Detecção de Choro de Bebê Utilizando Redes Neurais: Uma Abordagem Prática

Georgino da Silva Baltazar
National Institute of Telecommunications - Inatel
Santa Rita do Sapucaí - Brazil
georginosilva@mtel.inatel.br

***Resumo* — Este estudo investigou o desempenho de dois modelos de Redes Neurais Convolutivas (CNNs) na classificação de sinais de áudio em "chorando" e "não chorando". O primeiro modelo alcançou uma acurácia de 95,81%, enquanto o segundo, mais avançado, atingiu 95,29%. Ambos os modelos foram treinados em um conjunto de dados, demonstrando a capacidade das CNNs em lidar com tarefas de classificação de áudio. Os resultados da matriz de confusão destacaram a eficácia do segundo modelo, fornecendo insights valiosos para aplicações futuras. Este estudo contribui para o entendimento das potencialidades das CNNs na análise de áudio para detecção de choro.**

Keywords: Detecção de Choro de Bebê; Redes Neurais Convolucionais (CNNs); Processamento de Sinais de Áudio; Classificação de Áudio Infantil.

I. INTRODUCTION

O avanço da inteligência artificial e do aprendizado de máquina tem desempenhado um papel crucial na transformação de diversas áreas, proporcionando soluções inovadoras e eficientes para desafios complexos. Um desses desafios abrange a detecção e interpretação de sinais acústicos em contextos específicos, como a identificação do choro de bebês [1][2].

O choro é uma das principais formas de comunicação dos bebês, sendo uma indicação vital de suas necessidades e estados emocionais. Pode ser considerado como o comportamento natural de um bebê na construção da interação social com seu cuidador [2][3]. A capacidade de detectar automaticamente se um bebê está chorando ou não tem implicações significativas em ambientes como creches, monitoramento remoto de bebês, e até mesmo no apoio a pais e cuidadores. Abaixo a descrição de alguns cenários para destacar ainda mais a relevancia da detecção automatizada de choro de bebê:

Monitoramento Doméstico: Possibilita aos pais monitorar o bem-estar de seus filhos, especialmente durante períodos de sono ou quando fora do campo de visão direto.

Creches e Ambientes Infantis: Facilita o gerenciamento de grupos de crianças, permitindo respostas personalizadas às demandas individuais.

Dispositivos de Monitoramento Infantil: Pode ser integrado a dispositivos de monitoramento para alertar sobre eventos significativos, como choro prolongado.

Neste contexto, a aplicação de técnicas de aprendizado de máquina torna-se uma abordagem promissora. Este artigo apresenta o desenvolvimento de um sistema baseado em redes neurais convolucionais (CNNs) para a detecção automatizada de choro de bebês. A escolha de utilizar Redes Neurais Convolucionais é motivada pela capacidade dessas redes em capturar padrões complexos em dados temporais como áudio, tornando-as particularmente adequadas para a análise de sinais acústicos, como os presentes no choro infantil[4].

Uma CNN representa um modelo de aprendizado profundo especialmente concebido para a análise de dados visuais, como imagens, vídeos e representações de áudio em tempo-frequência. A extração de características cruciais ocorre por meio de operações de convolução e pooling, aplicadas sucessivamente em cada camada. À medida que avançamos nas camadas, a resolução espacial diminui, permitindo a obtenção de características abstratas mais profundas. A essência da CNN reside na exploração da conectividade local, compartilhamento de pesos, pooling e na incorporação de várias camadas [5][6].

A aplicação desta arquitetura no âmbito da detecção de choro infantil visa capturar características sutis e complexas presentes nos sinais de áudio, proporcionando uma abordagem mais sofisticada em comparação com métodos convencionais. Este projeto tem como objetivo desenvolver um sistema prático e eficaz de detecção de choro de bebê, empregando uma abordagem de aprendizado de máquina usando a plataforma EDGE Impulse que facilita o treinamento, a implantação e a integração do modelo em ambientes diversos. Ao explorar o potencial das CNNs, este estudo permiti identificar o choro de bebês em diferentes contextos sonoros e ambientes diversos.

O restante deste artigo está organizado da seguinte forma: A Seção II descreve alguns trabalhos relacionados; O modelo do sistema é apresentado na Seção III; Na Seção IV, descrevemos ametodologia utilizada; A Seção V apresenta a nalálise dos resultados; finalmente, a Seção VI apresenta a conclusão e trabalhos futuros.

