UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

Mateus Cavalcante de Almeida Farias Aires

Voice Pathology Detector: Desenvolvimento de uma Aplicação Móvel para Detecção de Patologias de Voz

CAMPINA GRANDE - PB

2023

Mateus Cavalcante de Almeida Farias Aires

Voice Pathology Detector: Desenvolvimento de uma Aplicação Móvel para Detecção de Patologias de Voz

Trabalho de Conclusão de Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Herman Martins Gomes

CAMPINA GRANDE - PB

2023

Mateus Cavalcante de Almeida Farias Aires

Voice Pathology Detector: Desenvolvimento de uma Aplicação Móvel para Detecção de Patologias de Voz

Trabalho de Conclusão Curso apresentado ao Curso Bacharelado em Ciência da Computação do Centro de Engenharia Elétrica e Informática da Universidade Federal de Campina Grande, como requisito parcial para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA:

Professor Dr. Herman Martins Gomes

Orientador – UASC/CEEI/UFCG

Professor Dr. Wilkerson de Lucena Andrade

Examinador – UASC/CEEI/UFCG

Professor Dr. Francisco Vilar Brasileiro

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 28 de junho de 2023.

CAMPINA GRANDE - PB

2023

ABSTRACT

In the last decade, several research studies have been developed with the aim of classifying and identifying voice pathologies. In this context, the existence of a widely accessible tool to provide non-invasive pre-diagnosis for voice disease detection would be highly relevant in motivating users to seek medical assistance. This work proposes the development of a mobile application that classifies healthy and unhealthy voices in a binary manner. Supervised machine learning techniques are employed, along with a voice database containing healthy voice signals and signals affected by some pathologies, for training and validation purposes. The resulting application has an intuitive use and presents a relevant social impact by helping reduce the gap between doctors and individuals with voice pathologies. By leveraging this application, users will have an accessible means to assess their voice health, encouraging timely medical intervention.

Voice Pathology Detector: Desenvolvimento de uma Aplicação Móvel para Detecção de Patologias de Voz

Trabalho de Conclusão de Curso
Mateus Cavalcante de Almeida Farias Aires, Herman Martins Gomes
mateus.aires@ccc.ufcg.edu.br, hmg@computacao.ufcg.edu.br
Universidade Federal de Campina Grande
Campina Grande, Paraíba

ABSTRACT

In the last decade, several research studies have been developed with the aim of classifying and identifying voice pathologies. In this context, the existence of a widely accessible tool to provide non-invasive pre-diagnosis for voice disease detection would be highly relevant in motivating users to seek medical assistance. This work proposes the development of a mobile application that classifies healthy and unhealthy voices in a binary manner. Supervised machine learning techniques are employed, along with a voice database containing healthy voice signals and signals affected by some pathologies, for training and validation purposes. The resulting application has an intuitive use and presents a relevant social impact by helping reduce the gap between doctors and individuals with voice pathologies. By leveraging this application, users will have an accessible means to assess their voice health, encouraging timely medical intervention.

RESUMO

Na última década, diversas pesquisas vêm sendo desenvolvidas com o objetivo de classificar e identificar patologias de voz. Nesse contexto, a existência de uma ferramenta, de amplo acesso a usuários, que provesse um pré-diagnóstico não invasivo para a detecção de doenças de voz seria de grande relevância no sentido de motivar usuários a buscarem assistência médica. Propõe-se, neste trabalho, o desenvolvimento de uma aplicação móvel que classifica, de forma binária, vozes saudáveis e não saudáveis. Para isso, são utilizadas técnicas de aprendizagem de máquina supervisionada, além de uma base de dados de vozes com quantidade suficiente de amostras para viabilizar o treinamento satisfatório e permitir a generalização da ferramenta para amostras não treinadas. Dessa forma, buscou-se desenvolver uma aplicação pontual, simples, de uso intuitivo e de alto impacto social, que encurtará a distância entre médicos e pessoas com alguma patologia de voz.

PALAVRAS-CHAVE

Voz, patologia, classificação, aprendizagem de máquina, aplicação.

LINK PARA O APLICATIVO

https://play.google.com/store/apps/details?id=com.airesapps.myapplication&hl=pt_BR&gl=US

1. INTRODUÇÃO

A voz desempenha um papel fundamental na comunicação humana. O seu surgimento é considerado um dos pilares da diferenciação entre seres humanos e outros animais, por ter permitido uma maior capacidade de organização em grupos e possibilitado a transmissão de conhecimentos ao longo de gerações. Tal qual é inegável a importância da comunicação no processo de formação de comunidades humanas, é também inegável a essencialidade da voz no fenômeno da comunicação como um todo.

A produção de voz não é um processo simples: envolve uma complexa interação fluido-estrutural dentro da glote, controlada pela ativação do músculo laríngeo (ZHANG, 2016). Por isso, as patologias nas dobras vocais são multiformes; e suas causas são, muitas vezes, multifatoriais. Desordens funcionais (geradas por abusos de voz) e patologias laríngeas (nódulos nas pregas vocais, pólipos, úlceras, carcinomas e paralisia do nervo laríngeo)

são as principais fontes de desordens de voz em seres humanos. Os diferentes procedimentos clínicos tradicionais para a aplicação de exames laríngeos são, em sua natureza, invasivos (MARTINEZ; RUFINER, 2000).

Nesse contexto, faz-se necessária a busca por métodos não invasivos para a detecção de patologias de voz. Ferramentas com esse propósito não somente facilitariam o processo de obtenção de um pré-diagnóstico, pela sua natureza não invasiva, como também motivariam usuários a buscarem assistência médica em casos de detecção de disfonias.

