

# INF1771 - INTELIGÊNCIA ARTIFICIAL

## TRABALHO 3 – APRENDIZADO DE MÁQUINA

### Descrição:

“Até o momento existem **718 espécies** conhecidas de pokémons. Cada pokémon possui uma propriedade que define o seu **tipo**. No total, existem **18 tipos** (Figura 1), e um pokémon pode ter um ou dois tipos diferentes. Por exemplo, o Charmander é somente do tipo Fire, enquanto que o Bulbasaur é simultaneamente do tipo Grass e Poison. Constantemente novos pokémons são descoberto e é extremamente importante que os seus tipos sejam devidamente catalogados.

Você é o novo estagiário no laboratório do professor Carvalho e está responsável pelo desenvolvimento de um **novo algoritmo para a Pokédex** (Figura 2).

O seu algoritmo deve ser capaz de **reconhecer automaticamente o tipo de pokémons desconhecidos.**”



Figura 1. Tipos de pokémon.

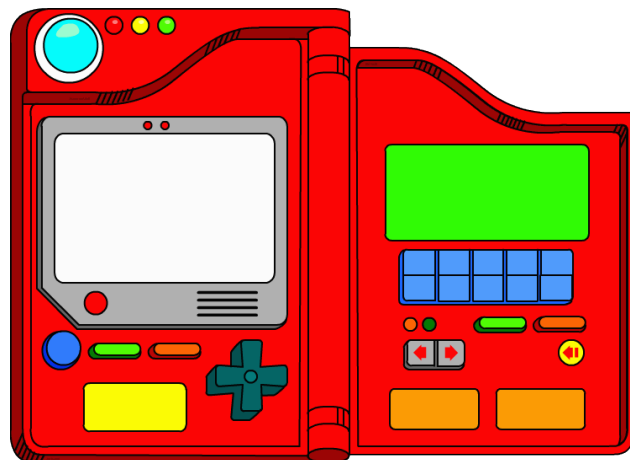


Figura 2. Pokédex.

O **objetivo do Trabalho 3** é desenvolver o algoritmo de reconhecimento de tipos de pokémons utilizando a base de dados de pokémons conhecidos. O sistema deve receber como entrada as características do pokémon e ser capaz de identificar automaticamente o seu tipo.

Para essa tarefa, você tem a disposição o conjunto de dados da **Bulbapedia** sobre as **718 espécies** conhecidas de pokémons. Para desenvolver o sistema de reconhecimento de tipos de pokémons, você deve utilizar um método de **aprendizado de máquina**

**supervisionado**, visto que podemos criar um conjunto rotulado de exemplos para treinamento. Para isso, você deve seguir os seguintes passos:

- 1) Definir quais atributos serão usados para descrever os exemplos de treinamento. Para isso você deve analisar as informações disponíveis na Bulbapedia e escolher os atributos mais interessantes.
- 2) Extrair automaticamente ou manualmente os atributos escolhidos dos pokémons conhecidos da Bulbapedia, de forma a gerar um conjunto de treinamento e um conjunto de validação. Normalmente gera-se um único conjunto de dados e depois se divide em dois conjuntos (treinamento e validação). Cada pokémon conhecido será um exemplo de treinamento.
- 3) Utilizar pelo menos 4 algoritmos de aprendizado supervisionado, treinando-os com o conjunto de treinamento e depois realizando a classificação do conjunto de testes para verificar qual algoritmo apresenta a melhor taxa de reconhecimento. Devem ser utilizados, no mínimo, os seguintes algoritmos:
  - Árvores de Decisão
  - K-Nearest Neighbor (KNN)
  - Support Vector Machine (SVM)
  - Rede Neural (usando backpropagation)

Não é necessário programar os algoritmos, é permitida a utilização de ferramentas de aprendizado de máquina, como por exemplo, o Weka (<http://www.cs.waikato.ac.nz/ml/weka/>).

Se a taxa de reconhecimento estiver muito baixa para todos os algoritmos, deve-se retornar para a etapa 1 e selecionar melhor os atributos para descrever os exemplos de treinamento.

- 4) Implementar o algoritmo que apresentar melhores resultados na linguagem de sua preferência. Árvores de Decisão e KNN são os mais simples de serem implementados. No caso do SVM, é permitido o uso da biblioteca LibSVM (<http://www.csie.ntu.edu.tw/~cjlin/libsvm/>). No caso de uma rede neural, é permitido o uso da biblioteca FANN (<http://leenissen.dk/fann/wp/>).

