Reconhecimento Automático de Fala em Português usando Redes Neurais Artificiais Profundas

Diana A. Bonilla C*, Nadia Nedjah†
Departamento de Engenharia Eletrônica e Telecomunicações
Faculdade de Engenharia
Universidade do Estado do Rio de Janeiro
*Email: dabonillac@gmail.com
†Email: nadia@eng.uerj.br

Luiza de Macedo Mourelle
Departamento de Engenharia de Sistemas e Computação
Faculdade de Engenharia
Universidade do Estado do Rio de Janeiro
Email: ldmm@eng.uerj.br

***Resumo*—O presente trabalho propõe uma modificação na técnica de treinamento de redes neurais profundas aplicadas em reconhecimento de imagens, para classificar um grupo de palavras representadas por sinais de fala. O processo extrai as características de frequência dos sinais de fala e realiza um pré-treinamento não supervisionado baseados nos dados para inferir os pesos iniciais da rede reconhecedora e culmina com o treinamento da rede neural profunda. O sistema de reconhecimento automático de fala proposto melhora a acurácia na identificação de palavras comparada àquela obtida por uma rede perceptron a multicamadas.**

***Palavras chave*—Reconhecimento Automático de Fala, Redes Neurais Artificiais, Redes Neurais Profundas.**

I. Introdução

O objetivo do reconhecimento automático de fala RAF,ou ASR (*Automatic Speech Recognition*) é a transcrição da fala humana para dados digitais. Isto é uma tarefa desafiante, já que os sinais obtidos no processo de fala são altamente variáveis, devido à grande quantidade de atributos da voz humana, além das características próprias no contexto de fala como os ruídos de fundo e as propriedades de cada idioma.

Os *Hidden Markov Models* (HMMs) em combinação com modelos básicos de redes neurais artificiais têm sido bem sucedidos no reconhecimento automático de fala [1]. No entanto, estes modelos são limitados pela complexidade idiomática, já que requerem, neste caso, muitos estados ocultos no HMM usado.

Nos últimos anos, o conceito de arquiteturas profundas tem chamado a atenção em estudos da inteligência e aprendizagem artificiais. Esta arquitetura profunda pode ser considerada como a terceira geração de redes neurais, as quais são caracterizadas por estar configuradas utilizando várias camadas ocultas, permitindo uma abstração maior da informação, e portanto proporcionando uma melhor aprendizagem.

Este trabalho propõe um projeto com uma arquitetura de rede neural artificial profunda (RNAP, no inglês *Deep Neural Network* (DNN)) para a classificação de palavras em Português. Para avaliar o desempenho da rede proposta, os resultados de comparação de acurácia deste projeto e de uma rede Perceptron multicamada (*Multilayer Perceptron* (MLP)) clássica, introduzida em [2], são apresentados e analisados.

O resto deste artigo é organizado em sete seções. Inicialmente, na Seção II, é apresentada uma visão global das arquiteturas de redes neurais artificiais exploradas neste trabalho, desde a rede MLP até as máquinas restritas de Boltzmann. Em seguida, na Seção III, é apresentado o algoritmo usado para treinar uma rede neural artificial profunda. Posteriormente, na Seção IV, são detalhados os passos de implementação da rede proposta, visando a classificação de palavras em Português, cujos sinais de fala foram pré-processados para extrair as características principais. Depois, na Seção V, são discutidos os resultados obtidos durante os experimentos realizados. Finalmente, na Seção VI, são apresentadas as conclusões referentes à topologia proposta e são apontadas algumas direções relevantes para os trabalhos futuros.

II. Arquiteturas de redes neurais artificiais

Nesta seção, serão apresentados os conceitos básicos relacionados às diferentes arquiteturas de redes neurais usadas neste trabalho para o treinamento e implementação de um reconhecedor automático de fala em Português.

A. Perceptron de múltiplas camadas

A arquitetura de uma rede Perceptron de múltiplas camadas consiste de uma sequência de camadas consecutivas, cada uma composta por um grupo de neurônios interconectados com outros neurônios de outras camadas, criando uma estrutura unidirecional (*feed-forward*), onde a saída da camada de entrada e das camadas intermediárias são enviadas e usadas como entradas nas camadas dos níveis seguintes.