Nesses últimos anos, tecnologia e assistência médica caminham em sintonia. De forma mais específica, a aplicação de algoritmos de aprendizagem de máquina para a detecção de quadros clínicos anômalos permite iniciar com maior antecedência o tratamento de eventuais patologias detectadas (ŠABIĆ; KEELEY; HENDERSON; NANNEMANN, 2020). Considerando a importância da rapidez de um pré-diagnóstico, ferramentas de detecção de anomalias fisiológicas não deveriam existir somente em contextos hospitalares.

A tecnologia móvel torna-se, então, o meio ideal para abarcar o desenvolvimento dessa ferramenta, especialmente pelos aspectos de acessibilidade e portabilidade que os dispositivos móveis oferecem. Aplicativos de *smartphones* se mostram como um verdadeiro ferramental, diverso em utilidades, que pode ser convenientemente carregado no bolso.

Dessa forma, o presente trabalho de conclusão de curso propõe o desenvolvimento de uma aplicação móvel para classificar, de forma binária, vozes saudáveis e não saudáveis, captadas pelo microfone de um *smartphone;* a fim de ser utilizada como ferramenta de pré-diagnóstico. Para isso, serão utilizadas técnicas de aprendizagem máquina supervisionada, além de uma base de dados de vozes com quantidade suficiente de amostras para viabilizar o treinamento satisfatório e permitir um alto poder de generalização da ferramenta.

Os desafios implícitos nesse desenvolvimento dizem respeito à complexidade do sinal de voz no domínio do tempo, por estar sujeito a variações de timbre e a ruídos gerados pelo processo de captura de voz, tornando necessária uma etapa de pré-processamento. Não obstante o reconhecimento de tais limitações, espera-se o desenvolvimento de uma aplicação pontual, simples e de alto impacto social, que encurtará a distância entre médicos e pessoas com alguma patologia de voz; enfatizando, assim, a importância de estudos que dialoguem com a saúde a partir da popularização e conveniência das tecnologias móveis.

O restante deste documento está organizado como segue. Inicialmente, os objetivos gerais e específicos são definidos de forma concisa, estabelecendo as metas a serem alcançadas. Em seguida, na seção de "Metodologia e Treinamento do Modelo", são detalhadas as etapas de obtenção e tratamento dos dados, extração de características dos sinais de áudio e o treinamento do modelo de aprendizado de máquina selecionado. Posteriormente, na seção "Desenvolvimento da Aplicação em Nuvem", aborda-se a concepção e implementação da aplicação em nuvem responsável pelo tratamento dos dados e realização das predições. Por fim, na seção "Aplicação Android", é descrito o processo de desenvolvimento do aplicativo *Voice Pathology Detector*, destacando aspectos relevantes de arquitetura de software e design.

2. OBJETIVOS

Este trabalho tem como objetivo principal desenvolver um aplicativo móvel para detecção de patologias de voz de fácil acesso ao público.

Para isso, foi implementada uma arquitetura de aprendizagem de máquina baseada em estudos existentes na área. Esse passo foi fundamental para garantir a eficiência e precisão da solução proposta na classificação de vozes saudáveis e anômalas. A arquitetura foi treinada com uma base de dados de vozes com quantidade suficiente de amostras para viabilizar o treinamento satisfatório e permitir um alto poder de generalização da ferramenta; tendo sido validado por dados de teste, não utilizados em seu treinamento. Por fim, foi também desenvolvida uma aplicação móvel que utiliza a arquitetura mencionada, e que pode ser utilizada para a realização de predições online de forma rápida e prática.

3. METODOLOGIA E TREINAMENTO DO MODELO

Esta seção descreve a obtenção e organização de dados; extração e normalização de características; e treinamento do modelo.

3.1 Obtenção e Organização dos Dados

Os dados foram obtidos da base de dados Saarbruecken Voice Database[1] (SVD). Essa base de dados é publicamente acessível através de uma interface Web e contém gravações de mais de 2.000 sujeitos. Cada sessão de gravação consiste em uma coleção de gravações das vogais sustentadas /a/, /i/

[1] https://stimmdb.coli.uni-saarland.de/

e /u/ em diferentes entonações, além da sentença "Guten Morgen, wie geht es Ihnen?" ("Bom dia, como vai você?", em alemão). Cada sessão de aquisição contém um total de 13 arquivos de áudio, com duração variando de 1 a 3 segundos para os sinais de voz das vogais sustentadas. A base de dados inclui 869 sessões de voz em condição normal e 1.356 sessões de voz afetadas por uma ou mais das 71 patologias catalogadas. Essas patologias apresentam origens diversas, incluindo condições neurológicas como Mal de Parkinson ou Paralisia Bulbar, condições psicológicas como Afonia e Microfonia Psicogênicas, e condições que afetam diferentes áreas do trato vocal, como Edema de Reinke (pregas vocais), doença de Forestier (faringe) ou Cisto Valecular (língua e epiglote) (MARINUS, 2019).

Os dados foram organizados em diretórios para facilitar o acesso e a manipulação. Foi criado um notebook Python para acessar os diretórios e realizar as etapas subsequentes, conforme detalhado a seguir.

3.2 Extração de Características

Para a extração de características, foram selecionadas diversas funções, com base nas bibliotecas *Librosa*[2] e *Parselmouth*[3], com o objetivo de capturar diferentes características acústicas das vozes analisadas. Abaixo, segue uma breve explicação sobre cada uma delas, e sua relevância na identificação de patologias de voz.

