

Reconhecimento de Caracteres Numéricos por meio de Redes Neurais Artificiais do Tipo Perceptron de Múltiplas Camadas

LEONARDO VIEIRA GUIMARÃES
MATHEUS SILVEIRA BORGES

UNIMONTES

19 de Julho de 2017

Sumário

- 1 Introdução, Motivação e Problematização
- 2 Objetivos
- 3 Metodologia
- 4 Resultados
- 5 Considerações Finais
- 6 Referências bibliográficas

- O reconhecimento inteligente de caracteres (ICR) é um segmento da Visão Computacional que tem como objetivo identificar os caracteres de diferentes fontes e caligrafias.
- A principal solução para problemas de ICR é o aprendizado de máquina, onde um algoritmo será treinado para dar soluções inteligentes para reconhecimento de caracteres.

Diversos problemas dependem do reconhecimento de caracteres. Dentre eles, pode-se destacar os seguintes:

- Validação de assinatura manuscrita em documentos;
- Identificação de placas de trânsito por veículo autônomo;
- Inspeção de gravação do número de chassi em veículos;
- Portaria automatizada por meio de identificação do número da placa do carro.

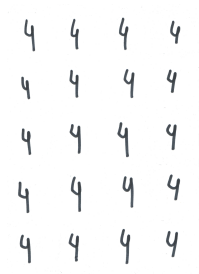
- Entretanto, como relata Menezes (2014), o reconhecimento de caracteres manuscritos ainda é um grande desafio da Visão Computacional.
- A dificuldade de identificar os caracteres manuscritos se dá pelas diversas caligrafias que podem ser utilizadas.
- Menezes (2014) relata também a dificuldade em definir os descritores para os caracteres, visto a complexidade em encontrar padrões nas amostras coletadas.

Em busca de ampliar os conhecimentos sobre a temática e empreender alguma solução, este trabalho teve como objetivos:

- Aplicar os conhecimentos adquiridos durante o curso de Inteligência Computacional para confeccionar uma rede neural do Tipo Perceptron Multi-Camadas (MLP);
- Desenvolver um banco de dados próprio com caracteres numéricos manuscritos;
- Identificar caracteres manuscritos utilizando o algoritmo de treinamento *Back-propagation*;
- Verificar a eficiência do algoritmo *Back-propagation* para reconhecimento de caracteres numéricos manuscritos com base no banco de dados desenvolvido pelo estudo;

Foi desenvolvido um banco de dados próprio a partir de amostras com caracteres manuscritos.

Figura: Exemplo de amostra coletada com o algoritmo 4.



Fonte: Os autores

- Nas amostras, foram contemplados os caracteres de 0 a 9.
- Cada caractere numérico foi escrito manualmente 80 vezes em papel sulfite A4, totalizando 800 amostras. Após, foram feitos os scanners de todas as amostras.
- Cada imagem de amostra foi inicialmente adquirida no formato 700x500, e foi posteriormente redimensionada para 14x12, passando a possuir 168 pixels.
- Assim, cada amostra contém 168 variáveis de entrada, referentes a matriz de pixel de cada imagem de amostra e 4 variáveis de saída.

- Após, as imagens sofreram conversão para escala de cinza e correção das intensidades de preto e branco.
- Com o banco de dados pronto, foram executados 2 treinamentos com os seguintes parâmetros:

Param.goal = 1×10^{-4}

Param.mc = 0.9

Param.lr = 0.1

Param.show = 25

O treinamento foi executado em uma máquina com as seguintes configurações:

Tabela: Configurações da Máquina de Testes.

Parâmetro	Configuração
Sistema Operacional	Windows 10 Pro
Processador	Intel Core i7-5500U CPU 2.40 GHz
Memória RAM	8 GB

Os dois testes foram realizado com 101 amostras de validação em algoritmo MLP com *momentum* e função de ativação senoidal.