A Figura 1 apresenta esta estrutura. Matematicamente, a estrutura de uma rede MLP em cada camada está descrita por:

$$o_i^{(l)} = f\left(\sum_{j=1}^{J} o_j^{(l-1)} w_{j,i}^{(l)} + w_{0,i}^{(l)}\right), \quad 1 \leq l \leq L \qquad (1)$$

onde f é a função de ativação da camada, geralmente configurada como uma função não linear tangente hiperbólica para as camadas intermediárias, também chamadas de escondidas, e uma função linear para apresentar os resultados da camada de saída. O índice l identifica a camada atual para uma rede de L camadas, $o_i^{(l)}$ representa a saída do neurônio i na camada atual

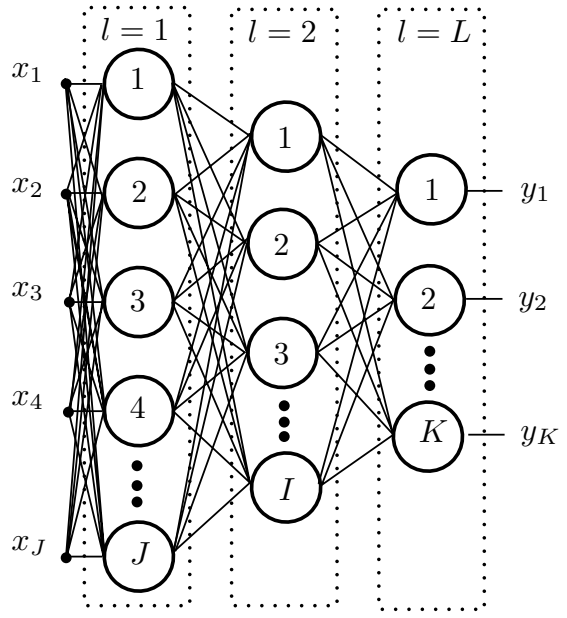

Figura 1. Arquitetura básica de uma rede perceptron multicamada

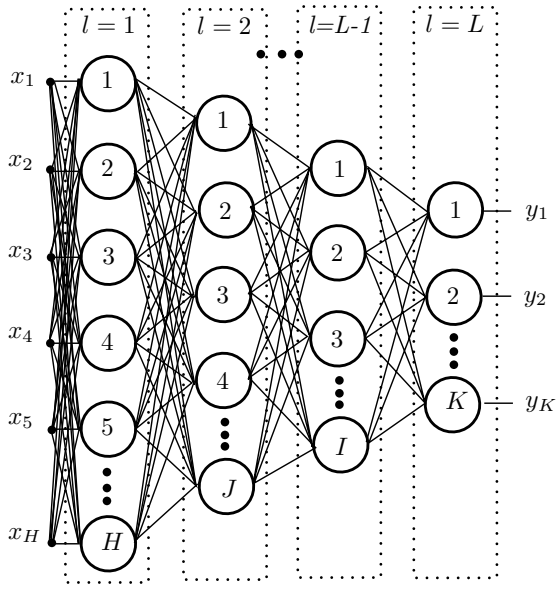

Figura 2. Arquitetura básica de uma rede artificial profunda

l, $w_{j,i}^{(l)}$ são os pesos associados às conexões dos neurônios da camada $l-1$ com os neurônios da camada atual e $w_{0,i}^{(l)}$ são os bias de cada neurônio da camada atual. Note que o vetor de saídas da camada $l = 0$ de longitude J é igual ao vetor de características de entrada, ou seja, $\mathbf{o}^{(0)} = \mathbf{x}$.

As redes MLP de uma camada escondida são consideradas como aproximadores universais de funções contínuas [3], e têm sido aplicadas com sucesso na resolução de problemas complexos usando a retropropagação de erro (*back-propagation*) como algoritmo de treinamento [4]. Porém, a modelagem de funções de alta variança requerem um número maior de neurônios comparado com aquele usado em uma arquitetura de mais de uma camada escondida [5].