Atributos Mel-Spectrograma: O mel-spectrograma é uma representação do espectro de frequência de um sinal de áudio, ponderado de acordo com a percepção humana de frequência. Essa técnica é útil para capturar informações relevantes sobre a distribuição de energia espectral em diferentes frequências ao longo do tempo (LOGAN, 2000).

Características Acústicas: As características acústicas, como pitch, jitter e shimmer, fornecem informações sobre a qualidade e a estabilidade da voz. O pitch mede a frequência fundamental da voz (TALKIN, 1995); e o jitter e o shimmer são medidas de variação na frequência fundamental e na amplitude do sinal de voz, respectivamente (CESARI, DE PIETRO, MARCIANO, NIRI, SANNINO, VERDE, 2018).

Contraste Espectral: O contraste espectral é uma medida que quantifica a diferença de energia entre diferentes regiões do espectro de frequência. Essa feature é útil para identificar mudanças abruptas ou anormais na distribuição de energia espectral da voz (JIANG, LU, ZHANG, TAO, CAI, 2002).

Taxa de Cruzamento por Zero: A taxa de cruzamento por zero é uma medida que indica a frequência com que o sinal de áudio atravessa o valor zero. Essa feature é útil para identificar a presença de ruídos ou alterações na voz que podem ser indicativos de patologias (SHETE, PATIL, PATIL, 2014).

Raiz do Valor Quadrático Médio (RMS): O RMS é uma medida que representa a energia média do sinal de áudio. Essa feature pode fornecer informações sobre a intensidade ou volume da voz. Alterações significativas no RMS podem indicar problemas vocais (KUMAR, 2004).

Todas as funções de extração de características foram chamadas em uma função geral chamada *"extract_features"*, que recebe um arquivo de áudio como entrada e retorna um array com as características correspondentes.

Para o subsequente armazenamento dos dados em uma tabela, todos os arrays precisam apresentar a mesma dimensão. Para garantir isso, foram descartados arquivos de áudio que apresentavam menos de um segundo de duração; e, dos que apresentavam mais de um segundo, foi considerado somente seu segundo inicial.

Dessa forma, para cada áudio, um array de 348 colunas e uma linha foi gerado. Todos os áudios processados foram combinados em um arquivo CSV, onde uma coluna adicional denominada *"has_pathology"* foi adicionada para indicar a presença de patologia vocal, sendo atribuído o valor 0 para vozes saudáveis e 1 para vozes com disfonia. O arquivo CSV final resultou em uma tabela com dimensões de 11772 linhas × 349 colunas, que foi salva para uso posterior.

3.3 Normalização dos Dados

Os dados foram normalizados utilizando o *StandardScaler*, uma técnica comumente empregada para padronizar vetores de características de entrada de um modelo de aprendizagem de máquina. O *StandardScaler* transforma valores numéricos para terem média zero e desvio padrão igual a um, o que é importante para garantir que todas as características estejam em uma escala comparável e para facilitar o treinamento adequado do modelo. Os dados normalizados foram salvos em outro arquivo CSV para uso posterior.

3.4 Treinamento do Modelo

Para o treinamento do modelo de classificação de vozes saudáveis e não saudáveis, foi selecionado o algoritmo *XGBoost*. O *XGBoost* é conhecido por sua eficiência e desempenho em problemas de classificação. Além disso, o

[2] https://librosa.org/doc/latest/index.html
[3] https://parselmouth.readthedocs.io/en/stable/

XGBoost oferece suporte a otimização do modelo por meio de regularização, lida bem com características não lineares e é capaz de lidar com grandes conjuntos de dados.

Para otimizar o desempenho do modelo *XGBoost*, foram escolhidos hiperparâmetros específicos. O parâmetro *"max_depth"* foi definido como 15 para controlar a profundidade máxima das árvores de decisão, permitindo que o modelo capture relações mais complexas nos dados. O valor de *"learning_rate"* foi estabelecido em 0.01, o que indica uma taxa de aprendizado baixa, garantindo uma abordagem mais cautelosa durante o processo de ajuste do modelo. Quanto ao parâmetro *"n_estimators"*, foi definido como 1000, determinando o número de árvores de decisão a serem criadas no modelo. Por fim, a regularização L1, representada pelo hiperparâmetro *"reg_alpha"*, é empregada para evitar o *overfitting*, penalizando os coeficientes do modelo. Nesse caso, um valor de 0.005 foi escolhido para aplicar uma penalidade leve, evitando que os coeficientes assumam valores extremamente altos, com o propósito de controlar o *overfitting* enquanto ainda permite que o modelo explore de forma flexível as relações entre as características. Essa configuração foi escolhida com base em experimentos anteriores e análise do desempenho do modelo, visando obter um equilíbrio entre a capacidade de generalização e o tempo de treinamento.

Antes de prosseguir com o treinamento, os dados foram divididos em conjuntos de treinamento e teste. Foi adotada uma proporção de 80% para treinamento e 20% para teste. A estratificação foi utilizada durante a divisão dos dados para garantir que as proporções de vozes saudáveis e não saudáveis fossem mantidas em ambas as partições, evitando qualquer viés de distribuição dos dados.

