



INSTITUTO SUPERIOR POLITÉCNICO DA HUÍLA
UNIVERSIDADE MANDUME YA NDEMUFAYO

RELATÓRIO DE APRENDIZAGEM COMPUTACIONAL

TEMA: CONSTRUÇÃO DE UMA APLICAÇÃO MÓVEL PARA ANÁLISE DO
SOM DA TOSSE E IDENTIFICAÇÃO DE PNEUMONIA E BRONQUITE.



Curso: Engenharia Informática

4º Ano

DOCENTE

Abel Zacarias

Lubango aos 13 de Janeiro de 2026



INSTITUTO SUPERIOR POLITÉCNICO DA HUÍLA
UNIVERSIDADE MANDUME YA NDEMUFAYO

RELATÓRIO DE APRENDIZAGEM COMPUTACIONAL

TEMA: CONSTRUÇÃO DE UMA APLICAÇÃO MÓVEL PARA ANÁLISE DO
SOM DA TOSSE E IDENTIFICAÇÃO DE PNEUMONIA E BRONQUITE.

Autores :

ADMILSON JOB - 2022109610

CRISTÓFER KAHALA - 2022121256

DOMINGOS CHIVANGULULA - 20221432

INÁCIO MAHOLI - 2022139183

Curso: Engenharia Informática

4º Ano

DOCENTE

Abel Zacarias

Lubango aos 13 de Janeiro de 2026

Índice

Índice de ilustrações	5
RESUMO	6
1. INTRODUÇÃO	7
1.1 Justificação do Estudo.....	8
1.2 Relevância Académica e Social	9
2. OBJECTIVOS DO PROJECTO	9
2. OBJECTIVOS DO PROJECTO	9
2.1 Objectivo Geral	9
2.2 Objectivos Específicos	9
3. FUNDAMENTAÇÃO TEÓRICA	10
3.1 Aprendizagem Computacional.....	10
3.2 Processamento de Sinais de Áudio	10
3.3 Doenças Respiratórias: Pneumonia e Bronquite	11
4. REVISÃO DE LITERATURA	11
5. DESCRIÇÃO DO PROBLEMA	11
5. DESCRIÇÃO DO PROBLEMA	12
5.1 Aprendizagem Computacional.....	12
5.3 Doenças Respiratórias: Pneumonia e Bronquite	12
6. DESCRIÇÃO DO PROBLEMA	12
7. METODOLOGIA	12
7.1 Abordagem Metodológica.....	12
7.2 Definição dos Requisitos	12
7.3 Recolha e Preparação dos Dados	13
7.4 Pré-processamento	13
8. EXTRACÇÃO DE CARACTERÍSTICAS	13
8.1 MFCC (Mel Frequency Cepstral Coefficients).....	13
8.2 Espectrogramas e Log-Mel Spectrogramas	14
8.3 MFCC (Mel Frequency Cepstral Coefficients).....	14
8.4 Espectrogramas e Log-Mel Spectrogramas	14
9. SELECÇÃO E TREINO DO MODELO	14
9.1 Modelos Considerados.....	14
9.2 Justificação da Escolha	15
9.3 Modelos Considerados.....	15
9.4 Justificação da Escolha	15

9.5 Processo de Treino	15
10. DIVISÃO DOS DADOS	16
11. TREINO DO MODELO	17
11.2 Recolha e Preparação dos Dados	17
12. EXTRACÇÃO DE CARACTERÍSTICAS	17
12.4 Outras Características	18
13. SELECÇÃO E TREINO DO MODELO	18
13.1 Modelos Considerados	18
13.2 Modelo Recomendado	18
14. DIVISÃO DOS DADOS	18
15. TREINO DO MODELO	18
16. AVALIAÇÃO DO MODELO	19
16.1 Métricas Utilizadas	19
16.2 Interpretação dos Resultados	19
17. DESENVOLVIMENTO DA APLICAÇÃO MÓVEL	20
17.1 Arquitectura da Aplicação	20
17.2 Front-end	20
17.4 Integração do Modelo	20
18. ARQUITETURA DO SISTEMA	21
18.1 Front-end	21
19. ARQUITETURA DO SISTEMA	21
20. BACKEND	21
21. RESULTADOS ESPERADOS	22
22. RESULTADOS ESPERADOS	22
23. BOAS PRÁTICAS E CONSIDERAÇÕES ÉTICAS	22
24. LIMITAÇÕES DO ESTUDO	22
25. CONCLUSÃO	23
26. REFERÊNCIAS BIBLIOGRÁFICA	24

Índice de ilustrações

Ilustração 1 - Tela principal.....	26
Ilustração 2- Tela de gravação de áudio.....	27
Ilustração 3 - Gravando áudio.....	28
Ilustração 4 - Áudio gravado e pronto a reproduzir.	29
Ilustração 5 - Recomendações.....	30
Ilustração 6 - Sobre o aplicativo	31

RESUMO

Este relatório descreve o desenvolvimento de um projecto prático cujo objectivo é a construção de uma aplicação móvel capaz de analisar o som da tosse e auxiliar na identificação de pneumonia e bronquite utilizando técnicas de Aprendizagem Computacional (Machine Learning). O trabalho aborda desde a definição do problema, recolha e preparação dos dados, extracção de características, selecção e treino de modelos, até à implementação da aplicação móvel e boas práticas. A proposta visa contribuir para soluções tecnológicas de apoio à saúde, especialmente em contextos com acesso limitado a serviços médicos.

1. INTRODUÇÃO

A utilização de técnicas de Aprendizagem Computacional (Machine Learning) na área da saúde tem crescido significativamente nos últimos anos, impulsionada pelo avanço das tecnologias móveis, pela disponibilidade de dados e pela necessidade de soluções rápidas e acessíveis para apoio ao diagnóstico (Brown et al., 2020). Em países em desenvolvimento, como Angola, o acesso a serviços de saúde especializados ainda é limitado em muitas regiões, o que torna as soluções tecnológicas uma alternativa promissora para triagem e apoio clínico.