B. Redes neurais artificiais profundas

Devido à limitação supracitada das redes MLP com mais de uma camada escondida, surge a necessidade de implementar *redes neurais artificiais profundas* que tenham mais de uma camada de componentes adaptativos não lineares, como é representado na Figura 2, permitindo a caracterização de uma ampla família de funções de forma mais compacta que as arquiteturas pouco profundas usadas normalmente.

A função de ativação da camada de saída depende do objetivo final da rede. Para uma classificação a multi-classe é geralmente usada a função *softmax*, definida como:

$$y_k = \frac{\exp\left(w_{0,k}^{(L)} + \sum_{i=1}^{I} o_i^{(L-1)} w_{i,k}^{(L)}\right)}{\sum_{k=1}^{K} \exp\left(w_{0,k}^{(L)} + \sum_{i=1}^{I} o_i^{(L-1)} w_{i,k}^{(L)}\right)} \tag{2}$$

onde $1 \leq k \leq K$. O resultado y_k é a probabilidade condicional da classe k dentro da camada de saída L.

Os parâmetros da camada escondida estão organizados na forma do vetor $\lambda = [W^{(1)}, W^{(2)}, ..., W^{(L+1)}]$, onde cada elemento $W^{(i)}$ abrange os pesos correspondentes à camada indicada assim como os fatores de polarização ou bias de cada neurônio desta camada. Os valores dos pesos podem ser otimizados usando um treinamento supervisionado através da minimização de uma função de custo. Esta função indica a diferença entre as etiquetas proporcionadas a cada sinal de fala usado durante o treinamento e as saídas obtidas pela rede usando um algoritmo de retropropagação. Para problemas de classificação, a função de custo usada considerando o valor correto d_k e o inferido y_k é a entropia cruzada:

$$Q(y_k, d_k) = -\sum_{k=1}^{K} d_k \log(y_k) \tag{3}$$

onde $Q(y_k, d_k)$ indica a divergência existente entre duas distribuições de probabilidade, neste caso entre d_k e y_k. Quando ambas distribuições coincidem, a entropia cruzada torna-se nula. Esta atingirá no valor máximo quando a diferença entre as distribuições é a maior possível. O processo de treinamento é iterativo e usando o algoritmo de retropropagação de erro, atualiza os parâmetros dentro do vetor λ de acordo com a saída indicada pela entropia cruzada até que a diferença entre a distribuição de probabilidades d_k e y_k seja a menor possível.

Devido ao aumento da complexidade do modelo usado, o algoritmo de retropropagação de erro tende a convergir para mínimos locais, já que a magnitude do erro diminui rapidamente quando a profundidade da rede é grande [6]. Além disso, a complexidade da rede permite aprender relações complexas entre as entradas e saídas. Se o banco de dados de treinamento não é considerável a rede pode aprender relações derivadas do ruido de fundo, por exemplo, gerando uma estrutura de classificação pouco generalizável para os dados de teste. Assim, a estrutura apresenta problemas de sobre-

treinamento (*overfitting*) [4]. A solução é implementar uma rede não supervisionada que faça um pré-treinamento para inferir os pesos iniciais.

C. Máquina restrita de Boltzmann

Um método popular usado para o pré-treinamento é uma máquina restrita de Boltzmann ou RBM (*Restricted Boltzmann Machine*) [7]. É um aprendizado não supervisionado, onde é modelada a dependência entre um grupo de variáveis aleatórias usando uma arquitetura de duas camadas. Em RBM, as unidades estocásticas visíveis $\mathbf{v} = [v_1, v_2, ..., v_V]^T$ são conectadas às unidades estocásticas escondidas $\mathbf{h} = [h_1, h_2, ..., h_H]^T$ onde V e H representam as quantidades de unidades em cada camada. A Figura 3 apresenta a estrutura da rede.

