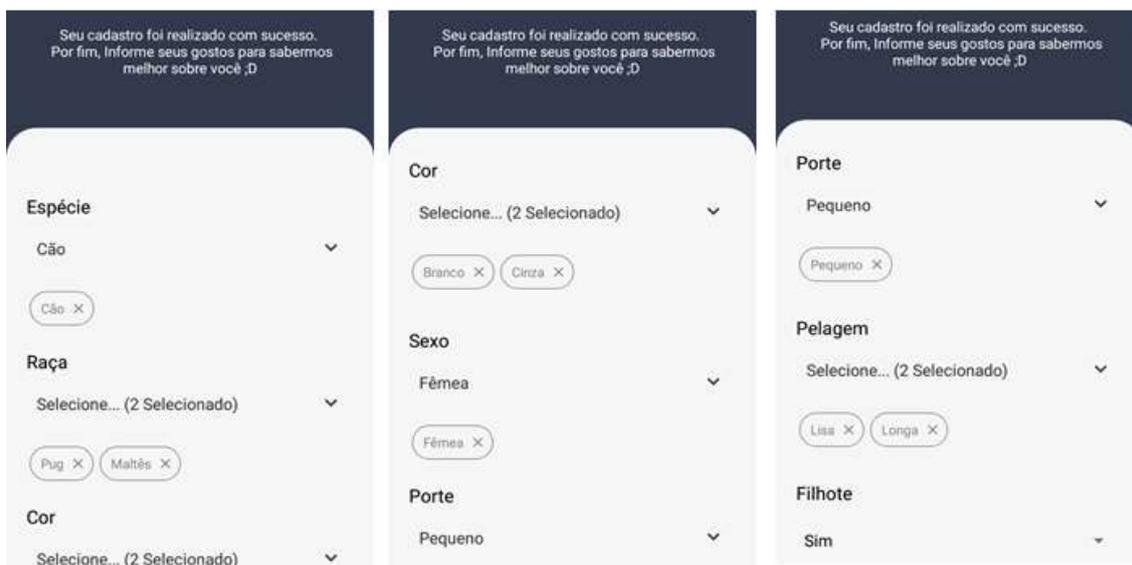


Figura 3: Cadastro de Gostos



Figura 4: Recomendações

A FIG. 5 demonstra a tabela de recomendações geradas a partir da tabela de gostos do usuário e sua nota gerada a partir da

execução do algoritmo de recomendação baseada em conteúdo.

Pode-se perceber que os três primeiros animais têm uma afinidade maior com os gostos do usuário, pois, a nota está em 79% o que vai de encontro com os dados fornecidos pelo usuário. A partir do quarto animal, percebe-se que os animais recomendados não possuem tanta afinidade, pois, a nota está em torno de 56% a 16%. Isso se deve à pequena base de animais cadastrados no aplicativo.

Já na FIG. 6 tem-se os resultados do algoritmo de recomendação colaborativo, no qual pode-se perceber que o campo de “Nota” para os três primeiros animais está abaixo de 70%. Isso ocorre devido a comparação com outros usuários no qual esse algoritmo utiliza.

Conteúdo								
Nome	Espécie	Raça	Sexo	Cor	Porte	Pelo	Filhote	Nota
Branquinha	Cachorro	Maltês	Feminino	Branco	Pequeno	Enrolada	Sim	79%
Princesa	Cachorro	Pug	Feminino	Cinza	Pequeno	Curta	Não	73%
Bolota	Cachorro	Pug	Feminino	Cinza	Pequeno	Curta	Não	73%
Jake	Cachorro	Spitz Alemão	Masculino	Marrom	Pequeno	Curta	Sim	56%
Thor	Cachorro	Buldogue	Masculino	Marrom	Médio	Lisa	Sim	38%
Magali	Cachorro	PitBull	Feminino	Bege	Médio	Curta	Não	36%
Iron	Cachorro	Buldogue	Masculino	Branco, Bege	Médio	Curta	Não	32%
Judy	Gato	Maine Coon	Feminino	Cinza	Grande	Lisa	Sim	16%
Flufy	Gato	Persa	Masculino	Bege	Médio	Longa	Não	2%
Garfield	Gato	Indefinida	Masculino	Bege	Médio	Longa	Não	2%

Figura 5: Notas Baseadas em Conteúdo

Colaborativa								
Nome	Espécie	Raça	Sexo	Cor	Porte	Pelo	Filhote	Nota
Flufy	Gato	Persa	Masculino	Bege	Médio	Longa	Não	66%
Garfield	Gato	Indefinida	Masculino	Bege	Médio	Longa	Não	66%
Magali	Cachorro	PitBull	Feminino	Bege	Médio	Curta	Não	64%
Black	Gato	Sphynx	Masculino	Preto	Grande	Curta	Não	46%
Iron	Cachorro	Buldogue	Masculino	Branco, Bege	Médio	Curta	Não	38%
Judy	Gato	Maine Coon	Feminino	Cinza	Grande	Lisa	Sim	30%
Thor	Cachorro	Buldogue	Masculino	Marrom	Médio	Lisa	Sim	26%
Princesa	Cachorro	Pug	Feminino	Cinza	Pequeno	Curta	Não	10%
Jake	Cachorro	Spitz Alemão	Masculino	Marrom	Pequeno	Curta	Sim	10%
Bolota	Cachorro	Pug	Feminino	Cinza	Pequeno	Curta	Não	10%