Doenças respiratórias como a pneumonia e a bronquite continuam entre as principais causas de morbilidade e mortalidade, especialmente em crianças, idosos e pessoas com sistema imunitário comprometido (World Health Organization [WHO], 2022). A tosse é um dos sintomas mais frequentes dessas patologias e apresenta padrões acústicos específicos que podem ser analisados por algoritmos de processamento de sinais e aprendizagem automática (Abeyratne et al., 2013).

Este projecto propõe o desenvolvimento de uma aplicação móvel capaz de captar o som da tosse, realizar o pré-processamento do áudio, extrair características relevantes e aplicar modelos de Machine Learning para auxiliar na identificação de possíveis casos de pneumonia e bronquite. A solução visa funcionar como ferramenta de apoio à triagem, não substituindo o diagnóstico médico, mas contribuindo para a detecção precoce e encaminhamento adequado (Imran et al., 2020).

1.1 Justificação do Estudo

A utilização de técnicas de Aprendizagem Computacional (Machine Learning) na área da saúde tem crescido significativamente nos últimos anos, impulsionada pelo avanço das tecnologias móveis, pela disponibilidade de dados e pela necessidade de soluções rápidas e acessíveis para apoio ao diagnóstico. Em países em desenvolvimento, como Angola, o acesso a serviços de saúde especializados ainda é limitado em muitas regiões, o que torna as soluções tecnológicas uma alternativa promissora para triagem e apoio clínico.

Doenças respiratórias como a pneumonia e a bronquite continuam entre as principais causas de morbilidade e mortalidade, especialmente em crianças, idosos e pessoas com sistema imunitário comprometido. A tosse é um dos sintomas mais frequentes dessas patologias e apresenta padrões acústicos específicos que podem ser analisados por algoritmos de processamento de sinais e aprendizagem automática.

Este projecto propõe o desenvolvimento de uma aplicação móvel capaz de captar o som da tosse, realizar o pré-processamento do áudio, extrair características relevantes e aplicar modelos de Machine Learning para auxiliar na identificação de possíveis casos de pneumonia e bronquite. A solução visa funcionar como ferramenta de apoio à triagem, não substituindo o diagnóstico médico, mas contribuindo para a detecção precoce e encaminhamento adequado.

A elevada incidência de doenças respiratórias, associada à escassez de profissionais de saúde em determinadas regiões, justifica a necessidade de soluções inovadoras e acessíveis. A utilização de smartphones, que são cada vez mais comuns, permite levar tecnologia de apoio ao diagnóstico a comunidades remotas, reduzindo custos e tempo de resposta.

1.2 Relevância Académica e Social

Do ponto de vista académico, o projecto permite a aplicação prática de conceitos de Aprendizagem Computacional, Processamento de Sinais e Desenvolvimento de Aplicações Móveis. Socialmente, contribui para a melhoria do acesso à informação de saúde e para a consciencialização sobre doenças respiratórias.

2. OBJECTIVOS DO PROJECTO

A evolução das tecnologias móveis e da inteligência artificial tem permitido o desenvolvimento de aplicações capazes de auxiliar em diversas áreas, incluindo a saúde. Doenças respiratórias como a pneumonia e a bronquite continuam a ser um problema de saúde pública, especialmente em países em desenvolvimento. A tosse é um dos principais sintomas dessas patologias e contém padrões acústicos que podem ser analisados por algoritmos de Machine Learning.

Este projecto tem como objectivo principal desenvolver uma aplicação móvel que capture o som da tosse, processe o áudio e utilize modelos de aprendizagem computacional para auxiliar na identificação de possíveis casos de pneumonia e bronquite.

2. OBJECTIVOS DO PROJECTO

2.1 Objectivo Geral

Construir uma aplicação móvel capaz de analisar o som da tosse e auxiliar na identificação de pneumonia e bronquite.

2.2 Objectivos Específicos

- Recolher e preparar dados de áudio de tosse.
- Extrair características relevantes do sinal sonoro.
- Treinar modelos de Machine Learning para classificação.
- Implementar o modelo numa aplicação móvel.
- Avaliar o desempenho do sistema.

3. FUNDAMENTAÇÃO TEÓRICA

3.1 Aprendizagem Computacional

A Aprendizagem Computacional é um subcampo da Inteligência Artificial que se concentra no desenvolvimento de algoritmos capazes de aprender padrões a partir de dados e fazer previsões ou decisões sem serem explicitamente programados (Mitchell, 1997). Os principais tipos de aprendizagem incluem aprendizagem supervisionada, não supervisionada e por reforço.

Na aprendizagem supervisionada, os modelos são treinados com dados rotulados, sendo esta a abordagem utilizada neste projecto, uma vez que os áudios de tosse são associados a rótulos como normal, bronquite ou pneumonia (Goodfellow, Bengio & Courville, 2016).

A Aprendizagem Computacional é um subcampo da Inteligência Artificial que se concentra no desenvolvimento de algoritmos capazes de aprender padrões a partir de dados e fazer previsões ou decisões sem serem explicitamente programados. Os principais tipos de aprendizagem incluem aprendizagem supervisionada, não supervisionada e por reforço.

Na aprendizagem supervisionada, os modelos são treinados com dados rotulados, sendo esta a abordagem utilizada neste projecto, uma vez que os áudios de tosse são associados a rótulos como normal, bronquite ou pneumonia.

3.2 Processamento de Sinais de Áudio

O processamento de sinais de áudio envolve técnicas matemáticas e computacionais para analisar, modificar e sintetizar sons (Oppenheim & Schaffer, 2010). No contexto da tosse, é fundamental realizar operações como filtragem, normalização, segmentação e transformação no domínio da frequência para extrair informações relevantes (Rabiner & Schaffer, 2011).

O processamento de sinais de áudio envolve técnicas matemáticas e computacionais para analisar, modificar e sintetizar sons. No contexto da tosse, é fundamental realizar operações como filtragem, normalização, segmentação e transformação no domínio da frequência para extrair informações relevantes.