Após o treinamento do modelo com os dados de treinamento, a acurácia do modelo foi avaliada utilizando os dados de teste. A acurácia é calculada dividindo o número de predições corretas pelo total de predições realizadas e multiplicando o resultado por 100 para obter uma porcentagem. A fórmula de cálculo da acurácia é a seguinte:

*Acurácia = (Predições corretas / Total de predições) * 100*

No presente estudo, o modelo obteve uma acurácia de 70,91%, o que significa que ele classificou corretamente 70,91% das vozes como saudáveis ou não saudáveis. Essa taxa de acurácia indica uma capacidade de predição aceitável para uma ferramenta móvel de suporte ao diagnóstico preliminar de patologias da fala..

Cabe ressaltar que o tempo de treinamento do modelo foi de aproximadamente 223 segundos, enquanto o tempo de predição foi de 0.04 segundos, utilizando-se de um computador equipado com um processador Intel Core i7 de 7ª geração e 32 GB de memória RAM.

Essa seção descreveu a metodologia utilizada para treinar o modelo de classificação de vozes saudáveis e não saudáveis. A próxima seção aborda o tratamento de novos dados e o desenvolvimento da aplicação móvel proposta.

4. DESENVOLVIMENTO DA APLICAÇÃO EM NUVEM

Nesta seção, é descrito o desenvolvimento da aplicação em nuvem que faz uso do modelo treinado na etapa anterior para realizar predições sobre os sinais de áudio de usuários do aplicativo móvel desenvolvido (descrito na próxima seção).

Inicialmente, é importante expor os motivos que baseiam a decisão de abstrair a predição de novos dados da aplicação Android. Em primeiro lugar, as funções de extração de características e pré-processamento de arquivos de áudio foram implementadas na linguagem de programação Python, ao invés da linguagem Java, escolhida para o desenvolvimento da aplicação móvel; porque a linguagem Python possui uma vasta biblioteca de código aberto voltada para ciência de dados. Além disso, optamos por abstrair o processo de predição e pré-processamento na aplicação Android para permitir que ela seja escalável de forma independente. Dessa forma, a carga de trabalho dessas etapas não recairá sobre o dispositivo do usuário.

Em formato semelhante aos dados de treino, são requeridos três áudios do usuário: um em que o usuário fala a vogal sustentada"a", outro em que fala a vogal "i" e outro em que fala a vogal "u". Com base nesses três áudios, a aplicação realiza a classificação de presença de sintomas de patologia.

No diretório da aplicação, encontram-se diferentes arquivos python com diferentes propósitos: um arquivo de execução da aplicação; um arquivo que contém funções de processamento, extração de características e de realização de classificação de arquivos de áudio; e, finalmente, um arquivo contendo funções utilitárias. Além disso, encontra-se no diretório raiz o modelo *XGBoost* treinado (v. seção anterior), juntamente com o *StandardScaler* mencionado para normalizar os dados de treinamento. Esses modelos são carregados no início da execução da aplicação.

O arquivo de execução carrega os modelos de classificação e normalização. Além disso, contém uma rota */predict* que, ao ser acessada pela aplicação Android, retorna o resultado

da detecção (positivo ou negativo) por meio de um valor booleano; e a probabilidade de uma classificação positiva.

Considerando que o modelo foi treinado com áudios de exatamente um segundo de duração, será necessário manter a mesma dimensionalidade para a classificação de novos dados. Para esse fim, foram realizadas as seguintes etapas:

1. Remoção de silêncio do áudio, para que os segmentos silenciosos não atrapalhem a classificação.
2. Descarte do primeiro meio segundo audível do áudio; pela observação de que existe uma tendência desproporcional de predição positiva no segmento inicial do áudio.
3. Divisão de cada áudio, sem silêncio, em segmentos de um segundo;
4. Extração de características, normalização e cálculo de probabilidade de classificação positiva para cada segundo do áudio,
5. Cálculo do valor representativo da probabilidade de classificação positiva de cada áudio. Considerando que, por vezes, os sintomas de patologia de voz se apresentam de forma pontual no sinal de voz, foi estabelecida a seguinte regra para o cálculo deste valor: se algum segmento do áudio apresentar uma probabilidade igual ou acima de 0.95 de classificação positiva, este será o valor; caso contrário, o valor será a média de probabilidades de cada segmento do áudio.
6. Considerando os três áudios do usuário (“a”, “i”, “u”), e um valor de limiar fornecido (sendo considerado o valor padrão 0.65), é realizado um esquema de votação para determinar se o usuário apresenta sintomas de patologia da seguinte forma: se o valor representativo de cada áudio for maior do que o limiar, a votação é positiva; se não negativa. O resultado mais votado, seja afirmativo ou negativo, determina se o usuário apresenta sintomas de patologia de voz.

A plataforma de implantação da aplicação escolhida foi *Rail*[4]*way*, por motivos de simplicidade e capacidade de execução contínua na nuvem.

A próxima seção descreve o desenvolvimento da aplicação Android, que é responsável pela gravação dos três áudios do usuário e pela exibição do resultado de classificação.

[4] https://railway.app/

5. APLICAÇÃO ANDROID

Nesta seção, é descrito o processo de desenvolvimento do aplicativo Android chamado *Voice Pathology Detector*.

5.1 Tecnologias de Desenvolvimento

O desenvolvimento da aplicação seguiu o desenvolvimento nativo em Java, aproveitando a ampla gama de bibliotecas e APIs para o desenvolvimento de aplicativos robustos e escaláveis que essa linguagem de programação provê. A escolha do desenvolvimento nativo permite um alto nível de controle sobre o desempenho e recursos do dispositivo, além de proporcionar uma experiência mais fluida e integrada aos usuários. A utilização do *Android Studio*[5] como IDE principal proporciona um ambiente de desenvolvimento robusto e completo, com recursos avançados de depuração, testes e desenvolvimento de interfaces.