Figura 6: Notas Colaborativas

As recomendações são atualizadas semanalmente a partir dos interesses reconhecidos através da utilização do aplicativo, isso melhora a experiência do usuário no aplicativo, considerando que o gosto pessoal pode ser algo variável ou não. Pode-se notar que o aplicativo consegue realizar as recomendações, mesmo com uma base pequena de animais cadastrados. À medida que o mesmo for sendo utilizado, essa base aumentará e

consequentemente a eficiência dos resultados da recomendação será maior a cada nova interação.

4 CONCLUSÃO

Após o desenvolvimento do projeto, nota-se que o objetivo do mesmo foi alcançado, pois foi desenvolvido um aplicativo com funcionalidade de recomendar animais disponíveis para adoção de acordo com gostos pessoais do usuário

conectado e de outros usuários ativos na plataforma.

Uma sugestão para a melhoria do mecanismo que atualiza as recomendações, são estudos de técnicas de aprendizado de máquina, pois são utilizadas especificamente em funcionalidades parecidas ou iguais

REFERÊNCIAS

Apache Software Foundation. “Overview”. Disponível em: <Foundation Project (apache.org)>. Acesso em 12 de dezembro de 2020.

BALDO, Stefano Monteiro; MACÁRIO, Carla Geovana do N. O Modelo Relacional. Instituto de Computação Unicamp. Campinas, p. 1-15, Dez. 2005. Disponível em: <<https://www.ic.unicamp.br/~geovane/mo410-091/Ch03-RM-Resumo.pdf>>

BURKE, Robin. Hybrid Recommender Systems: Survey and Experiments. Researchgate. Fullerton, v.12, p. 1-29, Nov. 2002. Disponível em: <https://www.researchgate.net/publication/263377228_Hybrid_Recommender_Systems_Survey_and_Experiments>

CARVALHO JÚNIOR, Osmar Abílio de et al. Avaliação dos Classificadores Espectrais de Mínima Distância Euclidiana Espectral Correlation Mapper em Séries Temporais NDVI-MODIS no Campo de Instrução Militar de Formosa (GO). Revista Brasileira de Cartografia. [s.l.], v.61, n.4, p. 1-14, Dez. 2009. Disponível em: <<http://www.seer.ufu.br/index.php/revistabrasileiracartografia/article/view/43661>>

DIANA, Maurício de; GEROSA, Marco Aurélio; NOSQL na Web 2.0: Um Estudo Comparativo de Bancos Não-Relacionais para Armazenamento de Dados na Web 2.0. 2010. 8f.

Tese - SISTEMAS DE INFORMAÇÃO UFRPE, [s.l.], 2010. Disponível em: <sbdd_wtd_12.pdf>

Firestore. “Firestore”. Disponível em: <Firestore (google.com)>. Acesso em 12 de dezembro de 2020.

Instituto Pet Brasil. “País tem 3,9 milhões de animais em condição de vulnerabilidade”. Disponível em: <País tem 3,9 milhões de animais em condição de vulnerabilidade – Instituto Pet Brasil>. Acesso em 12 de dezembro de 2020.

Lucidchart. “Lucidchart”. Disponível em: <Software online de diagramas e comunicação visual | Lucidchart>. Acesso em 12 de dezembro de 2020.

MAILIDÚ, Ruy Luiz; SOUZA, Bruno Figueiredo Melo e. Modelos de Fatoração Matricial para Recomendação de Vídeos. 2011. 67f. Dissertação - Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2011.

PHP. “O que é o PHP”. Disponível em: <PHP: O que é o PHP? - Manual>. Acesso em 12 de dezembro de 2020.

React Native. “React Native”. Disponível em: <React Native · A framework for building native apps using React>. Acesso em 12 de dezembro de 2020

SANTOS, Daniele Silva; SILVA, Michelle Maia Paris. A Importância do Marketing Empresarial Para Micro e Pequenas Empresas. Facima Digital. [s.l.], p. 1, 2016. Disponível em: <revista_facima_ano_1_importancia_marketing_empresarial.pdf>

Slim. “Slim: um micro-framework PHP”. Disponível em: <Slim Framework - Slim Framework>. Acesso em 12 de dezembro de 2020.

TAKAHASHI, Marcos M.; Estudo comparativo de Algoritmos de Recomendação. 2015. 50f. Monografia - Instituto de Matemática e Estatística, [s.l.], 2015.

Visual Studio Code. “Visual Studio Code”. Disponível em: <Visual Studio Code - Code Editing. Redefined>. Acesso em 12 de dezembro de 2020.