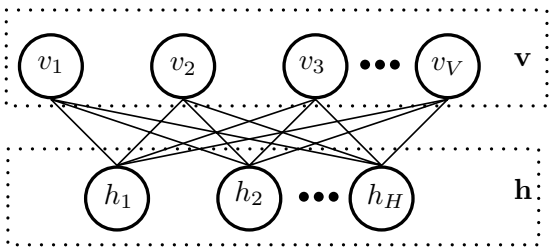

Figura 3. Arquitetura para uma máquina restrita de Boltzmann

Para redes do tipo RBM Gaussianas, onde $v_i \in \mathbb{R}^V$ e $h_j \in \{0,1\}^H$, ou seja h_i binários, a energia é definida por:

$$E(\mathbf{v}, \mathbf{h}; \lambda) = \sum_{i=1}^{V} \frac{(v_i - a_i)^2}{2\sigma_i^2} - \sum_{j=1}^{H} b_j h_j - \sum_{i=1}^{V}\sum_{j=1}^{H} w_{i,j} h_j \frac{v_i}{\sigma_i^2} \quad (4)$$

onde λ é o conjunto de parâmetros que consiste de $\mathbf{a} = [a_1, a_2, ..., a_V]$ e $\mathbf{b} = [b_1, b_2, ..., b_H]$ como termos de bias para a camada visível e a camada escondida respetivamente. O termo $w_{i,j} \in \mathbb{R}^{V \times H}$ são os pesos das conexões entre os elementos das camadas. Note que σ^2 representa a variância entre os elementos do vetor $\mathbf{v}$.

A probabilidade condicional para cada neurônio visível v_i e escondido h_j que dependem dos outros neurônios é dada por:

$$p(h_j = 1|\mathbf{v}) = f\left(b_j + \sum_{i=1}^{V} w_{i,j}\frac{v_i}{\sigma_i^2}\right) \quad (5)$$

$$p(v_i = v|\mathbf{h}) = \mathcal{N}\left(\mathbf{v} \mid \left(a_i + \sum_{j=1}^{H} h_j w_{i,j}\right), \sigma_i^2\right) \quad (6)$$

onde f denota a função sigmoide: $f(x) = \frac{1}{1+\exp -x}$ e $\mathcal{N}(\mathbf{v} \mid \mu, \sigma^2)$ a função de distribuição Gaussiana de v, com média μ e variância σ^2.

Durante o processo de treinamento e atualização dos parâmetros, é usado o critério de máxima verossimilhança (*maximum-likelihood estimation*- MLE). Este método estima um conjunto λ que maximiza a função de custo

$$\mathcal{L}(\lambda) = \sum_{j=1}^{H} \log -E(\mathbf{v}, \mathbf{h} \mid \lambda)$$

onde $E(\cdot)$ representa a energia conforme definida na Equação 4.

Para diminuir o custo de processamento é geralmente usado o algoritmo de divergência contrastiva (*constrastive divergence*-CD), que permite após um processo iterativo obter os valores iniciais dos pesos que serão usados na rede profunda. Este algoritmo inicia o processo com o vetor de dados de treinamento da rede profunda $\mathbf{v}$ na camada de unidades estocásticas visíveis. Depois, as unidades escondidas $\mathbf{h}$ são calculadas usando a Equação 5. Com os resultados obtidos deste calculo são geradas as unidades da camada $\mathbf{v}^1$, usando a próxima probabilidade $p(\mathbf{v}^{(t+1)} \mid \mathbf{h}^t)$. Este processo é realizado iterativamente até alcançar à instante $t = T$. A Figura 4 ilustra este processo.

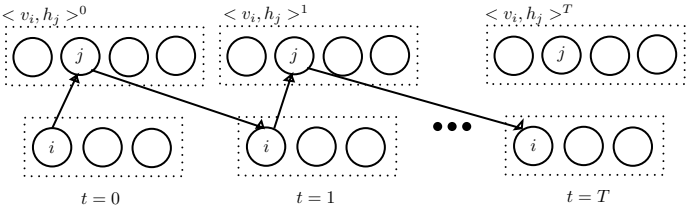

Figura 4. Amostragem feita sobre uma máquina restrita de Boltzmann

Depois de atingir a instancia $t = T$, o conjunto de parâmetros λ é modificado conforme as seguintes regras de atualização:

$$\nabla w_{i,j} = \eta\left(M\left[\frac{1}{\sigma^2}v_i h_j\right]^{(0)} - M\left[\frac{1}{\sigma^2}v_i h_j\right]^{(T)}\right) \quad (7)$$

$$\nabla a_i = \eta\left(M\left[\frac{1}{\sigma^2}v_i\right]^{(0)} - M\left[\frac{1}{\sigma^2}v_i\right]^{(T)}\right) \quad (8)$$

$$\nabla b_j = \eta\left(M\left[h_j\right]^{(0)} - M\left[h_j\right]^{(T)}\right) \quad (9)$$

onde η representa a taxa de atualização e $M[\cdot]$ o valor esperado. O processo de atualização é iterado até que o os dados convergem ao estado inicial [7].

