Modelagem de Rede Neural Artificial para Reconhecimento de Gênero pela Fala

Luiz Felipe M. Votto
Departmento de Engenharia Elétrica e de Computação (SEL)
Escola de Engenharia de São Carlos, Universidade de São Paulo (EESC - USP)
São Carlos, Brazil
luizvotto@gmail.com

***Resumo*—Este trabalho visa apresentar um modelo de Rede Neural Perceptron Multicamada (MLP - *Multi-Layer Perceptron*) para, a partir de uma base de dados de fala, efetuar o reconhecimento do gênero de um enunciador num arquivo de áudio. Utilizando e adaptando os métodos descritos na literatura realizados com processamento de áudio, será discutida uma alternativa para reconhecimento de gêneros implementando um MLP do zero e analisando seu desempenho.**

I. Introdução

A partir do momento em que as grandes plataformas, tais as de mídia social ou até mesmo de compras, customizam seu conteúdo a partir do perfil de cada usuário, conseguir maneiras rápidas de extrair informações destes indivíduos toma um valor considerável na atualidade. Neste caso, conseguir informação em tempo real sobre atributos como o gênero de um indivíduo pode ser valioso. Tomando em vista, o aspecto da necessidade de rapidez na coleta destes dados, a implementação de uma Rede Neural Artificial – RNA – para realizar tais tarefas é adequada: uma vez que, depois de treinadas, retornam resultados com celeridade [1].

Utilizando o *corpus* TIMIT de fala na língua inglesa em microfone [2], o qual possui abundância de amostras de áudio com diversos falantes e dialetos; propõe-se treinar uma RNA na arquitetura de Percetron Multicamada (MLP - *Multi-Layer Perceptron*) para identificar se o gênero de um falante em uma amostra de áudio em inglês é masculino ou feminino.

Na seção II, será discorrido em mais detalhes o que deve ser feito para se atingir o objetivo do trabalho, levantando os principais cuidados a serem tomados e os obstáculos a serem superados. Adiante, na seção III, apresentam-se os detalhes de como contornar os problemas levantados na seção II, desde o pré-processamento do sinal digital até o treinamento e avaliação das RNAs projetadas. A seção IV e V tratam-se da exposição e análise dos resultados obtidos com o treinamento e teste das redes neurais modeladas bem como o processo de validação cruzada entre as redes.

II. Definição do Problema

Tendo uma base de dados extensa como o *corpus* TIMIT, é importante organizar a informação com cuidado. Primeiramente, checar os formatos dos arquivos de áudio para tratá-los e, depois, separar a base de dados de maneira a executar um treinamento adequado para a rede. Para o processamento do sinal digital de áudio, vários cuidados devem ser tomados e serão discutidos em detalhe mais adiante.

O *corpus* TIMIT possui um total de 6300 frases – 10 sentenças enunciadas por 630 indivíduos de 8 regiões de dialetos majoritários dos Estados Unidos da América. Categoriza-se um falante como pertencente a uma região de dialeto de acordo com a área geográfica em que ele viveu durante sua infância [2]. Cada arquivo de áudio é categorizado de acordo com a **região de dialeto**, **gênero do enunciador** e a **sentença falada**.

É importante para o projeto a escolha de uma arquitetura de RNA adequada para a aplicação que temos em mente. Nota-se ainda que os arquivos de áudio possuem durações variáveis e este fator deve ser tomado em consideração.

III. Solução do Problema

A partir dos problemas levantados anteriormente, divide-se nas seguintes etapas a solução do problema: **pré-processamento dos sinais**, **escolha de arquitetura**, **divisão dos conjuntos de treinamento e de teste** e **pós-processamento da resposta da rede**.

As RNAs modeladas neste projeto foram implementadas na linguagem Python com pacotes comuns de tratamento de álgebra linear, de entrada e saída, e de imagem. O código-fonte é aberto, exceto pela base TIMIT, e está disponível na plataforma GitHub [3].

A. Pré-Processamento

Em processamento de sinais de áudio, alguns métodos são comumente utilizados para se extrair características-chaves destes sinais [4] – [6]. Para reconhecimento de fala, um método utilizado é extrair os **Coeficientes Cepstrais de Frequência-Mel** (**MFCC** – *Mel-Frequency Cepstrum Coefficients*) de quadros de alguns milissegundos de cada sinal como principal elemento de coleta de características relevantes [6].

