

Especificação do trabalho de IA

O trabalho de Inteligência Artificial versa sobre a rede neural artificial Multilayer Perceptron (MLP), a rede neural artificial Convolutional Neural Network (CNN) e abrange a prática com modelagem, implementação, testes e análise de resultados. Observe as regras abaixo e na sequência, verifique os objetivos do trabalho e mais detalhes sobre como devem ser as implementações e entregas.

Algumas regras gerais

- Os alunos devem se organizar em grupos de até **seis** integrantes.
- Todas as entregas deverão ser feitas via Sistema e-Disciplinas, dentro da data limite estabelecida neste documento (e no syllabus).
 - Qualquer eventual problema com o sistema e-Disciplina deve ser observado com antecedência suficiente para que alguma solução de contorno possa ser estabelecida em conjunto com a professora. Isso significa que o grupo não deve deixar para fazer *upload* de arquivos no último minuto possível. Preferencialmente, o *upload* deve ser feito com uma antecedência mínima de um dia. O último dia de entrega deve ser deixado para *upload* de arquivos que representam apenas ajustes finos no trabalho.
- O grupo deverá produzir três vídeos, com conteúdo referente a requisitos detalhados abaixo. **Todos os alunos do grupo deverão participar dos três vídeos**, sendo que cada aluno deve ficar responsável por explicar algum aspecto do trabalho tratado em cada vídeo. Uma janela com a imagem do aluno realizando a explicação deverá fazer parte dos vídeos.
- Slides que eventualmente sejam usados no vídeo podem ser disponibilizados no sistema e-Disciplinas (esse não é um requisito mandatório, já que a produção do vídeo não necessariamente é baseada em slides).
- Os códigos desenvolvidos pelo grupo deverão ser postados no sistema e-disciplinas.
- Os artefatos entregues (código, vídeos, slides, pdfs) deverão conter o **nome e número USP de todos os integrantes do grupo**.
- **Todos os artefatos devem ser cuidadosamente construídos, com boa estética de apresentação.** Artefatos apresentados com desmazelo/desleixo e de forma negligente poderão implicar em desconto de nota ou atribuição de nota 0 ao artefato.
- Data limite para a postagem dos arquivos do trabalho MLP: **4 de junho**.
- Data limite para a postagem dos arquivos do trabalho CNN: **25 de junho**.

Requisitos para o vídeo 1 (MLP): o objetivo do vídeo 1 é propiciar a avaliação das implementações da rede neural artificial MLP em termos de seu funcionamento.

- Tempo de vídeo: mínimo de 10 minutos e máximo de 20 minutos.
- Conteúdo: ilustração da execução dos códigos para:
 - Treinamento (sem e com **validação cruzada** e **parada antecipada**) da MLP para o conjunto de dados CARACTERES COMPLETO (os conjuntos de

- portas lógicas e o conjunto CARACTERES da Fausett servirão apenas para teste de implementação, então não precisam entrar nos vídeos).
 - Estudo dos parâmetros (busca por valores adequados – grid)
 - Teste da MLP para o conjunto de dados CARACTERES COMPLETO.
- Detalhes importantes que devem ser comentados no vídeo:
 - Que parte dos conjuntos de dados está sendo usada para treinamento.
 - Que parte dos conjuntos de dados está sendo usada para validação (quando for o caso).
 - Que parte dos conjuntos de dados está sendo usada para teste.
 - Apresentação do console (ou de um elemento de interface gráfica) para um acompanhamento, ainda que resumido (ou seja, parte da execução), do treinamento da rede MLP.
 - Apresentação do resultado da rede MLP, no console (ou um elemento de interface gráfica), para os dados de teste.
 - Apresentação do resultado da busca por parâmetros – gráficos/tabelas
 - Apresentação da matriz de confusão resultante do teste das redes.
- Outros detalhes que o grupo considerar importantes para agregar valor ao seu trabalho podem ser inseridos desde que se mantenham dentro do tempo máximo de vídeo.
- **IMPORTANTE:** o tempo de vídeo é um critério de estabelecimento de avaliação igualitária entre os alunos, ou seja, vídeos “acelerados” implicarão em descontos na nota, podendo inclusive ser um fator determinante para a atribuição de nota 0 (zero) para esse quesito do trabalho. Uma possibilidade pode ser acelerar o vídeo enquanto mostra a execução de um treinamento, **mas a gravação da voz deve estar em velocidade normal.**
- **IMPORTANTE:** você pode disponibilizar o seu vídeo em qualquer repositório público ou no sistema e-Disciplinas. **Se você disponibilizar em um repositório público** (e não no sistema e-Disciplinas) você deverá depositar no sistema e-Disciplinas, até a data limite de entrega do trabalho, um arquivo **.pdf** com a minutagem do seu vídeo (com minuto de início e de fim para cada trecho de sua explicação), considerando os requisitos importantes de conteúdo solicitado para aquele vídeo.