A exibição da aplicação foi implementada em XML, uma linguagem de marcação amplamente utilizada no desenvolvimento de interfaces para aplicativos Android. Essa abordagem permite uma separação clara entre a lógica de programação em Java e a definição da interface em XML, facilitando a manutenção e personalização visual da aplicação para cada estado do sistema. Foram escolhidas imagens de livre acesso ao público, garantindo a acessibilidade e a conformidade com as licenças de uso do conteúdo visual utilizado.

5.2 Arquitetura do Software

Dada a simplicidade da solução proposta, que consiste apenas em gravar os áudios do usuário, comunicar-se com a aplicação em nuvem e exibir os resultados, não se fez necessário o desenvolvimento de etapas de autenticação nem configuração de banco de dados. A arquitetura da aplicação, ilustrada na Figura 1, é composta por dois módulos principais:

MainActivity: Responsável pela exibição dos componentes visuais para o usuário de acordo com cada estado da aplicação. Além disso, é responsável pela gravação dos áudios e pela chamada da função que realiza a requisição à API em nuvem para obter o resultado da classificação. A MainActivity gerencia a interação com o usuário e coordena o fluxo de execução da aplicação.

PathologyPredictionClient: É o cliente responsável por realizar a requisição à API em nuvem e obter o resultado da classificação. Esse módulo é responsável por lidar com a

[5] https://developer.android.com/studio

comunicação de rede e processamento dos dados retornados pela API.

Além dos dois módulos principais, o sistema conta com algumas classes utilitárias:

ObservationsUtil: Classe utilitária responsável por construir o texto exibido na tela de instruções.

Threshold: Classe que armazena o limiar padrão utilizado para montar a requisição de classificação. Essa classe foi criada devido à necessidade de utilizar esse limiar em tempo de execução na tela de instruções.

Constants: Armazena constantes utilizadas em diferentes partes do aplicativo.

A Figura 2 mostra a estrutura de arquivos do projeto.

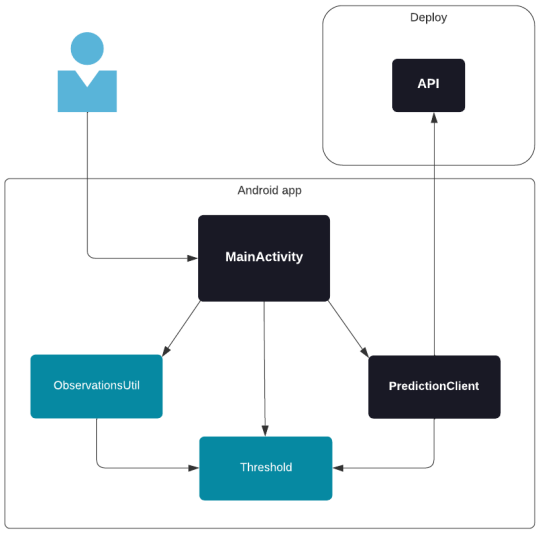

FIGURA 1: Arquitetura da Aplicação Android

5.3 Fluxo de Execução e Telas

O aplicativo segue um fluxo de execução simples e intuitivo. O usuário é guiado por diferentes telas e interações que ocorrem durante o processo de gravação e classificação dos áudios.

A máquina de estados do sistema, ilustrada na Figura 3, compõe os seguintes estados, cada qual com uma tela de aplicação diferente:

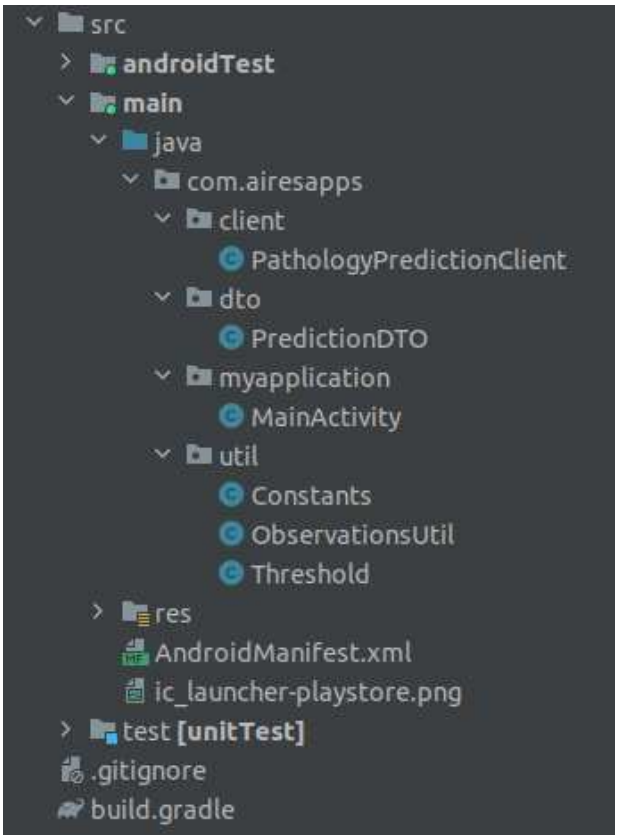

FIGURA 2: Estrutura de Arquivos e Diretórios da Aplicação Android

Tela Inicial (Figura 4): Quando o aplicativo é iniciado, o usuário é direcionado para a tela inicial, onde é exibido o nome da aplicação e um botão de “Iniciar”.

