

Inteligência Artificial

Prof. Fabrício Olivetti de França, Prof. Denis Fantinato
3º Quadrimestre de 2019

Inteligência Artificial

Inteligência corresponde a habilidade de obter e aplicar novos conhecimentos ou habilidades.

Ou ainda: Perceber, entender, predizer e manipular.

Somos seres inteligentes pois aprendemos, avançamos, desenvolvemos.

Um dos campos de pesquisa mais recentes da computação (embora já tenha cerca de 70 anos).

Tenta entender **E** construir agentes inteligentes.

Diversas definições e ramificações:

- Racionalização
- Comportamento

ou

- Desempenho comparável a de um humano
- Desempenho desejável (possivelmente acima de um humano)

Famoso **teste de Turing** com o objetivo de fazer com que um agente inteligente se passe por um ser humano.

Tenta nos imitar!

Quem é o humano?

<https://www.youtube.com/watch?v=Qv4esGWOg7w>

Prêmio Loebner - Chatbot Mitsuku

Raciocínio Humano

Para replicarmos a inteligência humana, primeiro precisamos entendê-la.

Estudos de **ciência cognitiva**, não basta criarmos um agente que gera um raciocínio correto, mas deseja-se que ele siga a mesma linha de raciocínio que um humano.

Um outro ramo é o de criar máquinas que tenham um raciocínio lógico capaz de gerar provas irrefutáveis.

Notação lógica como uma notação precisa sobre todos os objetos do mundo e suas relações.

Expectativa: algoritmo que pode resolver qualquer problema utilizando linguagem lógica.

Realidade: incertezas, escalabilidade.

Agente racional

Um **agente racional** ou **agente inteligente** é aquele que age e reage de acordo com o ambiente para obter o melhor resultado (ou o melhor resultado esperado, quando o ambiente possui incertezas).

Não necessariamente depende da existência de uma resposta correta, o agente deve executar uma ação independente disso.

Definições:

- **Ser Racional:** atingir o máximo de uma ou mais metas pré-definidas.
- **Racionalidade:** quais decisões foram tomadas, não o motivo.
- **Utilidade:** função que define o objetivo.

Ou seja, ser racional significa maximizar a utilidade esperada.

História da Inteligência Artificial

A área de pesquisa em Inteligência Artificial surgiu do desejo dos pesquisadores em criar e entender consciência.

Inicialmente acreditava-se que o segredo para construir IA estava em entender nosso próprio cérebro.

McCulloch e Pitts criaram o primeiro modelo de neurônio artificial em 1943.

O modelo proposto era de neurônios interconectados que estavam *ligados* ou *desligados* de acordo com o estímulo recebido pelos neurônios vizinhos.

Eles mostraram que qualquer função computável poderia também ser computada por uma rede de neurônios do modelo proposto e que as funções lógicas E, Ou, Não, Ou-exclusivo, poderiam ser implementadas com redes simples.

Neurônios que aprendem

Esses autores também comentaram que tais redes de neurônio poderiam aprender. Hebb, em 1949, demonstrou isso com a seguinte regra de atualização:

$$w_{i,j}(t + 1) = w_{i,j}(t) + \eta x_i(t)x_j(t)$$

Ou seja, se dois neurônios interligados se tornam ativos simultaneamente, o sinal na sinapse é reforçado.

Minsky e Edmonds construíram o primeiro computador neural, SNARC, em 1950 simulando uma rede neural com 40 neurônios.

Alguns anos depois Minsky demonstrou algumas limitações da pesquisa em percéptrons, principalmente a falta de não-linearidade.

Isso esfriou um pouco os estudos das Redes Neurais...

Teste de Turing

No artigo **Computing, Machinery and Intelligence**, Alan Turing introduziu o famoso teste de Turing além de ideias sobre aprendizado de máquina, algoritmos genéticos e aprendizado por reforço.

“E se ao invés de tentarmos criar programas que simulam a mente de um adulto, não tentamos criar um que simule a mente de uma criança?”

Pesquisadores famosos da área se reuniram durante 2 meses em Dartmouth College para discutir sobre as possibilidades e futuro da Inteligência Artificial.

Essa reunião despertou o interesse de diversas empresas que investiram na ideia.

Logic Theorist

Nesse encontro, dois pesquisadores da Universidade de Carnegie, Newell e Simon, mostraram seu trabalho mais recente, o **Logic Theorist**, capaz de provar a maioria dos teoremas do capítulo 2 do livro *Principia Mathematica*, de A. Whitehead e B. Russell.

Em seguida, submeteram um trabalho de co-autoria entre Newell, Simon e o programa Logic Theorist. Os editores do *Journal of Symbolic Logic* rejeitaram o artigo.

General Problem Solver

Seguindo o sucesso do *Logic Theorist* (não no journal de Symbolic Logic), Newel e Simon criaram o General Problem Solver, a ideia era resolver problemas da mesma forma que nós, seres pensantes, resolvemos: análise de meios para se chegar a um fim.

General Problem Solver

Exemplo dado no artigo:

*Quero levar meu filho para a escola. Qual a diferença entre o que eu tenho e o que eu quero? Um passo de distância. O que muda a distância? Meu carro. Meu carro quebrou. O que preciso fazer para ele funcionar? Uma nova bateria. Onde encontro uma nova bateria? Na oficina mecânica. Eu quero que um mecânico coloque a bateria no meu carro, mas o mecânico não sabe que eu quero isso. Qual o problema? De comunicação. O que permite comunicação? Um telefone...e assim por diante. — Newell e Simon. book *Human Problem Solving* (1972).*

Com os resultados iniciais, muitas empresas e governos passaram a investir nas pesquisas da área (principalmente o DARPA).

