



FUNDAMENTOS DE COMPUTAÇÃO

AULA 4



Prof. eng. Gian Carlo Brustolin M.Sc.



CONVERSA INICIAL

Nesta aula iremos apresentar um apanhado geral de inteligência artificial partindo dos fundamentos históricos e da taxonomia clássica. Evoluiremos até o entendimento dos processos heurísticos aplicados em sistemas especialistas e concluiremos com o estudo de técnicas básicas de treinamento e rudimentos de aprendizagem de máquina (Machine Learning).

Concluindo este passeio pelas potencialidades da inteligência artificial apresentaremos algumas ferramentas comerciais que implementam as técnicas que estudamos.

Vamos, então, iniciar nosso estudo. Por se tratar de uma abordagem bastante rudimentar dos temas, recomendamos a você, caso deseje aprofundá-los o mergulho na literatura indicada ao final deste capítulo.

TEMA 1: INTELIGÊNCIA ARTIFICIAL E MACHINE LEARNING: HISTÓRICO E CONCEITOS

Apesar do tema de inteligência artificial (IA) habitar nossas conversas cotidianas, como se o conhecêssemos plenamente, não há ainda um conceito sedimentado sobre esta ciência. A própria definição de inteligência humana, a qual buscamos reproduzir na IA, é ainda lacunosa. Apesar desta incerteza, as pesquisas em torno da IA evoluíram por aderirem a uma abordagem mais pragmática, buscando reproduzir os “efeitos” da inteligência. Vamos agora conhecer um pouco da história científica deste tema.

1.1 Conceito de Inteligência

As tentativas de definir inteligência remontam à Grécia Clássica, com Aristóteles, que pretendia definir a inteligência como a capacidade de pensar racionalmente. Esta definição esbarra na compreensão do pensamento e, particularmente, de um tipo específico de pensamento. O pensamento racional, segundo o filósofo, seria obtido pela disciplina lógica, ou seja, seria o resultado



de um treinamento da mente humana no sentido de condiciona-la ao uso da lógica como norte do pensar. A famosa técnica do silogismo aristotélico, posteriormente revisitado por Descartes na consolidação do pensamento cartesiano, é originado desta ideia de pensamento correto.

Alguns investigadores hodiernos, como o neurocientista português Antônio Damásio, localizam o processo do raciocínio (e consequentemente da inteligência) como uma associação de redes especializadas de neurônios (DAMASIO, 2012). Esta descrição fisiológica, embora desvende a topografia cerebral do raciocínio ainda não o explica como fenômeno em si. Vamos, por hora, aceitar a ideia aristotélica do treinamento lógico e voltar ao “conforto” da computação, para entender as máquinas inteligentes.

A melhor aproximação que temos para a definição de uma máquina inteligente foi proposta por Alan Turing em 1950 (NORVIG, 2013, pg.1). Na verdade, não se trata propriamente de uma definição, mas de uma constatação do comportamento inteligente. O chamado teste de Turing compara as reações de uma máquina computacional com o comportamento humano.

Neste ponto você deve estar se perguntando para que toda esta discussão filosófica, certo? A questão é que uma boa definição de inteligência tornaria possível sua modelagem matemática, permitindo a sintetização do raciocínio.

Esta busca pelo Graal da computação, entretanto, revelou-se um beco sem saída. Abandonar esta aproximação filosófica, voltando a pesquisa no sentido de estudar os princípios e efeitos do raciocínio, foi a solução encontrada para permitir, finalmente, a reprodução sintética.

1.2 Fundamentos de IA

Ao escolhermos uma abordagem pragmática, embora não possamos definir diretamente inteligência, podemos observar suas características, investigando as regras lógicas do raciocínio. Se conhecermos tais regras poderemos, então, emula-las computacionalmente de forma a criar um mecanismo artificial de pensamento lógico. Isto nos devolve ao teste de Turing.

O teste, que citamos acima, compara uma máquina computacional com um humano. Na época de sua concepção, em virtude das limitações do entendimento das fronteiras de ação do computador, acabou criando uma falsa



associação do pensamento com a linguagem. Isto ocorreu porque, para passar no teste, a máquina precisaria “enganar” um humano. O que seria mais factível se ela, a máquina, fosse capaz de expressar-se em uma língua vernácula.

O pensador indiano Panini já havia, em 350 AC, dissociado o pensamento da linguagem, mas a confusão criada pelo teste de Turing, acabou direcionando os desenvolvimentos de IA para a interpretação e criação de textos por um bom tempo.