Requisitos para o vídeo 2 (MLP): o objetivo do vídeo é propiciar a avaliação do entendimento dos alunos quanto ao conteúdo “redes neurais artificiais” e “multilayer perceptron”.

- Tempo de vídeo: mínimo de 10 minutos e máximo de 20 minutos
- Conteúdo - apresentação de como os seguintes conceitos foram implementados no código (variáveis e procedimentos):
 - Camada de entrada da rede neural.
 - Camada de saída da rede neural.
 - Camada escondida da rede neural.
 - Função (ou funções) de ativação dos neurônios.
 - Taxa de aprendizado.

- Termos de regularização ou procedimentos de otimização adaptativos (se estiver sendo usado).
 - Loop do procedimento de treinamento.
 - Critério de parada do treinamento.
 - Conjunto de pesos sinápticos em cada camada.
 - Atualização dos pesos sinápticos em cada camada.
 - Procedimentos de cálculo de erro na camada de saída (para cada neurônio e para a rede neural).
 - Procedimento de cálculo de informação de erro para a retropropagação.
 - Procedimento de cálculo de contribuição de erro na camada escondida.
 - Procedimento de cálculo da resposta da rede em termos de reconhecimento do caractere.
 - Procedimento de cálculo da matriz de confusão.
- Outros detalhes que o grupo considerar importantes para agregar valor ao seu trabalho podem ser inseridos desde que se mantenham dentro do tempo máximo de vídeo.
 - **IMPORTANTE:** o tempo de vídeo é um critério de estabelecimento de avaliação igualitária entre os alunos, ou seja, vídeos “acelerados” implicarão em descontos na nota, podendo inclusive ser um fator determinante para a atribuição de nota 0 (zero) para esse quesito do trabalho. Uma possibilidade pode ser acelerar o vídeo enquanto mostra um treinamento, **mas a gravação da voz deve estar em velocidade normal.**
 - **IMPORTANTE:** você pode disponibilizar o seu vídeo em qualquer repositório público ou no sistema e-Disciplinas. **Se você disponibilizar em um repositório público** (e não no sistema e-Disciplinas) você deverá depositar no sistema e-Disciplinas, até a data limite de entrega do trabalho, um arquivo **.pdf** com a minutagem do seu vídeo (com minuto de início e de fim para cada trecho de sua explicação), considerando os requisitos importantes de conteúdo solicitado para aquele vídeo.

Requisitos para o vídeo 3 (CNN): o objetivo do vídeo 3 é propiciar a avaliação das implementações da rede neural artificial CNN em termos de seu funcionamento.

- Modelagem das entradas da CNN.
- Camadas de aplicação de kernel e pooling, com detalhamento sobre o kernel e pooling aplicados.
- Modelagem das camadas densas e camada de saída.
- Lógica do treinamento da estrutura completa.
- Critério de parada do treinamento.
- Procedimentos de cálculo de erro na camada de saída.
- Procedimento de cálculo da resposta da rede em termos de reconhecimento de caractere.
- Teste da CNN para o conjunto de dados CARACTERES ou MNIST.
- Procedimento de cálculo da matriz de confusão.

- Outros detalhes que o grupo considerar importantes para agregar valor ao seu trabalho podem ser inseridos desde que se mantenham dentro do tempo máximo de vídeo.
- **IMPORTANTE:** o tempo de vídeo é um critério de estabelecimento de avaliação igualitária entre os alunos, ou seja, vídeos “acelerados” implicarão em descontos na nota, podendo inclusive ser um fator determinante para a atribuição de nota 0 (zero) para esse quesito do trabalho. Uma possibilidade pode ser acelerar o vídeo enquanto mostra um treinamento, **mas a gravação da voz deve estar em velocidade normal.**
- **IMPORTANTE:** você pode disponibilizar o seu vídeo em qualquer repositório público ou no sistema e-Disciplinas. **Se você disponibilizar em um repositório público** (e não no sistema e-Disciplinas) você deverá depositar no sistema e-Disciplinas, até a data limite de entrega do trabalho, um arquivo **.pdf** com a minutagem do seu vídeo (com minuto de início e de fim para cada trecho de sua explicação), considerando os requisitos importantes de conteúdo solicitado para aquele vídeo.

Objetivo 1: Multilayer Perceptron

Implementar uma rede neural artificial Multilayer Perceptron (MLP) - sem fazer uso de bibliotecas especializadas em redes neurais artificiais - com **uma** camada escondida e treinada com o algoritmo **Backpropagation em sua versão de Gradiente Descendente** - algoritmo de treinamento **discutido em sala de aula.**

Quatro **conjuntos de dados** devem ser usados para treinamento e teste:

- Conjuntos de dados OR, AND e XOR (para testes rápidos de corretude de implementação)
- Conjunto de dados CARACTERES COMPLETO (similar ao conjunto visto em aula)
 - O conjunto CARACTERES COMPLETO possui a versão limpa e as versões com ruídos. As versões com ruído são adequadas para usar nos testes.