III. Reconhecimento de palavras usando redes neurais artificiais profundas

Inicialmente, os sinais são pré-processados, a fim de obter a representação dos atributos de entrada da rede na forma de um conjunto de frequências espectrais do sinal. Os atributos são conhecidos como MFSC (*Mel-Frequency Spectral Coefficients*). Os detalhes dessa representação serão comentados posteriormente na seção IV-B. A Figura 5 mostra uma representação de um sinal que contém a pronuncia do digito 'dois' como atributos MFSC.

Em seguida são inicializados os pesos da rede. O Algoritmo 1 apresenta os passos do processo de pré-treinamento visando obter os pesos iniciais correspondente para cada uma das l camadas ocultas da rede neural artificial profunda.

Os pesos e bias da camada de saída $l = L$ são inicializados aleatoriamente. Finalmente, a rede neural artificial profunda é treinada usando o algoritmo de gradiente descendente estocástico com retropropagação.

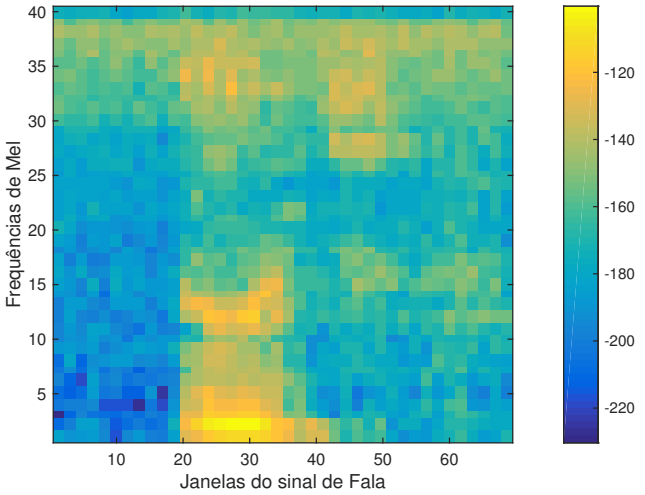

Figura 5. Representação em frequências de Mel do sinal de fala do dígito 'dois'

Algoritmo 1: PRÉ-TREINAMENTO DE RNAP

Entrada: Número de camadas intermediarias da rede neural $L-1$ e vetor de atributos do sinal de fala (**x**)

Saída: Conjunto de vetores iniciais dos parâmetros $\lambda^{(l)}$, $0 \leq l \leq L-1$

inicio
  para $l = 0$ *até* $L-1$ **faça**
    Construir o grupo de treinamento não supervisionado **v**;
    se $l = 0$ **então**
      $\mathbf{v} = \mathbf{x}$;
    senão
      $\mathbf{w} = \lambda^{(l-1)}$;
      $\mathbf{v} = [o^{(l-1)}]$ usando Equação 1 **w**;
    fim
    Garantir que o número de neurônios da camada escondida **h** da rede RBM seja igual ao número de neurônios da camada escondida $l+1$ da RNAP;
    Treinar a rede RBM usando **v**;
    Obter os parâmetros $\lambda^{(l)}$;
  fim
fim

IV. ASPECTOS DE IMPLEMENTAÇÃO

Nesta seção serão abordados os aspectos principais que permitiram o projeto, a implementação e a validação do reconhecedor automático de fala proposto neste trabalho.

A. Corpora

O banco de dados usado é composto por 403 gravações, onde cada gravação contém um dígito de 'zero' até 'nove' falado em Português por quatro indivíduos distintos, sendo duas mulheres e dois homens. Note que cada sinal de fala gravado inclui também ruido de fundo no começo e fim do sinal. Tabela I apresenta a distribuição das gravações correspondentes aos quatro indivíduos, identificados por I_1, I_2, I_3 e I_4, assim como ao dígito falado.