II. TRABALHOS RELACIONADOS

A detecção de choro de bebê tem sido uma área de pesquisa ativa, com diversas abordagens sendo exploradas. Métodos clássicos geralmente envolvem a extração manual de características, enquanto abordagens mais recentes, como o uso de redes neurais, aproveitam a capacidade do modelo em aprender automaticamente a partir dos dados. Abordagens existentes na literatura de detecção de choro de bebê consistem na extração de características significativas de quadros de sinal

de áudio. A maioria delas utiliza características espectrais, como coeficientes cepstrais de frequência mel (MFCCs), combinadas com classificadores binários, como máquinas de vetores de suporte (SVMs) [7]. Pesquisas recentes têm explorado o uso de redes neurais convolucionais (CNNs) adaptadas à detecção de choro de bebê [8]. Em [9] os autores propuseram uma rede baseada em CNN para o reconhecimento de pré-choro infantil, obtendo resultados de precisão superiores em comparação com abordagens baseadas em características de baixo nível. O estudo proposto em [10] utiliza extração de caraterísticas para analisar os sinais do choro infantil, onde os autores extrairam 12 ordens de consficientes de coeficientes cepstrais de frequência mel (MFCC) para o desenvolvimento do modelo. Exitem vários estudos que abordam a deteção de choro infantil, a maioria deles procuram classificar esses choros por tipo, esses estudos focam em abordagens para prever sensações fisiológicas, como fome, dor, troca de fralda e desconforto [7][8], [11]-[13].

Dirferente da maioria modelos propostos anteriormente, nosso estudo, propõe um sistema de deteção de choro em diferentes ambientes, ou seja, o modelo apresentado aqui, classifica apena o choro e o não choro. Na próxima sessão apresntaremos o modelo do sistema.

III. MODELO DO SISTEMA

O sistema de detecção de choro proposto utiliza uma abordagem baseada em aprendizado profundo, empregando uma CNN para análise de áudio. A seguir, descrevemos detalhadamente cada componente do modelo, desde a captura de áudio até a decisão de detecção:

1. Captura de Áudio:

Nesta fase inicial, o sistema captura áudio ambiente por meio de um dispositivo de entrada, como um microfone. Esse sinal de áudio bruto é crucial para a identificação de padrões sonoros associados ao choro de um bebê.

2. Pré-Processamento de Áudio:

O áudio capturado passa por uma etapa de pré-processamento essencial. Nesse estágio, a transformação do sinal de áudio em um espectrograma é realizada. Esse espectrograma representa visualmente as características temporais e de frequência do áudio, oferecendo uma representação adequada para a análise subsequente.

3. Entrada do Modelo CNN (Dataset):

Os dados pré-processados, geralmente na forma de espectrogramas, são fornecidos como entrada para a Convolutional Neural Network (CNN). O modelo é treinado utilizando um conjunto de dados robusto e diversificado, contendo exemplos de áudio com e sem choro. Esse conjunto de dados é fundamental para capacitar a CNN a discernir padrões distintivos associados ao choro de um bebê.

4. Processamento pelo Modelo CNN:

A entrada do modelo passa por várias camadas convolucionais, de pooling e densas da CNN. Essas camadas são projetadas para aprender automaticamente características hierárquicas do espectrograma, destacando padrões relevantes para a detecção de choro.

5. Saída do Modelo:

A última camada da CNN produz uma saída que reflete as previsões do modelo. Em um problema de classificação binária, como a detecção de choro, a função de ativação softmax geralmente é empregada para gerar probabilidades associadas a cada classe.

6. Decisão de Detecção:

Com base nas probabilidades geradas pela saída do modelo, uma decisão de detecção é tomada. Um limiar pode ser definido para determinar se o choro está presente ou não. Se a probabilidade de choro for superior ao limiar, o sistema emite a decisão de que o choro foi detectado.

7. Resultado Final:

O resultado final da detecção é indicado, fornecendo uma resposta clara sobre a presença ou ausência de choro. Este resultado é obtido após a análise cuidadosa do espectrograma e as decisões tomadas pelas camadas da CNN.