Nos problemas de portas lógicas, a última coluna é o rótulo do dado, as demais são atributos descritivos. No problema dos caracteres, as últimas colunas compõem o rótulo (eventualmente, o grupo pode testar formas diferentes de trabalhar com composição de valores para construir o rótulo).

Arquivos de saída úteis para o seu trabalho:

- Um arquivo contendo os hiperparâmetros finais da arquitetura da rede neural e hiperparâmetros de inicialização.
- Um arquivo contendo os pesos iniciais da rede.
- Um arquivo contendo os pesos finais da rede.
- Um arquivo contendo o erro cometido pela rede neural em cada iteração do treinamento.
- Um arquivo contendo as saídas produzidas pela rede neural para cada um dos dados de teste realizados

Algumas regras referentes ao “objetivo 1”

- As implementações deverão ser feitas em linguagens baseadas em Java, C (e variações), Python, Matlab, Octave ou R.
- O código deve ser sempre muito bem documentado (em DETALHES) de forma que seja simples identificar passagens do código que são importantes para a verificação do entendimento do grupo sobre a lógica que implementa uma rede neural artificial. Isso significa que o uso de alguma biblioteca que implementa as funções de uma rede neural artificial pode se tornar extremamente complexo (você terá que explicar em detalhes o código da biblioteca, e por vezes, isso pode ser mais trabalhoso do que fazer o seu próprio código).
- Bibliotecas que implementem funções de I/O, operações de matrizes e funções de PLOT podem ser livremente usadas para implementar entrada de dados e interface para exposição de resultados (a interface pode ser construída em modo gráfico, modo texto ou apenas com uso de prompt e gravação das saídas do algoritmo em arquivos .txt ou .csv).

Objetivo 2: Convolution Neural Network

Implementar uma rede neural artificial Convolution Neural Network (CNN) – aqui você pode fazer uso de bibliotecas especializadas em redes neurais artificiais - com várias camadas escondidas e camada densa ao final da estrutura.

O conjunto de dados escolhido deve ser usado para teste da rede neural, considerando duas tarefas:

- Uma tarefa multiclasse, ou seja, é esperado que a rede neural saiba responder corretamente para todas as classes do conjunto.
- Uma tarefa de classificação binária, ou seja, apenas duas classes, ou duas composições de várias classes, devem ser usados no treinamento e a rede deverá reconhecer essas duas classes. Você pode variar os pares de classes em seus testes.

O conjunto MNIST pode ser obtido em:
<https://www.tensorflow.org/datasets/catalog/mnist?hl=pt-br>

A implementação da CNN deve ser feita com um framework à escolha do grupo.

O grupo pode testar sua implementação usando os dados em sua forma bruta ou então usando características extraídas dos dados (por exemplo, usando descritor HOG - *Histogram of Oriented Gradients*, LBP – *Local Binary Patterns*, Histograma de Blocos ou a transformada *wavelet* de Haar). O grupo deve obrigatoriamente apresentar uma das duas formas de trabalho. Apresentar as duas é benéfico para a avaliação do trabalho do grupo. A extração do descritor HOG ou qualquer outro descritor deve ser feita com uso de bibliotecas especializadas para garantir a correteza do resultado.

Arquivos de saída úteis para o seu trabalho:

- Um arquivo contendo os hiperparâmetros da arquitetura da rede neural e hiperparâmetros de inicialização.
- Um arquivo contendo os pesos iniciais da rede.
- Um arquivo contendo os pesos finais da rede.
- Um arquivo contendo o erro cometido pela rede neural em cada iteração do treinamento.
- Um arquivo contendo as saídas produzidas pela rede neural para cada um dos dados de teste.

Bibliotecas para manipulação de imagens (onde estão implementados os descritores):

- OpenCV: *Open Source Computer Vision*, disponível em <http://opencv.org/>, com interfaces para C++, C, Python, Java e MATLAB, sobre Windows, Linux, Mac OS e Android.
- scikit-image: disponível em <http://scikit-image.org/>, com interface para Python. Os responsáveis pedem gentilmente para que o uso da biblioteca seja citado. <http://dx.doi.org/10.7717/peerj.453>.
- MATLAB possui um pacote com funções úteis em processamento de imagens: *Computer Vision System Toolbox*.

Este é um documento preliminar e pode sofrer alterações ao longo do tempo no sentido de torná-lo mais didático e mais detalhado.