Muitos *cheques em branco* apostando nas direções que os pesquisadores achassem ser a mais interessante.

Durante as pesquisas alguns problemas surgiram:

- Os algoritmos criados tinham como objetivo resolver **qualquer** problema.
- Utilizavam manipulação sintática simples (estrutura e forma), sem pensar na semântica (significado).
- Ilusão do poder computacional ilimitado.

Tradutor de russo para inglês, na época do Sputnik:

O espírito tem força de vontade mas a carne é fraca

se tornou

A vodka é boa mas a carne é podre

O financiamento do governo americano foi encerrado.

Muitos pesquisadores tinham a ideia de um micromundo: se seu algoritmo resolvesse um problema pequeno, para resolver um problema grande bastaria aumentar o poder computacional.

Porém não contavam com a explosão combinatória do aumento dos problemas.

Um dos primeiros experimentos de algoritmos que evoluem foi feito em 1958 por Friedberg. Ele acreditava que se fizesse pequenas *mutações* em um código de máquina, eventualmente ele resolveria alguma tarefa alvo.

Milhares de horas de processamento depois, nada foi encontrado...

Minsky destacou em um livro que o Percéptron simples podia resolver qualquer problema representável por ele...porém poucos problemas podiam ser representados como um percéptron simples.

As pesquisas sobre Redes Neurais perderam o financiamento.

Com isso, surgiram os sistemas baseados em conhecimento e sistemas especialistas.

Uso de uma base de conhecimento extensa e regras de decisão bem definidas feitas por especialistas.

Esses novos sistemas fomentaram a criação de grupos de estudos sobre **Heurísticas**, que são funções que representam intuições razoáveis para um determinado problema.

Na mesma época surgiu a **lógica Fuzzy** capaz de modelar sistemas lógicos com incertezas.

Sistemas especialistas e de lógica Fuzzy passaram a dominar produtos eletrônicos na década de 80.

Os Conexionistas

Durante a década de 70, diversos pesquisadores *reinventaram* o algoritmo de derivação automática chamado de **Retropropagação**.

Isso permitiu o retorno das Redes Neurais e dos modelos **conexionistas** na área de Inteligência Artificial.

Avanços mais recentes levaram a possibilidade de trabalhar com grandes redes neurais e modelos profundos (**Deep Learning**).

Provavelmente Inteligente

Durante a década de 80 muitos pesquisadores passaram a incorporar modelos probabilísticos para gerar inteligência.

Com isso foi possível tratar incertezas.

Na década de 90, ressurgiu o interesse no conceito de *agentes autônomos* e com a popularização da Internet, surgiram os *bots*.

Isso permitiu a automatização de diversas tarefas do dia a dia e até de nosso assistente pessoal dentro de um Smartphone.

Finalmente, com uma maior capacidade computacional, computação distribuída e disponibilidade dos dados, muitos algoritmos que, anteriormente, não apresentavam um bom desempenho passaram a ser mais competentes que algoritmos mais elaborados.

Hoje já podemos ver avanços incríveis em:

- Veículos autônomos
- Automatização de tarefas
- Visão computacional
- Agentes que aprendem a jogar jogos complexos
- Logística
- Tradução de idiomas

e muito mais...

Sobre a disciplina

Sobre as aulas

- Aulas teóricas com alguns códigos exemplos.
- Exercícios propostos sobre a aula para refletir e praticar.
- Projeto.
- Implementação de alguns dos algoritmos em Python.

[http:
//professor.ufabc.edu.br/~denis.fantinato/teaching/IA](http://professor.ufabc.edu.br/~denis.fantinato/teaching/IA)

- 03 Provas teóricas (1 hora de duração), cada uma valendo 10 pontos
- Projeto = até 5 pontos a serem distribuídos para as notas da prova (+ bônus de competição)

Nota - Conceito

p1 - 18/10/2019

p2 - 08/11/2019

p3 - 06/12/2019

Será calculada a média harmônica das provas:

$$H(p1, p2, p3) = \frac{3}{\frac{1}{p1} + \frac{1}{p2} + \frac{1}{p3}}$$

Nota - Conceito

Projeto (em grupo, até 3 integrantes) valendo até 5 ptos a serem distribuídos nas notas da prova.

Bônus de Competição:

Primeiro lugar UFABC: +1 pto Segundo lugar UFABC:
+0.6 pto Terceiro lugar UFABC: +0.3 pto

Se ganharem a competição internacional: +0.6 para todos

Importante: Plágio no projeto implicará em reprovação.

Nota - Conceito

```
conceito :: Double -> Char
```

```
conceito nota
```

```
| nota >= 8.5 = 'A'
```

```
| nota >= 7 = 'B'
```

```
| nota >= 6 = 'C'
```

```
| nota >= 5 = 'D'
```

```
| otherwise = 'F'
```

O: Se frequência < 75% (Resolução ConsEPE 139)

- Quartas-feiras, das 16:00 às 17:00 e das 19:00 às 21:00, Sala 509-2
- Sextas-feiras, das 11:00 às 12:00, L113, Bloco L

A Prova Substitutiva abordará todo o conteúdo da disciplina e substituirá uma das provas em que houve falta.

A prova substitutiva poderá ser feita pelos alunos que não puderam fazer alguma prova pelos motivos descritos na Resolução ConsEPE 227.

Recuperação

Prova teórica compreendendo todo o conteúdo da disciplina.

$nota = 0.5 * H(p1, p2, p3) + 0.5 * recuperação$

Conversão nota - conceito:

```
conceito :: Double -> Char
```

```
conceito nota
```

```
| nota >= 7 = 'C'
```

```
| nota >= 5 = 'D'
```

```
| otherwise = 'F'
```

No site da disciplina.