3.3 Doenças Respiratórias: Pneumonia e Bronquite

A pneumonia é uma infecção aguda do parênquima pulmonar, caracterizada por inflamação dos alvéolos e acumulação de líquido ou pus. A bronquite, por sua vez, é a inflamação dos brônquios e pode ser aguda ou crônica. Ambas apresentam manifestações clínicas semelhantes, incluindo tosse persistente, produção de secreções, dor torácica e dificuldade respiratória.

3.4 Tosse como Biomarcador Acústico

Estudos recentes demonstram que a tosse contém padrões acústicos específicos que podem ser utilizados como biomarcadores para diferentes patologias respiratórias. Parâmetros como frequência dominante, duração, energia e padrão temporal variam de acordo com a doença.

4. REVISÃO DE LITERATURA

Diversos trabalhos científicos têm explorado o uso de Machine Learning na análise de sons respiratórios. Abeyratne et al. (2013) demonstraram que a análise acústica da tosse pode ser utilizada para diagnosticar pneumonia em crianças com elevada precisão. Brown et al. (2020) exploraram o diagnóstico automático da COVID-19 a partir de sons respiratórios recolhidos de forma colaborativa.

Imran et al. (2020) desenvolveram um sistema baseado em inteligência artificial para diagnóstico preliminar de COVID-19 a partir de amostras de tosse, demonstrando a viabilidade de soluções móveis para triagem de doenças respiratórias. Estes estudos reforçam o potencial do uso de redes neurais convolucionais e modelos híbridos na classificação de patologias respiratórias.

5. DESCRIÇÃO DO PROBLEMA

Diversos trabalhos científicos têm explorado o uso de Machine Learning na análise de sons respiratórios. Pesquisas indicam que redes neurais convolucionais aplicadas a espectrogramas de tosse podem alcançar elevados níveis de precisão na classificação de doenças respiratórias. Outros estudos utilizam modelos híbridos CNN-LSTM para capturar características espaciais e temporais do sinal.

A utilização de datasets públicos como COUGHVID, Coswara e ICBHI tem sido fundamental para o avanço nesta área, permitindo a comparação de métodos e a validação de modelos em cenários reais.

5. DESCRIÇÃO DO PROBLEMA

5.1 Aprendizagem Computacional

A Aprendizagem Computacional é um ramo da Inteligência Artificial que permite que sistemas aprendam padrões a partir de dados e tomem decisões com base nesses padrões.

5.2 Processamento de Áudio

O processamento de áudio envolve a análise de sinais sonoros para extrair informações relevantes. No contexto da tosse, são analisados aspectos como frequência, intensidade, duração e padrão temporal.

5.3 Doenças Respiratórias: Pneumonia e Bronquite

A pneumonia é uma infecção dos pulmões que pode ser causada por bactérias, vírus ou fungos. A bronquite é a inflamação dos brônquios, geralmente associada a infecções ou exposição a irritantes. Ambas apresentam sintomas como tosse, dificuldade respiratória e produção de secreções.

6. DESCRIÇÃO DO PROBLEMA

O problema consiste em identificar, a partir do som da tosse, padrões que indiquem a presença de pneumonia ou bronquite. O desafio inclui lidar com ruído ambiental, variações entre indivíduos, qualidade do microfone e limitações dos dispositivos móveis.

7. METODOLOGIA

7.1 Abordagem Metodológica

A metodologia adoptada neste projecto é de natureza experimental e aplicada, combinando técnicas de processamento de sinais, aprendizagem supervisionada e desenvolvimento de aplicações móveis.

7.2 Definição dos Requisitos

- Captação de áudio em tempo real através do microfone do dispositivo móvel.
- Processamento local para garantir privacidade e reduzir dependência de internet.

- Classificação em categorias: normal, bronquite e pneumonia.
- Interface simples e intuitiva para o utilizador.

7.3 Recolha e Preparação dos Dados

Foram considerados datasets públicos de tosse e sons respiratórios, como COUGHVID, Coswara, ICBHI e VIRUFY, amplamente utilizados na literatura científica (Orlandic, Teijeiro & Atienza, 2021). Os áudios são normalizados para 16 kHz, 16-bit e mono. É realizada remoção de ruído utilizando filtros digitais e técnicas de redução de ruído. Posteriormente, os sinais são segmentados para isolar eventos de tosse.

Foram considerados datasets públicos de tosse e sons respiratórios, como COUGHVID, Coswara, ICBHI e VIRUFY. Os áudios são normalizados para 16 kHz, 16-bit e mono. É realizada remoção de ruído utilizando filtros digitais e técnicas de redução de ruído. Posteriormente, os sinais são segmentados para isolar eventos de tosse.

7.4 Pré-processamento

O pré-processamento inclui:

- Normalização do volume
- Remoção de silêncio
- Filtragem de ruído
- Padronização da duração dos áudios

8. EXTRACÇÃO DE CARACTERÍSTICAS

A extracção de características é uma etapa crucial para transformar o sinal de áudio em dados utilizáveis pelos modelos de Machine Learning (Eyben, Wöllmer & Schuller, 2010).

8.1 MFCC (Mel Frequency Cepstral Coefficients)

Capturam características do timbre e do espectro da tosse, sendo amplamente utilizados em reconhecimento de fala e análise de áudio (Davis & Mermelstein, 1980).

8.2 Espectrogramas e Log-Mel Spectrogramas

Representam visualmente a frequência ao longo do tempo e são usados principalmente com Redes Neurais Convolucionais (Hershey et al., 2017).

A extracção de características é uma etapa crucial para transformar o sinal de áudio em dados utilizáveis pelos modelos de Machine Learning.

8.3 MFCC (Mel Frequency Cepstral Coefficients)

Capturam características do timbre e do espectro da tosse, sendo amplamente utilizados em reconhecimento de fala e análise de áudio.

8.4 Espectrogramas e Log-Mel Spectrogramas

Representam visualmente a frequência ao longo do tempo e são usados principalmente com Redes Neurais Convolucionais.

