

UNIVERSIDADE FEDERAL DE UBERLÂNDIA – CAMPUS SANTA MÔNICA
CIÊNCIA DA COMPUTAÇÃO
INTELIGÊNCIA ARTIFICIAL
CARLOS CÉSAR MORAIS

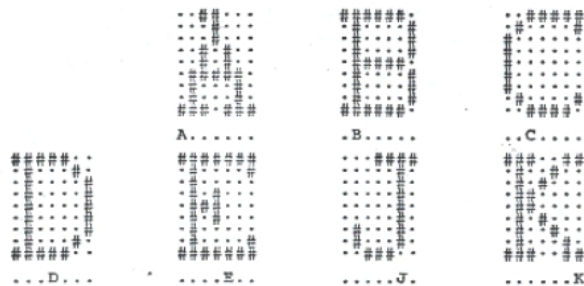
TRABALHO FINAL

UBERLÂNDIA, 2019

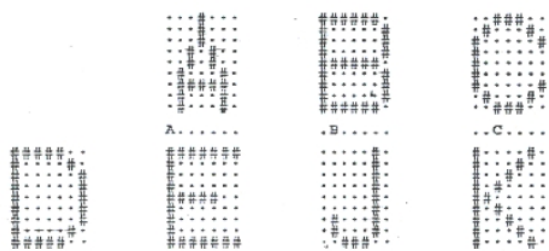
Como motivação das apresentações feitas em sala de aula, e todo conhecimento já adquirido no curso, resolvi implementar um classificador de letra com detecção de ruído. Funciona da seguinte forma, é apresentado um conjunto inicial de letras do alfabeto em forma de matriz. Com isso, é feito um treinamento de uma rede neural, no caso a utilizada nesse projeto foi a Adeline, e com isso aplicar a regra Delta, para que seja feito a análise da entrada e poder fornecer com a quantidade de ruído que a letra possui e de fato indentificar a letra.

Então, neste trabalho a rede deverá reconhecer as letras do alfabeto A, B, C, D, E, J e K. Também é estabelecido três formatos de fontes diferentes da letra.

• Fonte 1



• Fonte 2



• Fonte 3



Considerando que cada entrada é uma matriz de 7X9, ou melhor, um vetor de 63 posições; então a rede terá 63 neurônios de entrada, mais o bias. E ainda, a rede terá 7 neurônios de saída: cada neurônio faz o reconhecimento de uma letra. Para o treinamento da rede, utilizou - se a regra Adaline, 63 neurônios de entrada (mais o Bias), erro mínimo de 0,001 e devioPadrao = 0,02. Para testar a rede o usuário deve aplicar o caracter “#” onde queira que seja preenchido e aplicar “0” onde queira aplicar ruído.

Implementação do código foi feita utilizando ao IDE VisualStudio e utilizando a linguagem de programação C#. Toda implementação segue em anexo a este arquivo.