## Conjunto de Dados:

A Bulbapedia possui uma enorme quantidade de informações sobre todos os pokémons conhecidos. A seguir estão listadas algumas sugestões de dados que podem ser uteis para a definição dos atributos que serão utilizados:

- Lista principal de pokémons e tipos: <http://goo.gl/6ifWYL>
- Lista de pokémons por altura: <http://goo.gl/6VVgkU>
- Lista de pokémons por peso: <http://goo.gl/v5cJI9>
- Lista de pokémons por status base: <http://goo.gl/volYYI>
- Lista de pokémons por estilo de corpo: <http://goo.gl/m8DqyZ>
- Lista de pokémons por cor: <http://goo.gl/pXBfIG>
- Lista de pokémons por habitat: <http://goo.gl/V8v8eo>
- Lista de pokémons por habilidades: <http://goo.gl/haeRSX>
- Lista de pokémons por performance: <http://goo.gl/NGN2II>
- Lista de pokémons por taxa de captura: <http://goo.gl/6h7QVV>
- Lista de pokémons por rendimento/esforço: <http://goo.gl/xtfvQQ>
- Lista de pokémons por amizade base: <http://goo.gl/TM6byr>

Você pode utilizar outras informações disponíveis na Bulbapedia, desde que elas não estejam diretamente ligadas ao tipo do pokémon.

## Informações Adicionais:

- Durante a etapa de teste dos 4 algoritmos não é necessário implementar todos os classificadores para realizar os experimentos. É permitida a utilização de ferramentas de aprendizado de máquina, como por exemplo, o Weka (<http://www.cs.waikato.ac.nz/ml/weka/>).
- A divisão dos exemplos de treinamento e teste deve ser feita de forma aleatória. Normalmente 70% dos exemplos devem ser utilizados para treinamento e 30% para testes. Você deve garantir também que os exemplos de cada classe sejam distribuídos nestas mesmas proporções nos dois conjuntos.
- A implementação de um dos algoritmos pode ser feita em qualquer linguagem (C/C++, C#, Java...).
- Você deve decidir quais atributos serão utilizados pelos classificadores. A classificação pode piorar ou melhorar dependendo do conjunto de atributos utilizado. Você deve escolher quais atributos serão utilizados e fazer testes para verificar qual conjunto de atributos apresenta melhores resultados.

- O trabalho pode ser feito **individualmente** ou em **grupos** de no máximo 3 pessoas.
- **IMPORTANTE:** O programa deve ser apresentado durante a aula por **todos os membros do grupo**:
  - O membro do grupo que **não comparecer** receberá nota **zero**;
  - **Todos os membros** do grupo perderam **5.0 pontos** se alguém do grupo **não souber explicar** algo relacionado ao trabalho.

### **Forma de Avaliação:**

Será avaliado se todas as etapas do processo foram cumpridas corretamente. A avaliação também será baseada na **apresentação dos resultados** durante a aula.

Essa apresentação deverá conter:

- Descrição da modelagem dos exemplos de treinamento:
  - Atributos selecionados para descrever os exemplos;
  - Justificativa para a escolha dos atributos;
  - Estrutura dos exemplos;
- Descrição dos experimentos realizados:
  - Variações na modelagem dos exemplos;
  - Variações no conjunto treinamento e testes;
  - Variação nos parâmetros dos algoritmos;
- Comparação dos algoritmos analisados:
  - Taxa de reconhecimento;
  - Tempo gasto no processo de treinamento;
  - Tempo gasto no processo de classificação de um exemplo desconhecido;
- Implementação:
  - Descrição de como o algoritmo escolhido foi implementado;
  - Resultados alcançados pelo algoritmo que foi implementado (tempo gasto no processo de treinamento, tempo gasto no processo de classificação de um exemplo desconhecido, taxa de reconhecimento).

**Bônus:**

- O trabalho que conseguir a maior taxa de reconhecimento receberá 2.0 pontos extras na nota.
- O trabalho que implementar uma Rede Neural ou SVM sem utilizar nenhuma biblioteca externa receberá 2.0 pontos extras na nota.

**Data de Entrega:**

02/12

**Forma de Entrega:**

Os trabalhos devem ser **apresentados** na aula do dia 02/12 (segunda) e 04/12 (quarta).

Todos devem enviar o trabalho até o dia 04/12 para o email [edirlei.slima@gmail.com](mailto:edirlei.slima@gmail.com).

**Não serão aceitos trabalhos enviados depois desta data.**