8.5 Zero Crossing Rate (ZCR)

Mede a taxa de mudança de sinal, útil para identificar explosividade e irregularidades da tosse.

8.6 Energia e Entropia do Sinal

A energia indica a intensidade da tosse, enquanto a entropia mede a complexidade do sinal.

8.7 Duração e Ritmo

A duração e o ritmo da tosse ajudam a diferenciar tipos de tosse seca, produtiva ou crónica.

9. SELECÇÃO E TREINO DO MODELO

9.1 Modelos Considerados

- Redes Neurais Convolucionais (CNN) (LeCun et al., 1998)
- Redes Recorrentes (RNN, LSTM) (Hochreiter & Schmidhuber, 1997)
- Modelos híbridos CNN + LSTM
- Transformers para áudio (Gong, Chung & Glass, 2021)

9.2 Justificação da Escolha

As CNNs são eficazes na análise de padrões visuais em espectrogramas, enquanto as LSTM capturam dependências temporais. Para dispositivos móveis, é recomendada uma CNN leve, como MobileNet adaptada para áudio (Howard et al., 2017).

9.3 Modelos Considerados

- Redes Neurais Convolucionais (CNN)
- Redes Recorrentes (RNN, LSTM)
- Modelos híbridos CNN + LSTM
- Transformers para áudio

9.4 Justificação da Escolha

As CNNs são eficazes na análise de padrões visuais em espectrogramas, enquanto as LSTM capturam dependências temporais. Para dispositivos móveis, é recomendada uma CNN leve, como MobileNet adaptada para áudio.

9.5 Processo de Treino

O treino é realizado em Python utilizando TensorFlow ou PyTorch. São definidos hiperparâmetros como taxa de aprendizagem, número de épocas, tamanho do batch e função de perda. O processo é acompanhado por validação para evitar overfitting.

```
Treinando...
Epoch 1/50
5/6 ----- 0s 486ms/step - accuracy: 0.4168 - loss: 1.1457
Epoch 1: val_accuracy improved from -inf to 0.12500, saving model to best_model.keras
6/6 ----- 9s 578ms/step - accuracy: 0.4361 - loss: 1.1344 - val_accuracy: 0.1250 - val_loss: 1.2960 - learning_rate: 0.0010
Epoch 2/50
5/6 ----- 0s 478ms/step - accuracy: 0.6042 - loss: 0.8854
Epoch 2: val_accuracy improved from 0.12500 to 0.62500, saving model to best_model.keras
6/6 ----- 3s 461ms/step - accuracy: 0.5913 - loss: 0.9053 - val_accuracy: 0.6250 - val_loss: 1.1232 - learning_rate: 0.0010
Epoch 3/50
5/6 ----- 0s 465ms/step - accuracy: 0.5929 - loss: 0.8196
Epoch 3: val_accuracy did not improve from 0.62500
6/6 ----- 3s 449ms/step - accuracy: 0.5939 - loss: 0.8272 - val_accuracy: 0.6250 - val_loss: 1.1331 - learning_rate: 0.0010
Epoch 4/50
5/6 ----- 0s 462ms/step - accuracy: 0.6523 - loss: 0.8876
Epoch 4: val_accuracy did not improve from 0.62500
6/6 ----- 5s 420ms/step - accuracy: 0.6540 - loss: 0.8767 - val_accuracy: 0.6250 - val_loss: 1.1894 - learning_rate: 0.0010
Epoch 5/50
5/6 ----- 0s 460ms/step - accuracy: 0.6768 - loss: 0.7293
Epoch 5: val_accuracy did not improve from 0.62500
6/6 ----- 3s 445ms/step - accuracy: 0.6715 - loss: 0.7291 - val_accuracy: 0.6250 - val_loss: 1.1083 - learning_rate: 0.0010
Epoch 6/50
5/6 ----- 0s 539ms/step - accuracy: 0.8198 - loss: 0.5322
Epoch 6: val_accuracy did not improve from 0.62500
6/6 ----- 5s 517ms/step - accuracy: 0.8003 - loss: 0.5544 - val_accuracy: 0.6250 - val_loss: 1.0226 - learning_rate: 0.0010
Epoch 7/50
5/6 ----- 0s 608ms/step - accuracy: 0.6853 - loss: 0.7246
Epoch 7: val_accuracy did not improve from 0.62500
```

```

6/6 ----- 5s 517ms/step - accuracy: 0.8003 - loss: 0.5544 - val_accuracy: 0.6250 - val_loss: 1.0226 - learning_rate: 0.0010
Epoch 7/50
... 5/6 ----- 0s 608ms/step - accuracy: 0.6853 - loss: 0.7246
Epoch 7: val_accuracy did not improve from 0.62500
6/6 ----- 3s 540ms/step - accuracy: 0.6971 - loss: 0.6913 - val_accuracy: 0.6250 - val_loss: 0.9949 - learning_rate: 0.0010
Epoch 8/50
5/6 ----- 0s 495ms/step - accuracy: 0.7428 - loss: 0.5387
Epoch 8: val_accuracy did not improve from 0.62500
6/6 ----- 5s 472ms/step - accuracy: 0.7418 - loss: 0.5420 - val_accuracy: 0.6250 - val_loss: 1.0200 - learning_rate: 0.0010
Epoch 9/50
5/6 ----- 0s 471ms/step - accuracy: 0.8070 - loss: 0.4705
Epoch 9: val_accuracy did not improve from 0.62500
6/6 ----- 3s 452ms/step - accuracy: 0.8000 - loss: 0.4878 - val_accuracy: 0.6250 - val_loss: 1.0126 - learning_rate: 0.0010
Epoch 10/50
5/6 ----- 0s 514ms/step - accuracy: 0.8211 - loss: 0.4411
Epoch 10: val_accuracy did not improve from 0.62500
6/6 ----- 5s 463ms/step - accuracy: 0.8172 - loss: 0.4434 - val_accuracy: 0.6250 - val_loss: 1.0006 - learning_rate: 0.0010
Epoch 11/50
5/6 ----- 0s 482ms/step - accuracy: 0.8097 - loss: 0.4137
Epoch 11: val_accuracy did not improve from 0.62500
6/6 ----- 5s 466ms/step - accuracy: 0.8090 - loss: 0.4231 - val_accuracy: 0.5893 - val_loss: 1.0031 - learning_rate: 0.0010
Epoch 12/50
5/6 ----- 0s 484ms/step - accuracy: 0.7324 - loss: 0.5160
Epoch 12: ReduceLROnPlateau reducing learning rate to 0.000500000237487257.

Epoch 12: val_accuracy did not improve from 0.62500
6/6 ----- 3s 463ms/step - accuracy: 0.7414 - loss: 0.5103 - val_accuracy: 0.5893 - val_loss: 1.0120 - learning_rate: 0.0010
Epoch 13/50
5/6 ----- 0s 728ms/step - accuracy: 0.8351 - loss: 0.4030