Segue-se aqui uma descrição da sequência de métodos utilizados para preparar as entradas. A figura 1 ilustra os passos mais importantes para a obtenção da entrada das RNAs modeladas.

Em primeiro lugar, foi necessário executar a conversão dos arquivos no formato NIST SPHERE para WAV. A implementação é feita no arquivo `sph2wav.py` no repositório do trabalho [3].

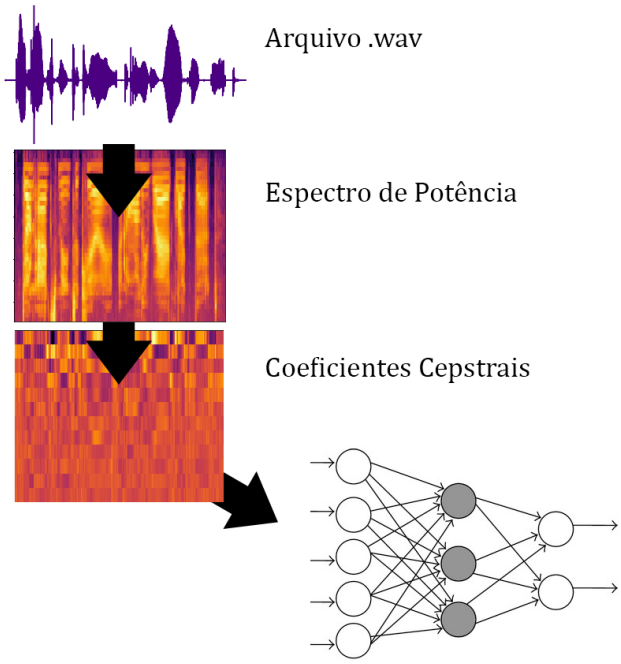

Figura 1. Aspectos do pré-processamento de uma amostra de áudio feitos pelo programa em Python.

Feita a conversão de formatos, é importante aplicar um **filtro de pré-ênfase** no sinal a fim de equilibrar o espectro de frequência já que as frequências mais baixas tendem a ter magnitudes maiores que as altas; também para evitar problemas numéricos na posterior aplicação da transformada de Fourier e para melhorar a relação sinal-ruído [4], [5]. Optou-se por utilizar o seguinte filtro no domínio do tempo:

$$y[n] = x[n] - \alpha x[n-1],$$

com a escolha de $\alpha = 0,97$.

Seguindo o processo, costuma-se dividir o sinal em vários **quadros** para aplicar a transformada de Fourier já que, em quadros da ordem de milissegundos, pode-se assumir que as frequências são estacionárias [6]. Neste trabalho, entretanto, para preservar simplicidade e tempo, já que utilizar janelas pequenas criaria entradas muito grandes para a RNA, utilizamos um número menor de quadros – limitou-se a 10 quadros por arquivo de áudio.

Posteriormente, para cada quadro, aplica-se uma **janela de Hamming**, dada pela expressão [4], [5]:

$$w[n] = 0,54 - 0,46\cos\left(\frac{2\pi n}{N-1}\right),$$

onde N é o número de amostras no quadro em que se aplica a janela.

Assim, para cada quadro já tratado, aplica-se a transformada de Fourier a partir do método FFT (*Fast Fourier Transform*) e então, obtem-se o **espectro de potência** para cada janela:

$$P = \frac{|\texttt{FFT}(\mathbf{x}_i)|^2}{N},$$

onde $\texttt{FFT}(\mathbf{x}_i)$ é a transformada de Fourier aplicada no i-ésimo quadro – $\mathbf{x}_i$.

Para cada espectro de potência, aplicamos **uma sequência de filtros passa-banda de formato triangular** espaçados segundo a escala logarítmica de Mel, que simula a percepção do ouvido humano, já que sons de alta frequência são discriminados com menos exatidão que os sons de baixa frequência [6]. A escala de Mel se relaciona com a escala convencional em Hertz da seguinte maneira:

$$F_{\text{Mel}} = 2595\log_{10}\left(1 + \frac{F_{\text{Hz}}}{700}\right).$$

Assim, o espectro de potência filtrado se assemelha ao segundo quadro ilustrado na Figura 1. O banco de filtros triangulares na escala de Mel é ilustrado da Figura 2, retirada de [5].