IV. METODOLOGIA

A metodologia adotada neste projeto envolveu diversas etapas, desde a aquisição dos dados até a avaliação do desempenho do modelo. A seguir, detalhamos cada uma dessas etapas, proporcionando uma visão abrangente do processo de desenvolvimento e validação do sistema de detecção de choro.

Aquisição e Preparação do Conjunto de Dados: A base fundamental deste projeto é o conjunto de dados, essencial para treinar e avaliar o modelo de detecção de choro. Foi feita a coleta de amostras de áudio contendo sons de choro, e uma varidedade de outros sons, como de cahorros, gatos, passarinhos, sirenes, carros, motos, natureza e períodos de silêncio, essenciais para o treinamento do modelo. O conjunto de dados foi organizado de maneira a conter áudios que representam situações reais de choro e não choro de bebês, ou seja, as gravações foram coletadas de forma diversificada, considerando diferentes ambientes sonoros e condições.

Pré-Processamento de Dados: Dividimos o conjunto de dados em duas classes, “chorando” e “não chorando”. Utilizando a interface amigável do *Edge Impulse*, as amostras foram incorporadas à plataforma, que, por meio de sua funcionalidade de pré-processamento, normalizou e converteu-se os áudios em espectrogramas, uma representação visual das características temporais e de frequência. A normalização dos dados também foi realizada para garantir uma entrada

consistente para o modelo um formato compatível com os requisitos do modelo.

Arquitetura do Modelo CNN: Foi adotada uma arquitetura de CNN, conhecida por sua eficácia no processamento de dados espaciais, como imagens. A arquitetura foi projetada considerando camadas convolucionais, de pooling e densas para aprender padrões complexos presentes nos espectrogramas.

Treinamento do Modelo: O modelo foi treinado utilizando o conjunto de dados preparado, onde envolveu a divisão do conjunto de dados em conjuntos de treinamento e teste. No Edge Impulse, configuramos os parâmetros do modelo, como o tipo de arquitetura de rede neural a ser utilizada. A plataforma facilitou o processo de treinamento do modelo, ajustando os pesos da rede com base nos dados fornecidos. O treinamento ocorreu por um número específico de épocas, com a avaliação constante do desempenho no conjunto de validação.

Avaliação do Modelo: Após o treinamento, o desempenho do modelo foi avaliado utilizando o conjunto de validação. Métricas como acurácia, perda e matriz de confusão foram utilizadas para mensurar a eficácia do sistema na detecção de choro.

Exportação para o Ambiente Colab, Integração e Análise Detalhada: Após o treinamento no Edge Impulse, exportamos o modelo treinado para o ambiente Colab. Essa migração foi crucial para uma análise mais detalhada e flexível do modelo. A facilidade de integração do Colab com bibliotecas como Librosa permitiu uma exploração mais profunda dos dados de áudio.

Análise Detalhada no Ambiente Colab(Visualização de Espectrogramas): A primeira análise no Colab envolveu a visualização de espectrogramas gerados a partir dos áudios, as figuras 1 e 2, mostram exemplos de espectrogramas gerados de alguns audios com a devida classificação. Utilizamos a biblioteca Librosa para converter os áudios em representações visuais, proporcionando uma compreensão mais profunda das características temporais e de frequência que o modelo estava aprendendo.

Análise Detalhada no Ambiente Colab (Curvas de Aprendizado e Matriz de Confusão): Para avaliar o treinamento do modelo, plotamos curvas de aprendizado exibindo a evolução da acurácia e da perda durante as épocas. Isso permitiu uma compreensão mais clara de como o modelo

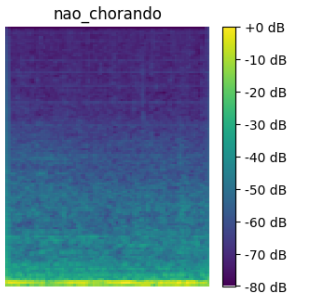

Fig 1. Espectrograma de não choro

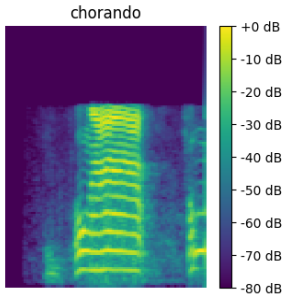

Fig2. Espectrograma de choro

estava convergindo e se havia sinais de overfitting ou underfitting. Plotamos também a matriz de confusão que permitiu fazer uma avaliação mais profunda das previsões do modelo.