```

```

Epoch 31/50
... 6/6 ----- 0s 560ms/step - accuracy: 0.8786 - loss: 0.2431
Epoch 31: val_accuracy did not improve from 0.71429
6/6 ----- 4s 637ms/step - accuracy: 0.8826 - loss: 0.2406 - val_accuracy: 0.6429 - val_loss: 0.9343 - learning_rate: 2.5000e-04
Epoch 32/50
5/6 ----- 0s 466ms/step - accuracy: 0.9431 - loss: 0.1737
Epoch 32: val_accuracy did not improve from 0.71429
6/6 ----- 3s 444ms/step - accuracy: 0.9363 - loss: 0.1894 - val_accuracy: 0.6429 - val_loss: 0.9419 - learning_rate: 2.5000e-04
Epoch 33/50
5/6 ----- 0s 873ms/step - accuracy: 0.9333 - loss: 0.1800
Epoch 33: val_accuracy did not improve from 0.71429
6/6 ----- 5s 857ms/step - accuracy: 0.9346 - loss: 0.1815 - val_accuracy: 0.6071 - val_loss: 0.9693 - learning_rate: 2.5000e-04
Epoch 34/50
5/6 ----- 1s 1s/step - accuracy: 0.9411 - loss: 0.1781
Epoch 34: ReduceLROnPlateau reducing learning rate to 0.0001250000059371814.

Epoch 34: val_accuracy did not improve from 0.71429
6/6 ----- 6s 1s/step - accuracy: 0.9402 - loss: 0.1814 - val_accuracy: 0.5893 - val_loss: 0.9969 - learning_rate: 2.5000e-04
Epoch 35/50
5/6 ----- 0s 471ms/step - accuracy: 0.9579 - loss: 0.1710
Epoch 35: val_accuracy did not improve from 0.71429
6/6 ----- 3s 453ms/step - accuracy: 0.9575 - loss: 0.1718 - val_accuracy: 0.5893 - val_loss: 1.0056 - learning_rate: 1.2500e-04
Epoch 36/50
5/6 ----- 0s 466ms/step - accuracy: 0.9321 - loss: 0.1715
Epoch 36: val_accuracy did not improve from 0.71429
6/6 ----- 5s 449ms/step - accuracy: 0.9320 - loss: 0.1723 - val_accuracy: 0.5536 - val_loss: 1.0193 - learning_rate: 1.2500e-04
Epoch 36: early stopping
Restoring model weights from the end of the best epoch: 24.

```

10. DIVISÃO DOS DADOS

Os dados são divididos em três subconjuntos:

- 70% para treino
- 15% para validação
- 20% para teste

Esta divisão garante que o modelo seja avaliado em dados não vistos, assegurando capacidade de generalização.

11. TREINO DO MODELO

11.1 Definição dos Requisitos

- Captação de áudio em tempo real.
- Processamento local para garantir privacidade.
- Classificação em normal, bronquite ou pneumonia.

11.2 Recolha e Preparação dos Dados

Foram considerados datasets públicos de tosse e sons respiratórios, como COUGHVID, Coswara e ICBHI. Os áudios são normalizados para 16 kHz, 16-bit e mono. É realizada remoção de ruído e segmentação dos sinais.

```
[3] print("\n" + "="*70)
✓ 3 s print("BAIXANDO DATASET ICBHI 2017 via kagglehub")
print("="*70)

try:
    print("📦 Baixando Respiratory Sound Database...")
    dataset_path = kagglehub.dataset_download("vbookshelf/respiratory-sound-database")
    print(f"✅ Dataset em: {dataset_path}")
except Exception as e:
    print(f"❌ Erro ao baixar: {e}")
    dataset_path = None
```

...

```
=====
BAIXANDO DATASET ICBHI 2017 via kagglehub
=====
📦 Baixando Respiratory Sound Database...
Using Colab cache for faster access to the 'respiratory-sound-database' dataset.
✅ Dataset em: /kaggle/input/respiratory-sound-database
```

12. EXTRACÇÃO DE CARACTERÍSTICAS

A extracção de características é uma etapa crucial para transformar o sinal de áudio em dados utilizáveis pelos modelos de Machine Learning.

12.1 MFCC (Mel Frequency Cepstral Coefficients)

Capturam características do timbre e do espectro da tosse.

12.2 Espectrogramas

Representam visualmente a frequência ao longo do tempo e são usados principalmente com CNNs.

12.3 Zero Crossing Rate (ZCR)

Mede a taxa de mudança de sinal, útil para identificar explosividade da tosse.

12.4 Outras Características

- Duração da tosse
- Energia do sinal
- Ritmo

13. SELECÇÃO E TREINO DO MODELO

13.1 Modelos Considerados

- CNN (Convolutional Neural Networks)
- RNN / LSTM
- CNN + LSTM
- Transformers para áudio

13.2 Modelo Recomendado

Uma CNN leve, como MobileNetV2 adaptada para áudio, devido à sua eficiência em dispositivos móveis.

14. DIVISÃO DOS DADOS

Os dados são divididos em:

- 70% Treino
- 15% Validação
- 15% Teste

Esta divisão garante avaliação justa do desempenho do modelo.