Tabela I
DISTRIBUIÇÃO DO BANCO DE DADOS DE FALA

Palavra	I_1	I_2	I_3	I_4	**Total**
Zero	10	10	11	11	42
Um	10	9	11	11	41
Dois	9	10	10	11	40
Três	9	9	11	11	40
Quatro	9	10	10	11	40
Cinco	10	9	11	10	40
Seis	9	10	10	11	40
Sete	9	10	10	11	40
Oito	10	9	10	10	40
Nove	10	9	10	11	40
Total	94	94	103	109	403

Dos dados organizados aleatoriamente, foram dedicados 240 sinais de fala, representando 60% aproximadamente, para realizar o treinamento, enquanto os 160 sinais de fala restantes, representando ceca de 40% dos dados, para validação da rede treinada.

B. Representação dos sinais de fala

Cada sinal de fala é tratado com uma janela de Hamming de 25ms, considerando uma taxa de amostragem de 10ms. Em seguida, os vetores de características foram gerados por meio de uma transformada de Fourier. O resultado obtido foi tratado por 40 filtros de frequências, gerando assim, 40 coeficientes de energia logarítmica, distribuídos em uma escala de Mel. Todos os coeficientes assim obtidos foram normalizados de tal forma que apresentem uma média de $0,0$ e um desvio padrão de $1,0$. Assim, os coeficientes de entrada estarão no mesmo intervalo de medida. A Figura 6 mostra o conjunto de vetores de características para cada uma das palavras a serem classificadas pela rede neural artificial profunda.

C. Topologia da rede neural artificial profunda

Durante o planejamento da topologia da rede profunda a ser usada foram testadas iterativamente diferentes quantidades de neurônios na primeira camada escondida. Para determinar o número de neurônios que proporcionam as melhores condições de aprendizagem é calculada a média e o desvio padrão dos erros de validação destas iterações, ou seja, a quantidade de neurônios na primeira camada escondida que gera em média um menor erro de validação da rede. A Figura 7 apresenta os resultados dos testes, onde pode ser observado que a melhor solução corresponde ao uso de 28 neurônios.

A quantidade de neurônios das demais camadas é reduzida linearmente usando um declive de $2/3$ da quantidade de neurônios da camada anterior. Assim, as transformações sucessivas feitas por cada camada sobre os dados de entrada agruparão as características frequenciais que identificam a cada palavra falada [8]. Portanto, a rede neural profunda

Tabela II
TAXA DE ACURÁCIA E SENSIBILIDADE DAS REDES COMPARADAS.

		1	2	3	4	5	6	7	8	9	0	Total Acertos
Etiquetas		18	19	18	17	17	15	12	15	13	19	163
RNAP	**#Acertos**	18	19	18	17	16	15	11	14	13	18	159
	%Acertos	100%	100%	100%	100%	94.1%	100%	91.7%	93.3%	100%	94.7%	97.5%
MLP	**#Acertos**	18	19	9	16	16	15	8	11	8	15	135
	%Acertos	100%	100%	50%	94.1%	94.1%	100%	66.7%	73.3%	61.5%	78.9%	82.8%

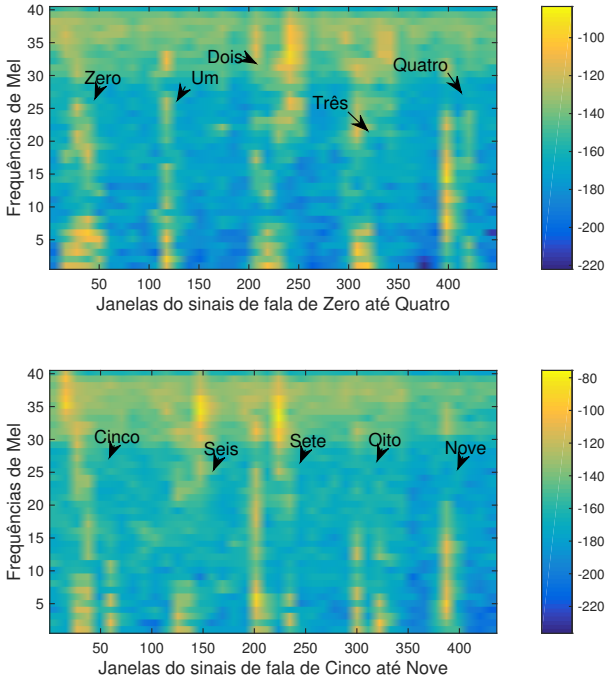

Figura 6. Representação espectral de cada palavra a ser classificada pela rede neural artificial profunda

