## Treinamento do Modelo Calm-Wave

**Objetivo:** Treinar um modelo de aprendizado de máquina para classificar áudios em duas categorias: "ruído" e "voz", utilizando um conjunto de dados de áudios diversos.

**Etapas do Treinamento:**

1. **Coleta e Preparação dos Dados:**
   * Um conjunto de dados de áudios foi coletado, contendo amostras de "ruído" e "voz".
   * Os áudios foram divididos em pastas separadas, "ruido" e "voz", para facilitar a organização e o carregamento.
   * Cada arquivo de áudio foi pré-processado para garantir uniformidade:
     + **Carregamento:** Os áudios foram carregados usando a biblioteca Librosa, com uma taxa de amostragem de 16000 Hz e duração máxima de 2 segundos.
     + **Padding:** Áudios com menos de 2 segundos foram preenchidos com zeros para atingir a duração padrão.
     + **Normalização:** A amplitude dos áudios foi normalizada para o intervalo [-1, 1].
2. **Aumento de Dados:**
   * Técnicas de aumento de dados foram aplicadas para aumentar a diversidade do conjunto de dados e melhorar a robustez do modelo:
     + **Time Masking:** Zera partes aleatórias do sinal de áudio.
     + **Frequency Masking:** Remove informações em bandas de frequência específicas do espectrograma.
     + **Pitch Shifting:** Altera o tom do áudio, tornando-o mais agudo ou mais grave.
     + **Time Stretching:** Altera a velocidade do áudio, alongando ou comprimindo o sinal no tempo.
     + **Adição de Ruído:** Adiciona ruído gaussiano ao áudio.
3. **Extração de Características:**
   * Mel-espectrogramas foram extraídos de cada áudio como características para o modelo:
     + Os áudios foram convertidos em mel-espectrogramas usando a biblioteca Librosa, com 128 bandas de mel, tamanho de FFT de 1024 e comprimento de hop de 128.
     + Os mel-espectrogramas foram convertidos para escala de decibéis (dB) e normalizados.
4. **Construção do Modelo:**
   * Uma rede neural convolucional (CNN) foi construída usando a biblioteca TensorFlow/Keras:
     + O modelo consiste em três camadas convolucionais com ativação ReLU, cada uma seguida por batch normalization e max pooling.
     + Após as camadas convolucionais, o modelo possui uma camada de flatten, seguida por duas camadas densas com ativação ReLU e dropout.
     + A camada de saída possui duas unidades com ativação softmax, para classificar o áudio como "ruído" ou "voz".
5. **Treinamento do Modelo:**
   * O modelo foi treinado usando o conjunto de dados preparado, com as seguintes configurações:
     + Otimizador Adam com taxa de aprendizado de 0.001.
     + Função de perda sparse categorical crossentropy.
     + Métricas de avaliação: acurácia.
     + Tamanho do batch: 32.
     + Número de épocas: 5.
     + Divisão de validação: 20% dos dados de treinamento foram usados para validação.
     + Pesos de classe balanceados foram usados para lidar com classes desbalanceadas.
6. **Avaliação do Modelo:**
   * O modelo treinado foi avaliado usando um conjunto de dados de teste separado:
     + A acurácia do modelo foi calculada para medir o desempenho geral.
     + Um relatório de classificação foi gerado para fornecer métricas detalhadas, como precisão, recall e F1-score para cada classe.
7. **Salvamento do Modelo e Resultados:**
   * O modelo treinado foi salvo em formato Keras para uso posterior.
   * As configurações do modelo e os resultados do treinamento foram salvos em arquivos JSON para documentação e análise.

**Ferramentas e Bibliotecas Utilizadas:**

* **Librosa:** Para análise e processamento de áudio.
* **NumPy:** Para computação numérica.
* **TensorFlow/Keras:** Para construir e treinar o modelo de aprendizado de máquina.
* **Scikit-learn:** Para métricas de avaliação e divisão de dados.

**Considerações:**

* O uso de técnicas de aumento de dados melhorou significativamente a robustez e a generalização do modelo.
* O processamento paralelo foi utilizado para acelerar o treinamento e o processamento de dados.
* A arquitetura do modelo CNN se mostrou eficaz na classificação de áudio, capturando padrões relevantes nos mel-espectrogramas.
* O modelo treinado pode ser usado para classificar novos áudios como "ruído" ou "voz" com alta precisão.