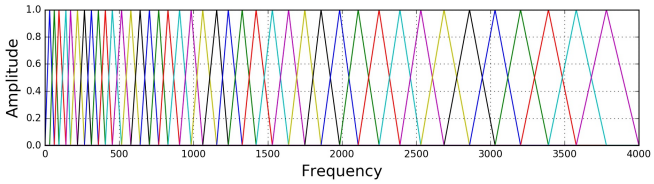

Figura 2. Banco de filtros na escala de Mel [5].

A extração dos MFCCs é feita a partir da aplicação da **Transformada Discreta do Cosseno** no espectro de potência, caracterizando agora o cepstro. Aqui, usaremos somente os 12 primeiros coeficientes capturados pela transformada, tornando-se agora algo que se assemelha ao terceiro quadro da Figura 1.

Após este processo, normalizamos os coeficientes, se armazenados na matriz S:

$$\hat{s_{ij}} = \frac{s_{ij}}{\max|s_{ij}|}.$$

B. Arquitetura e Topologia da Rede

Como dito anteriormente, para fins de simplicidade, cada amostra de áudio foi dividida em um número fixo de quadros – 10 – dos quais se retiraram os 12 MFCCs. Assim sendo, a rede recebe como entrada, 120 valores. Vale salientar que existem arquiteturas de rede neural potencialmente mais adequadas para este propósito, pois elas podem lidar com entradas de tamanho variável, como as Redes Neurais Recorrentes. Entretanto, seu uso levantaria a complexidade do projeto acima do que é razoável. Assim, trabalhou-se com uma rede **MLP convencional** com funções de ativação logísticas:

$$F(x) = \frac{1}{1+e^{-x}},$$

com fator de aprendizado $\eta = 0,1$; fator de inércia $\gamma = 0,8$ e fator de convergência $\epsilon = 10^{-6}$.

Para o processo de validação cruzada, foram treinadas majoritariamente redes com duas camadas escondidas dado que **é desconhecido se a fronteira de separação entre a classificação de gêneros é não-convexa no espaço de**

busca gerado. Vale lembrar que MLPs classificadores com apenas uma camada escondida apenas separam as classes com fronteira de separação convexa.

C. Treinamento e Teste

O *corpus* TIMIT já sugere, em sua documentação, uma separação de conjunto de treinamento e de teste, a qual separa 80% dos arquivos de áudio para treinamento e o restante para teste com o mínimo de intersecção de enunciadores entre os conjuntos. Adotou-se a sugestão dos criadores da base e extraiu-se a informação sobre o gênero de cada enunciador pela nomeação dos arquivos já feita previamente.

D. Pós-Processamento

Optou-se por adotar a codificação ***one of C-class***, ou seja, um neurônio da camada de saída para cada classe. Ou seja, cada neurônio ativado em 1 na camada de saída aponta se a faixa de áudio é enunciada por uma pessoa do gênero correspondente. Se o primeiro neurônio é ativado, diz-se que o falante é mulher e, caso o segundo esteja ativado, diz-se que é homem.

IV. Resultados de Implementação

Devido ao tempo extenso que leva o treinamento das redes, por enquanto temos duas redes. Cada uma com 120 entradas. Serão disponibilizadas mais topologias em arquivos `.pickle` no repositório do projeto [3]. Uma topologia – **topologia A** – com apenas **uma camada escondida de 20 neurônios**. Outra – a **topologia B** – com **duas camadas escondidas: a primeira com 10 neurônios e a segunda com 8**.

A Tabela I ilustra as estatísticas de treinamento de cada uma das redes. O tempo de treinamento em segundos, o número de épocas efetuadas e o erro quadrático médio final.

Tabela I
Resultados de treinamento das redes.

Topologia	Tempo de Treinamento [s]	Épocas	E.Q.M. final
A	15566	386	0,019
B	16334	816	0,048

A figura 3 mostra a evolução do erro quadrático médio no decorrer do treinamento de cada rede – topologia A à esquerda e, à direita, a topologia B. Percebe-se que o treinamento da topologia B segue um caminho mais complexo devido à não-linearidade que se impõe com mais de uma camada escondida.