A exportação para o ambiente Colab proporcionou uma transição suave do treinamento na plataforma Edge Impulse para análises mais detalhadas. Essa abordagem híbrida, combinando a facilidade de uso do Edge Impulse com a flexibilidade do Colab, demonstrou ser vantajosa para uma compreensão mais abrangente do sistema de detecção de choro em áudios infantis.

V. Resultados e Análise

Nesta seção, apresentamos uma análise detalhada dos resultados obtidos com o modelo de detecção de choro. Avalaiamos dois tipos de arquiteturas diferentes para construção do modelo, as configurações ulizadas em que cada um deles serão mostradas nas Próximas sessões. Antes, descrevemos o dataset (conjunto de dados) utilizado no nosso projeto.

Dataset: O *dataset*, ou conjunto de dados utilizado neste projeto, é composto por 953 amostras de áudios diversos, esses dados foram divididos em 80% para treinanmento (675) e

20% para testes (169), sendo 634 classificados como “Choro” e 319 “Não Choro” composto por uma diversidade de sons representando o ambiente real. A maioria dos dados foram coletados na internet no site da freesound [11], uma plataforma sem fins lucrativos. O Freesound tem como objetivo criar um enorme banco de dados colaborativo de trechos de áudio, amostras, gravações e todos os tipos de bipes tem diversos sons inclusive choro de bebês. Por outro lado, gravamos também algumas amostras de áudios utilizando celular com sistema android. Todos os audios foram convertidos para o formato WAV.

5.1 Configurações Paramétricas

O modelo principal o mesmo que foi utilizado no Edge impulse tem as seguintes configurações:

Camada de Entrada (InputLayer): recebe dados com a forma (X_train.shape[1], X_train.shape[2]), que representam a dimensionalidade dos nossos dados de treinamento.

Camadas Conv1D: Duas camadas Conv1D são empregadas para aprender padrões locais nos dados. A primeira camada tem 8 filtros (neurônios), cada um com um kernel de tamanho 3 e ativação ReLU. A segunda camada tem 16 filtros com configurações semelhantes. A técnica de padding 'same' é usada para garantir que a saída das camadas Conv1D tenha a mesma largura que a entrada. Camadas de Pooling (MaxPooling1D): Camadas de pooling ajudam a reduzir a dimensionalidade e o custo computacional. Duas camadas MaxPooling1D com tamanho de pool 2 e passo 2 são usadas para realizar o downsampling após as camadas Conv1D. Camadas de Dropout: Camadas de dropout são inseridas para prevenir overfitting durante o treinamento. O dropout de 25% após as camadas Conv1D e MaxPooling1D ajuda a regularizar o modelo.

Camada Flatten: A camada Flatten é responsável por converter o tensor de saída das camadas convolucionais em um vetor unidimensional, preparando os dados para a entrada nas camadas totalmente conectadas. Filnamente a Camada Dense (Totalmente Conectada): essa camada é a final, tem 2 neurônios (um para cada classe - 'Chorando' ou 'Não Chorando') com ativação softmax, que é adequada para tarefas de classificação multiclasse.

De forma simplificada apresentamos as configurações utilizadas para Construção do modelo 2:

Camadas Convolucionais: Três camadas Conv2D com ativação ReLU, que aprendem características visuais hierárquicas. Camadas de Pooling: Três camadas MaxPooling2D para reduzir a dimensionalidade e preservar características importantes. Camada Flatten: Transforma os mapas de características 2D em um vetor unidimensional.

Camada Dense: Uma camada Dense com 128 neurônios e ativação ReLU, que realiza aprendizado de características mais abstratas. Camada de Dropout: Dropout após a camada Dense, ajudando a evitar overfitting. Camada de Saída: Uma camada Dense com 1 neurônio e ativação sigmoid, adequada para tarefas de classificação binária. A tabela I, mostra os parámetros gerais de uma forma simplificada utilizados em cada um dos modelos.