15. TREINO DO MODELO

O treino é realizado em Python utilizando bibliotecas como TensorFlow ou PyTorch. São ajustados hiperparâmetros como taxa de aprendizagem, número de épocas e tamanho do batch.

16. AVALIAÇÃO DO MODELO

16.1 Métricas Utilizadas

- ✓ Accuracy (Precisão)
- ✓ Precision
- ✓ Recall
- ✓ F1-score
- ✓ Matriz de Confusão
- ✓ Curva ROC (One-vs-Rest)

Estas métricas são amplamente utilizadas para avaliação de modelos de classificação (Powers, 2011).

16.2 Interpretação dos Resultados

Um bom modelo deve apresentar alta precisão e recall equilibrados, indicando capacidade de identificar correctamente casos positivos sem gerar muitos falsos alarmes.

```

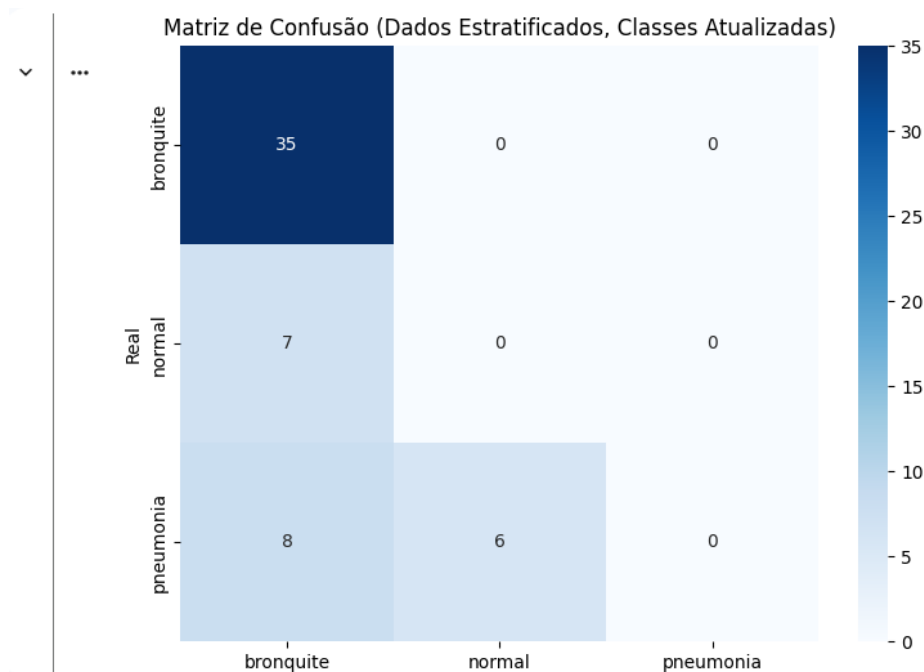
... AVALIAÇÃO FINAL DO MODELO ESTRATIFICADO COM CLASSES ATUALIZADAS
=====
Acurácia (estratificada, classes atualizadas): 0.6250

Classification Report (estratificado, classes atualizadas):
              precision    recall  f1-score   support

   bronquite      0.7000      1.0000      0.8235        35
      normal      0.0000      0.0000      0.0000         7
   pneumonia      0.0000      0.0000      0.0000        14

   accuracy                   0.6250        56
  macro avg      0.2333      0.3333      0.2745        56
 weighted avg      0.4375      0.6250      0.5147        56

AUC por classe (OvR, estratificado, classes atualizadas):
bronquite   : 0.7932
normal      : 0.4490
pneumonia   : 0.8231
=====
```



17. DESENVOLVIMENTO DA APLICAÇÃO MÓVEL

17.1 Arquitectura da Aplicação

A aplicação segue uma arquitectura em camadas, separando interface, lógica de negócio e processamento de dados.

17.2 Front-end

Pode ser desenvolvido em Flutter, React Native ou nativo (Kotlin/Swift), garantindo compatibilidade multiplataforma.

17.3 Captação de Áudio

Utiliza bibliotecas específicas para gravação do som da tosse, assegurando qualidade e baixa latência.

17.4 Integração do Modelo

O modelo treinado é integrado localmente utilizando TensorFlow Lite, permitindo inferência no dispositivo.

18. ARQUITETURA DO SISTEMA

18.1 Front-end

Usou-se o framework React Expo, uma framework baseado no React Native, que permite a criação de aplicativos multiplataforma (Android e IOS).

18.2 Captação de Áudio

Utiliza bibliotecas nativas para gravação do som da tosse.

18.3 Integração do Modelo

O modelo treinado foi integrado a uma api, Fastapi hospedado no provedor de serviços em cloud, Render, por ser um provedor adequado para testes e permite hospedagem gratuita.

A api expõe um endpoint que será acessado através do app react expo. Funciona da seguinte forma:~

O usuário grava o áudio da tosse e após a gravação o áudio é enviado diretamente na rota destinada para predição dos sons da tosse e a api devolve um POST contendo o resultado (Confiança e classe prevista ” Bronquite ou Pneumonia”).

19. ARQUITETURA DO SISTEMA

1. Captura do áudio
2. Pré-processamento
3. Extracção de características
4. Classificação pelo modelo
5. Exibição do resultado
- 6.

20. BACKEND

O backend será implementado para:

- Análise estatística
- Leitura e tratamento dos áudios
- Monitorização de uso

21. RESULTADOS ESPERADOS

Pode ser implementado um backend para:

- Leitura e tratamento dos áudios
- Análise estatística
- Monitorização

Tecnologias: Python, FastAPI.

22. RESULTADOS ESPERADOS

- Identificação automática de padrões de tosse
- Apoio à triagem de pacientes
- Redução do tempo de diagnóstico preliminar

23. BOAS PRÁTICAS E CONSIDERAÇÕES ÉTICAS

- Garantir consentimento informado dos utilizadores.
- Proteger dados pessoais e sensíveis.
- Cumprir princípios de confidencialidade.
- Informar claramente que a aplicação não substitui diagnóstico médico.