- Para o primeiro teste, o banco de dados contava apenas com as 80 amostras inicialmente inseridas.
- Este primeiro teste conquistou 84 acertos e 17 erros, com eficiência de 83,16%. Ele foi concluído em 2hs 39min 58s, com 500.000 épocas de treinamento e erro final de 0,0295.

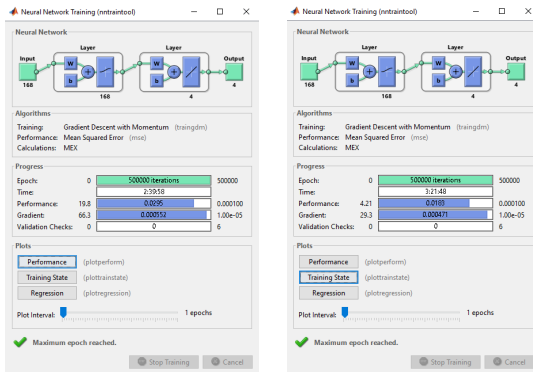
- Entretanto, para o segundo teste, com o intuito de melhorar a eficiência do banco de dados, foi aumentado o número de amostras de cada caractere para 180.
- O segundo teste, por sua vez, obteve 98 acertos e 3 erros, totalizando 97% de eficiência. Esse teste terminou em 3hs 21min 48s, com 500.000 épocas de treinamento e erro final de 0,0183.

Resultados

Treinamento

Após os treinamentos, foram verificados os seguintes resultados:

Figura: Telas obtidas após o treinamento 1 e 2.



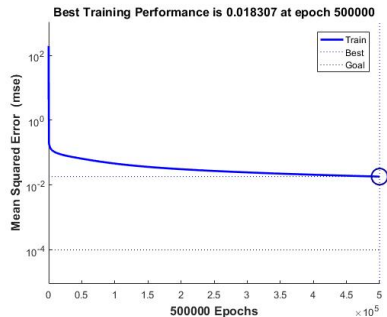
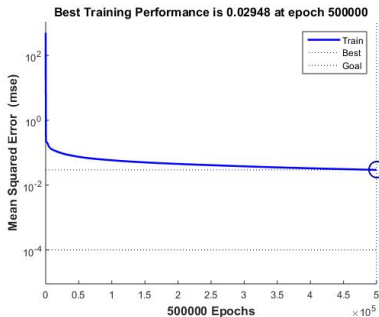
Fonte: Os autores

Resultados

Treinamento

Os resultados de cada treinamento podem ser contemplados nos gráficos abaixo:

Figura: Gráficos do 1º e 2º treinamentos do algoritmo com o erro quadrático médio em função do número de épocas.



Fonte: Os autores

Considerações Finais

- Ao final dos experimentos, foi notável a capacidade de aprendizado e generalização obtida pelo MLP por meio do algoritmo de treinamento *Back-propagation*, conquistando o objetivo da pesquisa, que era identificar caracteres manuscritos utilizando esse algoritmo.
- Também foi notável a diferença entre os resultados do primeiro para o segundo teste, demonstrando que a quantidade de amostras influencia na eficiência da rede neural desenvolvida.
- Não obstante, pesquisas adicionais podem ser feitas com o intuito de melhorar ainda mais a taxa de acerto na validação do algoritmo com base no banco de dados proposto por esse estudo utilizando outros parâmetros para treinamento, inserção de novas caligrafias e o consequente aumento da quantidade de amostras.

- ① BERG, Alexandre Cruz; MULLER, Daniel Nehme; ENGEL, Paulo Martins.
Reconhecimento de Caracteres Usando Redes Neurais. Universidade Federal do Rio Grande do Sul. Porto Alegre: 1993
- ② LIMA JÚNIOR, H. P. **Aplicação de Redes Neurais no reconhecimento de letras em placas de veículos automotores brasileiros.** Centro Brasileiro de Pesquisas Físicas: 2000.