Parâmetro	Modelo 1	Modelo 2
Epochs	100	100
Learning_rate	0,005	0,0001
Dropout_rate	0.25	0.3
Batch_size	32	32
Optimizer	Adam	Adam

Tabela I. Parámetros gerais utilizados

5.2 Acurácia

As figuras 3 e 4, mostram as curvas de desenpenho dos modelos 1 e 2 respetivamente, no modelo 1, a Acurácia no Conjunto de validação atingiu proximadamente 95.81% , e no modelo 2 praticamente a mesma com uma ligeira vantagem do modelo 1, ou seja atingiu 95.29% no conjunto de valiação.No entanto, observando a figura 3, nota um pouco de falta de convergencia, daí que decidimos testar outra configuração, que por sua vez coverge melhor. Esses resultados indicam que ambos os modelos conseguiram aprender eficientemente os padrões presentes nos dados de treinamento e generalizaram bem para novos dados.

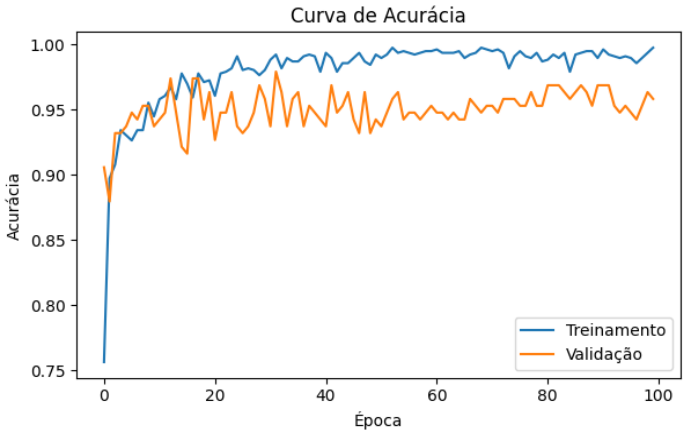

Fig. 3 - Acurácia do Modelo 1

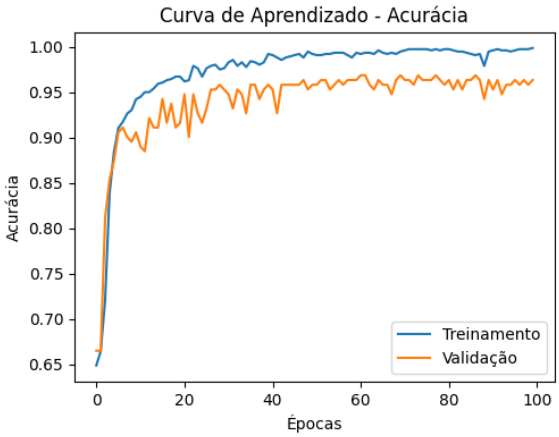

Fig. 4 - Acurácia do Modelo 2

5.3 Perda

As figuras 5 e 6, mostram as curvas da perda, observa-se que a a curva de treinamento do modelo1 vai aumentando depois de 20 épocas de treinamento, dando indices de sobreajuste no modelo, com 100 épocas de treinamento o modelo 1, atingiu 0,0375 de perda, já o modelo 2, como mostra a figura 6, apresenta uma perda significativamente menor (0.0281) em comparação com o Modelo 1 (0.3075). Essa redução na perda indica uma melhoria na capacidade do modelo de fazer previsões precisas e, consequentemente, sugere uma melhor generalização para dados não vistos.