Tela de Observações e Instruções (Figura 4): Após a tela inicial, o usuário é apresentado à tela de observações e instruções, em forma de uma janela Dialog Box. Nessa tela, são fornecidas orientações sobre como realizar a gravação dos áudios e são dadas informações importantes para o correto funcionamento do aplicativo. O usuário pode ler as instruções e fechar essa tela para prosseguir. O botão de ajuda, que abre novamente a janela de observações e instruções, encontra-se no canto superior durante todos os estados seguintes do sistema.

Pronto para Gravar (Figura 5): Uma vez fechada a tela de observações e instruções, o aplicativo está pronto para iniciar o processo de gravação. A próxima vogal a ser gravada é exibida na tela, juntamente com uma mensagem indicando ao usuário que comece a falar a vogal específica por mais de três segundos. Essa instrução é apresentada

como um balão na interface do aplicativo. Além disso, é exibido um botão de gravação na forma de um microfone.

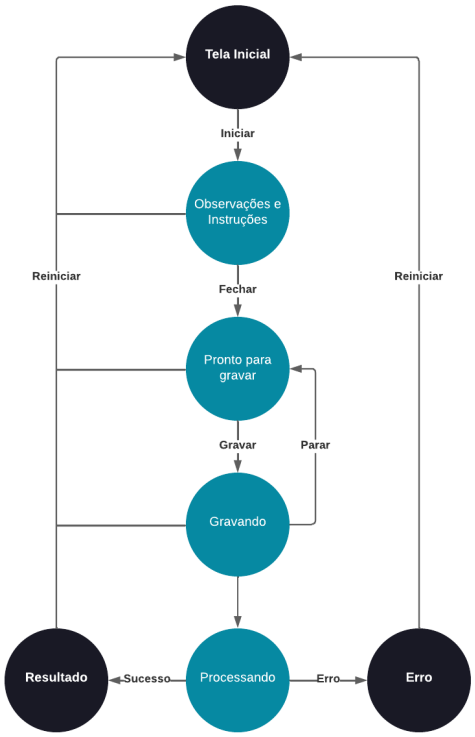

FIGURA 3: Máquina de Estados da Aplicação Android

Gravando (Figura 5): Ao clicar no ícone de gravar, representado por um microfone, o usuário inicia a gravação da vogal solicitada. Durante a gravação, a tela exibe a mensagem "Ouvindo vogal [vogal atual]", indicando ao usuário que o áudio está sendo registrado. Esse processo se repete para cada vogal a ser gravada. Para guardar a informação de qual vogal deve ser gravada no estado atual, o sistema utiliza, internamente, uma variável global.

Processando (Figura 6): Após a gravação das três vogais, quando a variável global possui o valor maior que 3, a função responsável por enviar a requisição para a API em nuvem é chamada. Durante esse processo, é exibida uma tela de processamento com a mensagem "Processando...".

Tela de Erro (Figura 6): Caso ocorra algum erro durante o envio da requisição ou no recebimento da resposta da API em nuvem, o aplicativo exibirá uma tela de erro correspondente ao tipo de falha ocorrida. As possíveis mensagens de erro incluem "Gravação audível muito curta. Fale por mais de três segundos.", para casos em que a gravação ou a gravação audível é muito curta; "Erro de conexão", se houver problemas de comunicação com a API; e "Erro interno", quando ocorre um erro desconhecido.

Tela de Resultados (Figura 7): Em caso de sucesso na requisição, os resultados da classificação são exibidos ao usuário. Se o resultado for negativo, ou seja, a voz é considerada saudável, é exibida a mensagem "Sua voz parece estar saudável". Caso o resultado seja positivo, indicando sintomas de disfonia, a mensagem exibida será "Sua voz está apresentando sintomas de disfonia", juntamente com a probabilidade de uma classificação positiva, calculada na seção anterior.

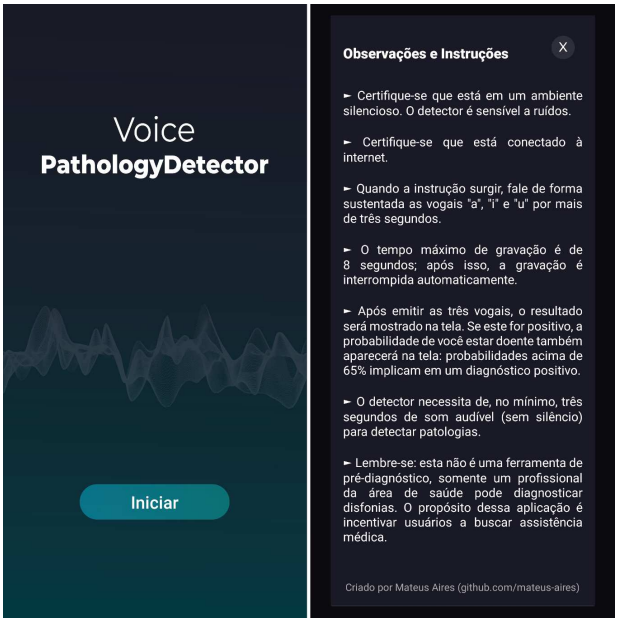

FIGURA 4: Tela Inicial e de Observações

5.4 Design da Aplicação

O design escolhido para o *Voice Pathology Detector* é simples e elegante, em consonância com a natureza séria da aplicação que se trata de uma ferramenta de auxílio à saúde vocal dos usuários, transmitindo confiança e profissionalismo.