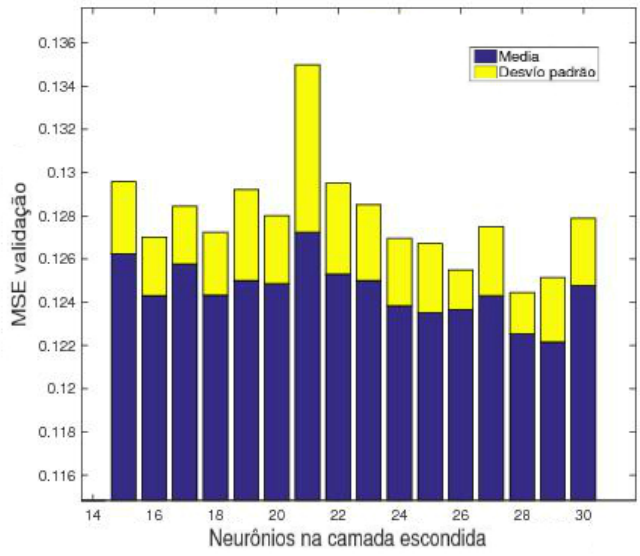

Figura 7. Erro médio de validação para as diferentes quantidades de neurônios na primeira camada escondida

projetada inclui 28 neurônios na primeira camada escondida, 18 na segunda e 12 na última, totalizando 58 neurônios.

V. RESULTADOS

Para avaliar o desempenho da rede neural profunda projetada, as experiências realizadas verificaram a efetividade da implementação da rede. Vale lembrar que os dados são sinais de fala. No caso deste trabalho, a rede é usada para classificar os os sinais de fala de números decimais.

O desempenho da rede neural artificial profunda, projetada na Seção IV, é comparado ao desempenho da rede MLP de três camadas escondidas de 25, 25 e 15 respetivamente, totalizando 65 neurônios, conforme proposta em [2].

A. Comparação das acurácias

Para os 163 sinais de fala usados na fase de teste, foram contados as quantidades de acertos das duas redes comparadas. Os resultados obtidos de acurácia são detalhados na Tabela II.

A rede neural profunda alcançou uma taxa de acurácia de $97,5\%$ emquanto a rede MLP obteve somente $82,8\%$. É válido concluir que a rede MLP teve pior desempenho. Vale lembrar que a rede neural profunda tem menos neurônios que a rede MLP.

A Tabela II também apresenta as taxas de sensibilidade para cada palavra classificada. Vale ressaltar que a rede neural profunda apresentou uma melhor sensibilidade que a rede MLP para o 60% palavras classificadas. Para as demais, as duas redes obtiveram a mesma sensibilidade. Os resultados de sensibilidade das redes demostram que as palavras contendo fonemas com alta vibração, tais como ‘s’, ‘z’ ou ‘tS’ ocasionaram maior dificuldade durante a classificação devido à semelhança desses fonemas com o ruído de fundo.

B. Comparação do número de neurônios

Mais uma experiência foi realizada para avaliar o impacto do número de neurônios no desempenho de uma rede MLP com mais de uma camada escondida. A Tabela III mostra a taxa de acurácia alcançada por uma rede MLP de duas camadas escondidas. Vale notar que não foi mantido o número de camadas escondidas conforme a experiência anterior para mitigar o efeito de sobre-treinamento.

Tabela III
ACURÁCIA DAS REDES MLP DE DUAS CAMADAS ESCONDIDAS DEPENDENDO DO NÚMERO DE NEURÔNIOS NELAS.