24. LIMITAÇÕES DO ESTUDO

- Dependência da qualidade do microfone.
- Variabilidade entre indivíduos.
- Possível ruído ambiental.

25. CONCLUSÃO

O projecto demonstra o potencial da Aprendizagem Computacional aplicada à saúde, especificamente na análise do som da tosse para identificação de pneumonia e bronquite. A solução proposta pode ser uma ferramenta de apoio importante, especialmente em regiões com acesso limitado a serviços de saúde.

26. REFERÊNCIAS BIBLIOGRÁFICA

- Abeyratne, U. R., Swarnkar, V., Setyati, A., & Triasih, R. (2013). Cough sound analysis can rapidly diagnose childhood pneumonia. *Annals of Biomedical Engineering*, 41(11), 2448–2462. <https://doi.org/10.1007/s10439-013-0836-0>
- Brown, C., Chauhan, J., Grammenos, A., Han, J., Hasthanasombat, A., Spathis, D., ... & Mascolo, C. (2020). Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, 3474–3484.
- Davis, S., & Mermelstein, P. (1980). Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 28(4), 357–366.
- Eyben, F., Wöllmer, M., & Schuller, B. (2010). Opensmile: The Munich versatile and fast opensource audio feature extractor. *Proceedings of the 18th ACM International Conference on Multimedia*, 1459–1462.
- Gong, Y., Chung, Y.-A., & Glass, J. (2021). AST: Audio Spectrogram Transformer. *Proceedings of Interspeech 2021*, 571–575.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT Press.
- Hershey, S., Chaudhuri, S., Ellis, D. P. W., Gemmeke, J. F., Jansen, A., Moore, R. C., ... & Wilson, K. (2017). CNN architectures for large-scale audio classification. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 131–135.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735–1780.
- Howard, A. G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., ... & Adam, H. (2017). MobileNets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*.
- Imran, A., Posokhova, I., Qureshi, H. N., Masood, U., Riaz, M. S., Ali, K., ... & Nabeel, M. (2020). AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app. *Informatics in Medicine Unlocked*, 20, 100378.

- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278–2324.
- Mitchell, T. M. (1997). *Machine learning*. McGraw-Hill.
- Oppenheim, A. V., & Schafer, R. W. (2010). *Discrete-time signal processing* (3rd ed.). Pearson.
- Orlandic, L., Teijeiro, T., & Atienza, D. (2021). The COUGHVID crowdsourcing dataset: A corpus for the study of large-scale cough analysis algorithms. *Scientific Data*, 8, 156.
- Powers, D. M. W. (2011). Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2(1), 37–63.
- World Health Organization. (2022). *Pneumonia*. WHO.
- Abeyratne, U. R., Swarnkar, V., Setyati, A., & Triasih, R. (2013). Cough sound analysis can rapidly diagnose childhood pneumonia. *Annals of Biomedical Engineering*, 41(11), 2448–2462.
- Brown, C., Chauhan, J., Grammenos, A., Han, J., Hasthanasombat, A., Spathis, D., ... & Mascolo, C. (2020). Exploring automatic diagnosis of COVID-19 from crowdsourced respiratory sound data. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*.
- Imran, A., Posokhova, I., Qureshi, H. N., Masood, U., Riaz, M. S., Ali, K., ... & Nabeel, M. (2020). AI4COVID-19: AI enabled preliminary diagnosis for COVID-19 from cough samples via an app. *Informatics in Medicine Unlocked*, 20, 100378.
- Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. *International Conference on Learning Representations*.
- TensorFlow. (2024). TensorFlow Lite documentation. <https://www.tensorflow.org/lite>
- PyTorch. (2024). PyTorch documentation. <https://pytorch.org/docs>

Anexos

início

Olá!

Teste possíveis sinais de pneumonia ou
bronquite, gravando a sua tosse



Clique para começar →



Este aplicativo não substitui a consulta médica.