O ícone do aplicativo (Figura 8), que é exibido tanto nos smartphones dos usuários quanto na loja do Google Play, foi projetado para representar os conceitos-chave relacionados à voz. Apresenta o símbolo clássico de representação da voz, em forma de ondas, sobreposto a um

símbolo que faz referência ao caminho percorrido pela voz no corpo, desde sua produção até sua emissão externa. A intersecção escura e negativa dos dois símbolos em um segmento de voz destaca a ideia de que alguns sintomas de patologia de voz podem se revelar de forma pontual, como uma falha ou anormalidade em um pequeno trecho de áudio. As cores escolhidas para o ícone foram selecionadas para combinar com as cores utilizadas nas telas da aplicação, que foram concebidas primeiramente.

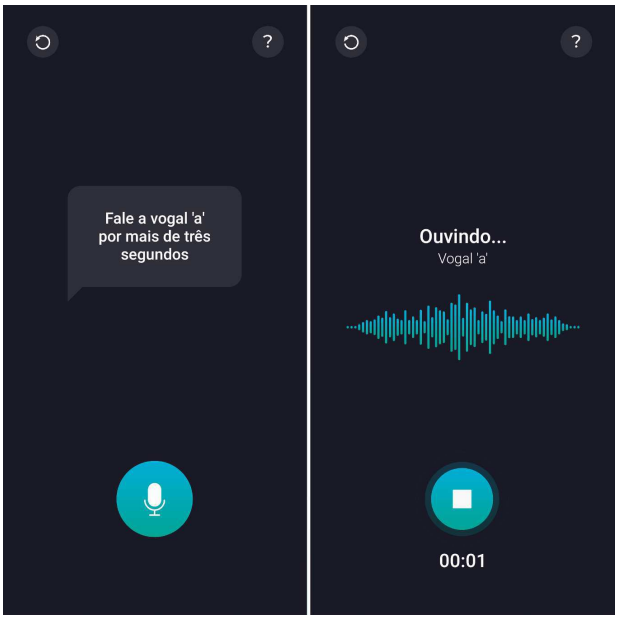

FIGURA 5: Telas “Pronto para Gravar” e “Gravando”

6. CONSIDERAÇÕES FINAIS

A investigação realizada com o objetivo de desenvolver a solução proposta não teve como premissa a criação de uma ferramenta de diagnóstico de patologias vocais. O objetivo central era o desenvolvimento de uma ferramenta que pudesse servir como pré-diagnóstico de forma a incentivar os usuários a buscar assistência médica, o que é de extrema importância devido à necessidade de tratamentos médicos rápidos e antecipados.

Nesse contexto, a aplicação proposta desempenha efetivamente o seu papel. Através da utilização de técnicas de aprendizado de máquina supervisionado e uma base de dados pública e adequada, a ferramenta classifica de forma binária vozes saudáveis e não saudáveis com uma acurácia aceitável, possibilitando um pré-diagnóstico não invasivo. Dessa forma, a aplicação contribui para encurtar a distância entre médicos especialistas e pessoas com possíveis patologias de voz, promovendo um maior acesso aos cuidados médicos necessários.

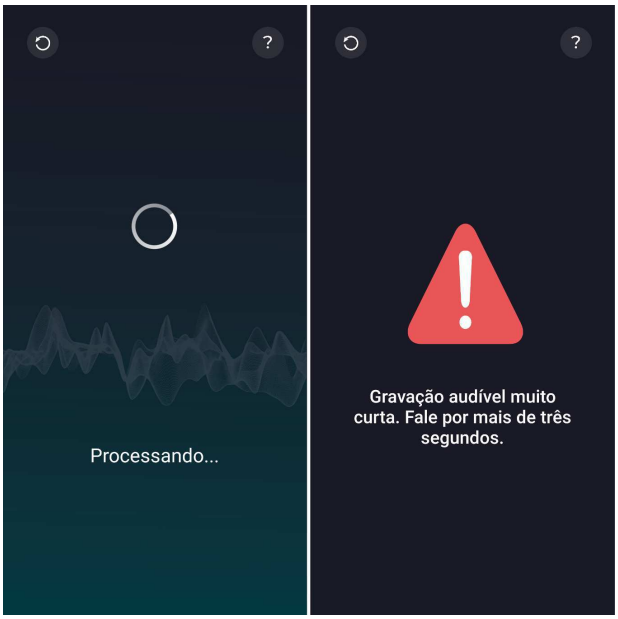

FIGURA 6: Telas “Processando” e “Erro”