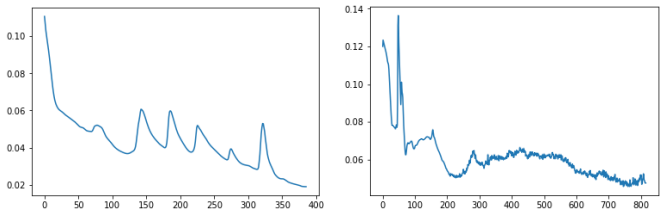

Figura 3. Evolução do erro quadrático médio no treinamento das redes em função do número de épocas.

A tabela II mostra as estatísticas finais dos testes realizados, ilustrando quantas pessoas cada topologia classificou para cada gênero.

Tabela II
Número de pessoas de cada gênero segundo a base de teste e as redes com suas respectivas taxas de acerto.

	TIMIT	Topologia A	Topologia B
Feminino	560	387	403
Masculino	1120	1293	1277
Acerto		78%	80%

V. Análise dos Resultados

Pela divisão dos conjuntos de dados sobre os quais foram realizados o treinamento e o teste das RNAs implementadas, pode-se dizer que as redes cumprem seu propósito apesar de haver uma chance de erros. Em se tratar de um processo que pode ser feito em tempo real e que as amostras de dados são falas de poucos segundos, ao se juntar várias falas de um enunciador, por exemplo, numa chamada telefônica, a chance de acerto tende a aumentar.

Vemos que existem diferenças consideráveis nos treinamentos de redes com topologias que diferem em número de camadas escondidas devido à não-linearidade imposta sempre que se adiciona uma camada escondida. Além disso, para uma MLP *feed-forward* convencional e, considerando que os dados extraídos são poucos para manter uma entrada de tamanho fixo – de 120 MFCCs –, as redes possuem performance acima do esperado, já que este tipo de trabalho costuma ser efetuado segmentando os áudios em janelas da ordem de milissegundos.

VI. Conclusões

É importante salientar que este é um exercício didático na modelagem de redes neurais. Portanto, apesar da classificação de gênero de indivíduos pela sua fala é um conceito sociologicamente ultrapassado [8], ela cumpre o objetivo de ilustrar como se pode extrair informações por meio de processamento de sinais digitais de áudio de forma bem-sucedida utilizando apenas MLPs convencionais.

Indo contra o *mainstream*, que é a utilização de *Deep Learning* ou redes neurais recorrentes, este trabalho mostra que é possível, até certo ponto, obter sucesso na classificação de características extraídas de arquivos de fala em microfone.

Referências

[1] I.N. da Silva, et al. *Artificial Neural Networks: a Practical Course*. São Paulo, SP: São Carlos, 2017, pp. 123-35.

[2] J.S. Garofolo, et al. *TIMIT Acoustic-Phonetic Continuous Speech Corpus LDC93S1*. Web Download. Philadelphia: Linguistic Data Consortium, 1993.

[3] L.F. Votto. *Neural Networks Assignments*. https://github.com/LVotto/neural_networks_assignments

[4] E. Loweimi, et al. *On the Importance of Pre-emphasis and Window Shape in Phase-based Speech Recognition*. In: Drugman T., Dutoit T. (eds) Advances in Nonlinear Speech Processing. NOLISP 2013. Lecture Notes in Computer Science, vol 7911. Springer, Berlin, Heidelberg

[5] H. Fayek. *Speech Processing for Machine Learning: Filter banks, Mel-Frequency Cepstral Coefficients (MFCCs) and What's In-Between*. https://haythamfayek.com/2016/04/21/speech-processing-for-machine-learning.html. 2016.

[6] E. Cakir. *Multilabel Sound Event Classification with Neural Networks*. MS thesis. Tampere University of Technology, 2014.
[7] I. Sutskever, O. Vinyals, Q.V. Le. *Sequence to Sequence Learning with Neural Networks*. Advances in Neural Information Processing Systems 27 (NIPS 2014).
[8] S. Stryker. *(De)Subjugated Knowledges: An Introduction to Transgender Studies*. 2006.