Início



Informações de saúde



Sobre

Ilustração 1 - Tela principal



Ilustração 2- Tela de gravação de áudio

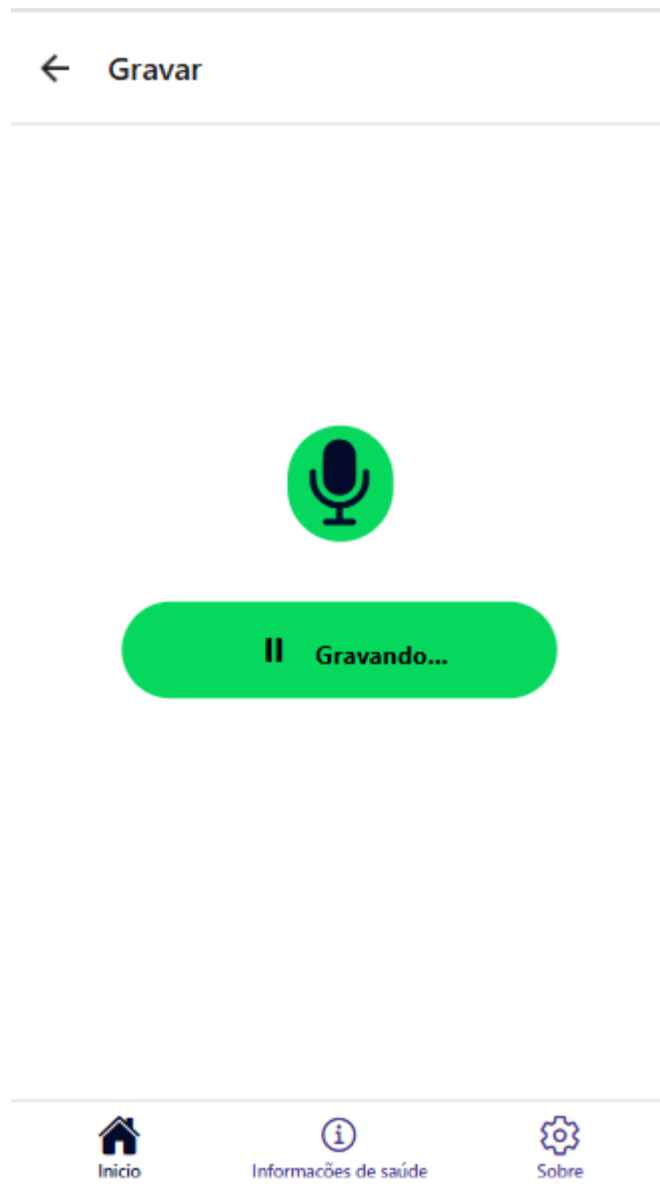


Ilustração 3 - Gravando áudio

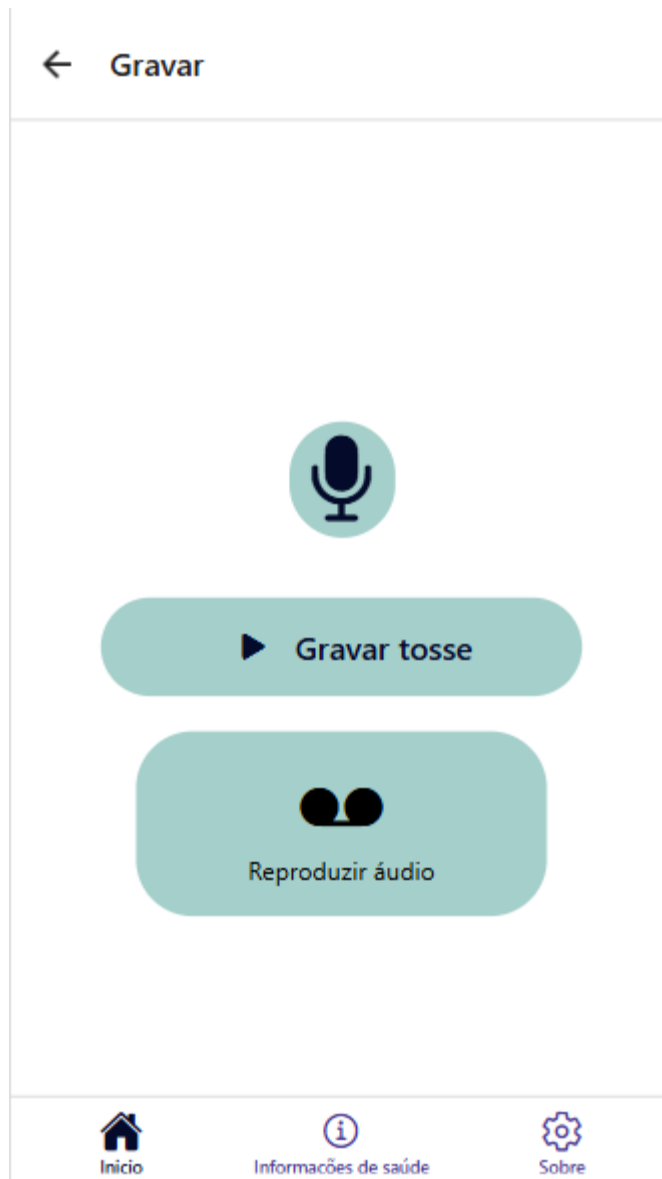


Ilustração 4 - Áudio gravado e pronto a reproduzir.

Informações de saúde

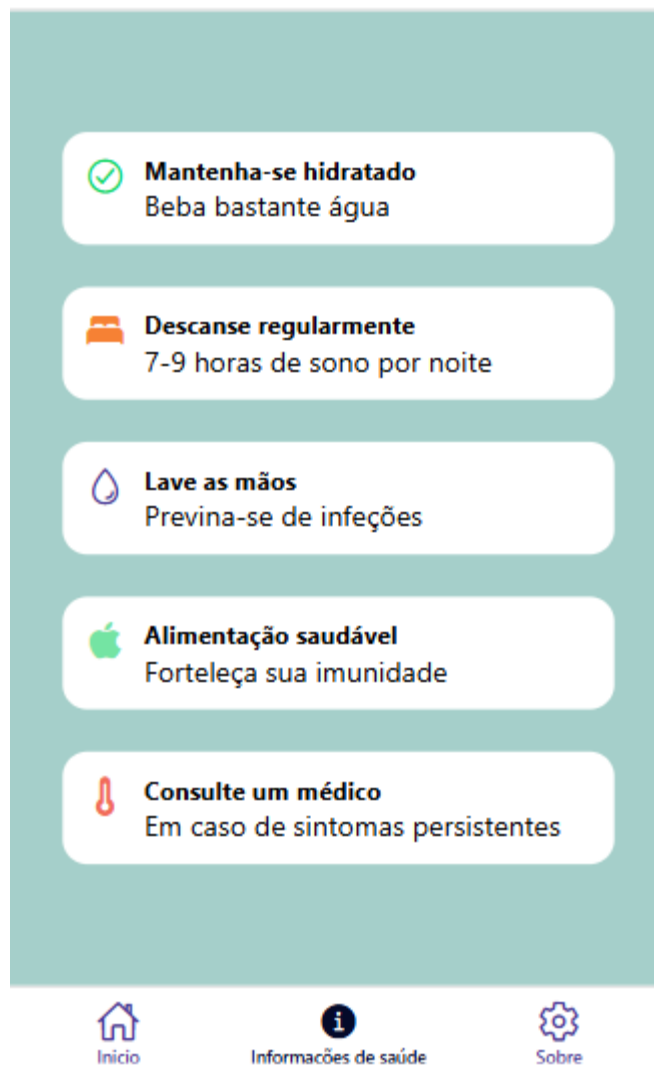


Ilustração 5 - Recomendações

Sobre



Tecnologia a favor da saúde.

Versão 1.0.0

Contacto ou suporte



araujo.io.dev@gmail.com

Projeto desenvolvido por estudantes do IPH
Instituto Politécnico da Huíla



Início



Informações de saúde



Sobre

Ilustração 6 - Sobre o aplicativo