Camada 1	**15**	**20**	**25**	**40**	**50**	**90**
Camada 2	**10**	**14**	**17**	**26**	**30**	**60**
Neurônios	**25**	**34**	**42**	**66**	**80**	**150**
Acurácia	50.3%	69.6%	71.6%	86.9%	94.5%	83.5%

Os resultados da Tabela III indicam que a melhor acurácia é obtida pela rede de configuração de (50-30), totalizando 80 neurônios. Assim, a rede neural profunda proposta, de configuração (28-18-12) demostra ser uma melhor opção, não somente pela melhor acurácia, mas também pelo tamanho reduzido em termos de quantidade de neurônios necessários para implementar uma solução de classificação dos sinais de fala.

VI. CONCLUSÕES E TRABALHOS FUTUROS

Este artigo propõe uma solução eficiente para reconhecimento automático de sinais de fala utilizando uma rede neural artificial profunda.

Tendo em vista os resultados obtidos, é notada a melhor acurácia das redes neurais artificiais profundas comparadas com as redes MLP. Contudo, é necessário realizar um pré-treinamento das camadas escondidas para evitar um sobre-treinamento da rede, limitando a generalização da rede para novas entradas.

Por outro lado, embora as redes MLP sejam representadores universais de qualquer função, estas precisam de grandes quantidades de neurônios para conseguir uma acurácia compatível com aquela obtida pela rede neural artificial profunda.

Para este trabalho foi usado um banco de dados próprios, devido à dificuldade de encontrar um conjunto de dados que tivesse as características necessárias e variabilidade das vozes gravadas para realizar um treinamento supervisionado correto.

Para próximos trabalhos, foi adquirida a base de dados *CSLU: Spoltech Brazilian Portuguese Version 1.0* que contém uma grande quantidade de gravações e palavras. Ademais, são usadas mais pessoas nas gravações. Além disso, contêm etiquetas para cada um dos fonemas da língua portuguesa presentes nos sinais de fala.

Este trabalho está sendo generalizado para permitir uma transcrição de fala para um texto equivalente. A base de dados CSLU permitirá a comparação dos resultados obtidos com aqueles publicados em trabalhos relacionados de ASR de língua Portuguesa.

AGRADECIMENTOS

Somos gratos a FAPERJ (Fundação de Amparo a Pesquisa do Estado do Rio de Janeiro, http://www.faperj.br), ao CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico, http://www.cnpq.br) e a CAPES (Coordenação de Aperfeiçoamento de Pessoal de Ensino Superior, http://www.capes.gov.br) por seu continuo apoio financeiro.

REFERÊNCIAS

[1] L. Rabiner e B.-H. Juang, *Fundamentals of speech recognition*. Prentice Hall, 1993.

[2] G. Dede e M. H. Sazlı, "Speech recognition with artificial neural networks", *Digital Signal Processing*, vol. 20, nº 3, pp. 763–768, 2010.

[3] C. M. Bishop, *Neural networks for pattern recognition*. Oxford University Press, 1995.

[4] S. Haykin, *Neural Networks: A Comprehensive Foundation*, sér. Second Edition. New Jersey, NJ, USA: Prentice Hall International, 1999.

[5] S. Grossberg, "Nonlinear neural networks: Principles, mechanisms, and architectures", *Neural networks*, vol. 1, nº 1, pp. 17–61, 1988.

[6] S. Hochreiter, Y. Bengio, P. Frasconi e J. Schmidhuber, "Gradient flow in recurrent nets: The difficulty of learning long-term dependencies", em *A Field Guide to Dynamical Recurrent Networks*, J. F. Kolen e S. C. Kremer, eds., Wiley-IEEE Press, 2001.

[7] Z.-H. Ling, S.-Y. Kang, H. Zen, A. Senior, M. Schuster, X.-J. Qian, H. M. Meng e L. Deng, "Deep learning for acoustic modeling in parametric speech generation: A systematic review of existing techniques and future trends", *Signal Processing Magazine, IEEE*, vol. 32, nº 3, pp. 35–52, 2015.

[8] H. Tabares, J. Branch e J. Valencia, "Dynamic topology generation of an artificial neural network of the multilayer perceptron type", *Revista Facultad de Ingeniería Universidad de Antioquia*, nº 38, pp. 146–162, 2006.