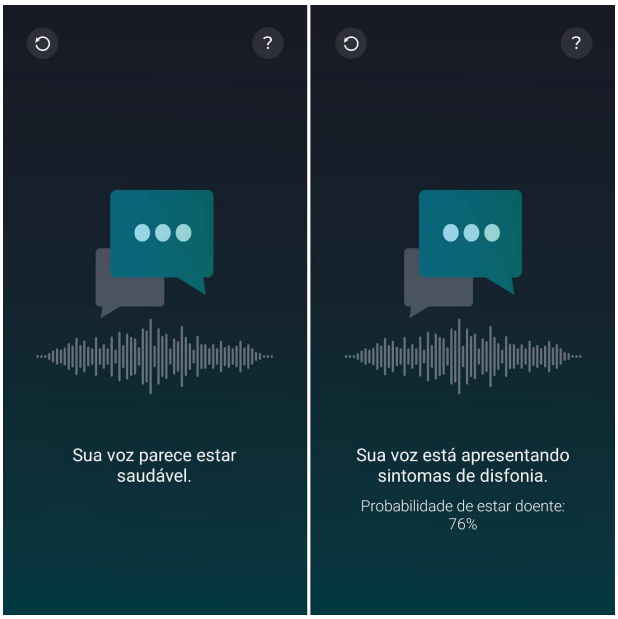

FIGURA 7: Telas de Resultados

FIGURA 8: Ícone do aplicativo

No entanto, algumas melhorias podem ser consideradas para aprimorar a aplicação e sua utilidade. Primeiramente, é desejável adicionar suporte a múltiplas linguagens, não se limitando apenas ao português brasileiro, a fim de tornar a ferramenta mais acessível a usuários de diferentes regiões e idiomas. Além disso, sugere-se realizar decisões arquiteturais, como a divisão de responsabilidades do módulo Java principal, de forma que sejam separadas as funções de gerenciamento de componentes visuais e de gravação de voz.

No que diz respeito ao processamento de áudio, é importante considerar melhorias para lidar com ruídos. O modelo atualmente empregado na aplicação é sensível a interferências sonoras, de forma que se faz necessária a investigação de técnicas de pré-processamento de áudio que permitam maior flexibilidade ao usuário em ambientes com ruídos mínimos. Além disso, identificar áudios que possuem ruídos inviabilizadores, e fornecer um aviso prévio de classificação inviável, comunicando essa informação como um erro, pode melhorar ainda mais a experiência do usuário.

Em suma, o desenvolvimento desta aplicação móvel demonstrou a viabilidade de utilizar técnicas de aprendizado de máquina para a classificação de patologias vocais e sua utilização pelo público geral de forma não invasiva. Através da combinação de acessibilidade e portabilidade proporcionadas por dispositivos móveis, a ferramenta busca incentivar a procura por assistência médica, promovendo um significativo impacto social. Com as melhorias sugeridas, a aplicação poderá ser aprimorada em termos de usabilidade, adaptabilidade linguística e robustez no processamento de áudio, ampliando seu potencial e alcance na área da saúde vocal.

7. AGRADECIMENTOS

Agradeço, primeiramente, aos meus pais, Juarez e Kalina, e ao meu irmão Abdias, pelo amor e apoio incondicionais, e por viabilizarem tudo de positivo em minha vida. Agradeço ao meu professor orientador Herman Martins Gomes, pelo suporte, disponibilidade e diligência. Agradeço a José Alberto Souza Paulino, pelo esforço inicial em impulsionar e auxiliar esta pesquisa. Agradeço também a Alana Souza Santos, pela concepção do design da aplicação Android, e pelo garantido apoio acadêmico, psicológico e, especialmente, emocional. Agradeço a Allyson Barbosa, pela concepção do ícone do aplicativo. Agradeço aos amigos e familiares por me proporcionarem as minhas melhores memórias. E agradeço à Universidade Federal de Campina Grande por servir de contexto concreto e abstrato à fase mais interessante da minha vida.

8. REFERÊNCIAS

MARTINEZ, C. E.; RUFINER, H. L. Acoustic analysis of speech for detection of laryngeal pathologies. *In: Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (Cat. No. 00CH37143)*, Chicago, v. 3, p. 2369-2372, jul. 2000. IEEE. http://dx.doi.org/10.1109/iembs.2000.900621.

ŠABIĆ, E.; KEELEY, D.; HENDERSON, B.; NANNEMANN, S. Healthcare and anomaly detection: using machine learning to predict anomalies in heart rate data. Ai & Society, [S.L.], v. 36, n. 1, p. 149-158, 7 maio 2020. Springer Science and Business Media LLC. http://dx.doi.org/10.1007/s00146-020-00985-1.

ZHANG, Z. Mechanics of human voice production and control. The Journal Of The Acoustical Society Of America, [S.L.], v. 140, n. 4, p. 2614-2635, out. 2016. Acoustical Society of America (ASA). http://dx.doi.org/10.1121/1.4964509

TALKIN, D. A robust algorithm for pitch tracking (RAPT). Speech Coding and Synthesis, vol. 495, p. 497, 1995.

CESARI, U., DE PIETRO, G., MARCIANO, E., NIRI, C., SANNINO, G., & VERDE, L. (2018). Voice Disorder Detection via an m-Health System: Design and Results of a Clinical Study to Evaluate Vox4Health. In BioMed Research International (Vol. 2018, pp. 1–19). Hindawi Limited. https://doi.org/10.1155/2018/8193694

JIANG, Dan-Ning et al. Music type classification by spectral contrast feature. In: Proceedings. IEEE International Conference on Multimedia and Expo. IEEE, 2002. p. 113-116.

SHETE, D. S.; PATIL, S. B.; PATIL, S. Zero crossing rate and Energy of the Speech Signal of Devanagari Script. IOSR-JVSP, v. 4, n. 1, p. 1-5, 2014.

KUMAR, Sanjay et al. EMG based voice recognition. In: Proceedings of the 2004 Intelligent Sensors, Sensor Networks and Information Processing Conference, 2004. IEEE, 2004. p. 593-597.

MARINUS, João Vilian de Moraes Lima. Classificação de sinais de voz afetada por patologia nas pregas vocais utilizando reconstrução do espaço de fases. Tese de Doutorado, Universidade Federal de Campina Grande, Campina Grande, 2019.