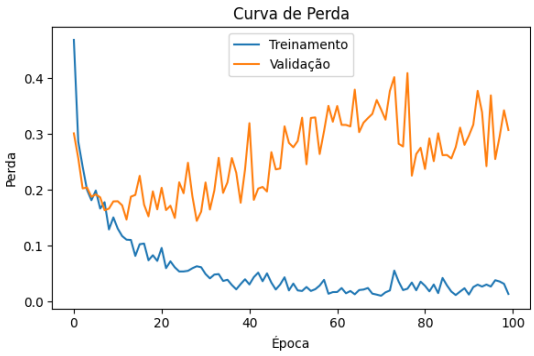

Fig. 5 - Curva da Perda Modelo 1

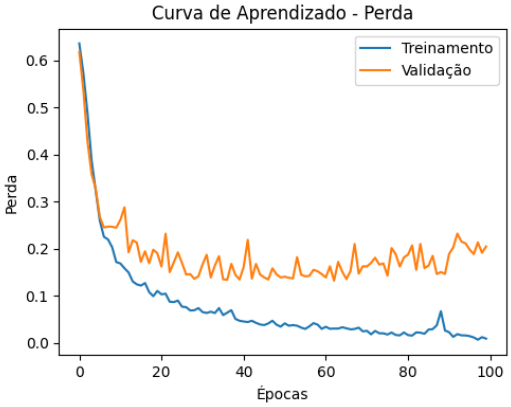

Fig. 6 - Curva da Perda Modelo 2

5.4 Matriz de Confusão

As figuras 7 e 8, mostram a matriz de confusão de cada modelo. A matriz de confusão é uma tabela que permite a visualização do desempenho de um algoritmo de classificação. Ela é especialmente útil em problemas de classificação binária, como o caso em que temos duas classes, como "positivo" e "negativo", "sim" e "não", ou, no contexto do nosso modelo, "chorando" e "não chorando". Como pode-se observar as figuras, o modelo 2 demonstrou uma melhoria em termos de falsos positivos (0 FP), indicando que o modelo aprimorado teve um desempenho superior na identificação correta dos eventos de interesse (choro).

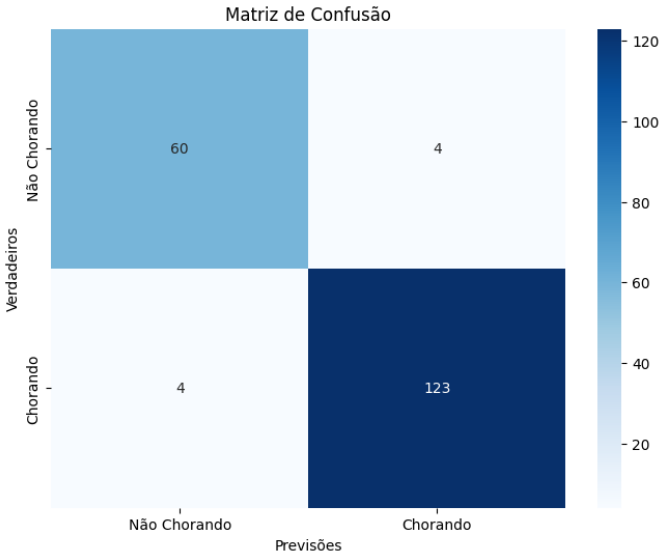

Fig. 7 – Matriz de Confusão Modelo 1

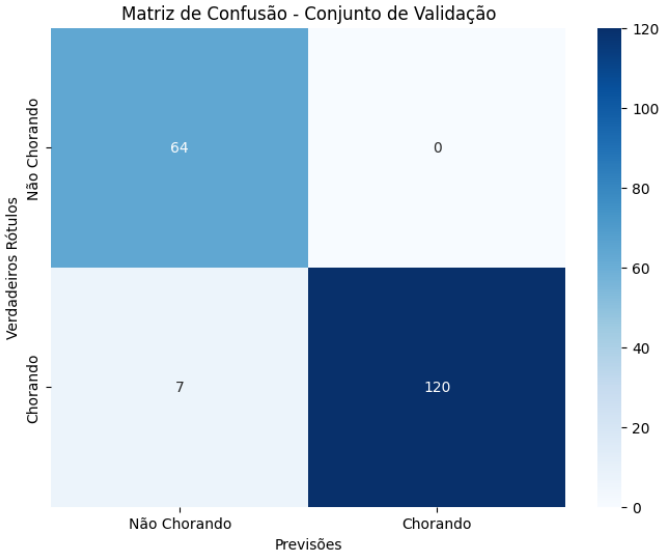

Fig. 8 – Matriz de Confusão Modelo 2

VI. CONCLUSÃO

Este artigo apresentou uma implementação eficaz de um sistema de detecção de choro em bebês usando CNNs e MFCCs. A abordagem oferece uma base sólida para futuras melhorias e aplicações práticas, contribuindo para a melhoria do suporte automatizado aos cuidados com o bebê.

O primeiro modelo demonstrou uma acurácia notável de aproximadamente 95.81% no conjunto de validação. A matriz de confusão revelou uma taxa de verdadeiros positivos (VP) de 123 e verdadeiros negativos (VN) de 60, evidenciando a capacidade do modelo em identificar corretamente as duas classes. Por outro lado, o segundo modelo, apresentou uma performance ligeiramente menor que a do modelo 1., alcançando uma acurácia de aproximadamente 95.29% no conjunto de teste. A matriz de confusão para este modelo destacou uma eficiência notável com 120 verdadeiros negativos (VN) e 64 verdadeiros positivos (VP).

Ambos os modelos demonstraram robustez na classificação de áudios. A escolha entre esses modelos pode depender de fatores como complexidade computacional e requisitos específicos do domínio.

Estas descobertas sublinham o potencial das redes neurais convolucionais na análise de sinais de áudio e ressaltam a importância da escolha adequada de características acústicas para a tarefa em questão

TRABALHOS FUTUROS

Para trabalhos futuros, extender o modelo para detectar o choro e também classificar o tipo de choro e tirar inferências. Outra abordagem seria explorar técnicas avançadas, como redes neurais recorrentes, para lidar com padrões temporais mais complexos.

REFERENCES

[1] Miranda, I.; Diacon, A.; Nielser, T. A comparative study of features for acoustic cough detection using deep architectures. In Proceedings of the 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Berlin, Germany, 23–27 July 2019; pp. 2601–2605.

[2] S. M. Bell and M. D. S. Ainsworth, “Infant Crying and Maternal,” vol. 43, no. 4, pp. 1171– 1190, 1972..

[3] A. D. Murray, “Infant crying as an elicitor of parental behavior: An examination of two models,” Psychol. Bull., vol. 86, no. 1, pp. 191–215, 1979.

[4] E. Franti, I. Ispas, and M. Dascalu, “Testing the Universal Baby Language Hypothesis -Automatic Infant Speech Recognition with CNNs,” 2018 41st Int. Conf. Telecommun. Signal Process. TSP 2018, 2018.

[5] Y. Mu, L. A. Hernández Gómez, A. C. Montes, C. A. Martínez, X. Wang, and H. Gao, “Speech Emotion Recognition Using Convolutional-Recurrent Neural Networks with Attention Model,” DEStech Trans. Comput. Sci. Eng., no. cii, pp. 341–350, 2017.

[6] Y. Lecun, Y. Bengio, and G. Hinton, “Deep learning,” Nature, vol. 521, no. 7553, pp. 436– 444, 2015.

[7] Saraswathy, J., Hariharan, M., Yaacob, S., Khairunizam, W.: Automatic classification of infant cry: A review. In: International Conference on Biomedical Engineering (ICoBE). (Feb 2012) 543–548

[8] Lavner, Y., Cohen, R., Ruinskiy, D., Ijzerman, H.: Baby cry detection in domestic environment using deep learning. In: IEEE International Conference on the Science of Electrical Engineering (ICSEE). (Nov 2016) 1–5.

[9] E. Franti, I. Ispas, and M. Dascalu, “Testing the Universal Baby Language Hypothesis - Automatic Infant Speech Recognition with CNNs,” 2018 41st Int. Conf. Telecommun. Signal Process. TSP 2018, pp. 1–4, 2018

[10] Yong, B.F.; Ting, H.; Ng, K. Baby cry recognition using deep neural networks. In World Congress on Medical; Springer: Prague, Czech, 2019; pp. 809–816.

[11] BĂNICĂ, I.-A., CUCU, H., BUZO, A., BURILEANU, D., & BURILEANU, C. (2016). Automatic methods for infant cry classification. 2016 International Conference on Communications (COMM), 51-54. https://doi.org/10.1109/ICComm.2016.752826 1.

[12] DEWI, S. P., PRASASTI, A. L., & IRAWAN, B. (2019). The Study of Baby Crying Analysis Using MFCC and LFCC in Different Classification Methods. 2019 IEEE International Conference on Signals and Systems (ICSigSys), 18-23. https://doi.org/10.1109/ICSIGSYS.2019.88110 70.

[13] KULKARNI, P., UMARANI, S., DIWAN, V., KORDE, V., & REGE, P. P. (2021). Child Cry Classification—An Analysis of Features and Models. 2021 6th International Conference for Convergence in Technology (I2CT), 1-7. https://doi.org/10.1109/I2CT51068.2021.9418129.

[14] https://freesound.org/