O estudo de IA, atualmente, entretanto, “abrange lógica, probabilidade e matemática do contínuo, além de percepção, raciocínio, aprendizado e ação” (NORVIG, 2013, pg.VI). Assim, quando enfrentamos um problema pela ótica da IA devemos esperar a presença rica de conceitos lógicos, matemáticos e estatísticos. Este fato justifica nosso foco inicial em lógica, matemática e probabilidade, nesta disciplina.

1.2 Quanto tempo IA possui?

O estudo da inteligência artificial tem seu início efetivo durante a Segunda Guerra Mundial. Naquele momento, criar uma máquina capaz de tratar um problema complexo, enfrentando-o pelo tempo necessário seria uma grande vantagem tática. A resposta a esse interesse foi dada pelo matemático inglês Alan Turing (ele novamente) que constrói uma máquina capaz de ajudar na decodificação das mensagens trocadas entre as tropas nazistas.

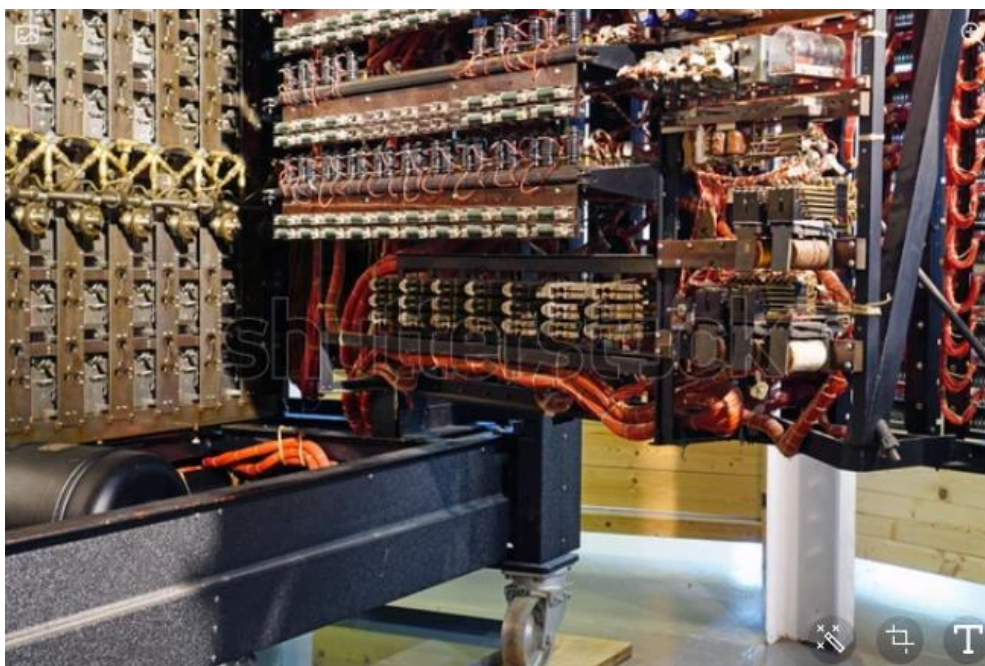


Figura 1 – Protótipo de Turing (<https://www.shutterstock.com/pt/image-photo/bletchley-united-kingdom-15-may-2016-1299695794>)

A ideia de IA, neste período, era sintetizar computacionalmente o neurônio humano. Este princípio era reforçado pelas pesquisas contemporâneas de Warren McCullock e Walter Pitts, que em 1943 propuseram a modelagem fisiológica e matemática do neurônio humano. O modelo matemático conhecido como neurônio de McCullock e Pitts foi tão bem aceito que, ainda hoje, é utilizado como base das redes neurais artificiais (RNAs).

A capacidade parca das primeiras máquinas computacionais, entretanto, impediu a aplicação prática do modelo na época, obrigando os cientistas computacionais ao abandono da modelagem neural como meio de implementação de IA.

Abandonada, temporariamente, esta aproximação neural, iniciaram-se pesquisas sobre o uso das soluções naturais evolutivas como modelos para o enfrentamento de problemas. Novamente a capacidade computacional reteve o desenvolvimento das pesquisas.

O primeiro sucesso em IA foi obtido em 1969, com uma terceira aproximação, diferente da neural e evolucionária, os sistemas especialistas. Estes sistemas codificavam, em linguagem computacional, o conhecimento de especialistas sobre o assunto, permitindo que a máquina assumisse o processo de decisão.



Somente no início do terceiro milênio as pesquisas de IA com uso de redes neurais encontraram aplicação prática. Em todos os métodos, até este ponto da história, as reações da máquina são completamente determinadas pelo seu criador, através do código ou do treinamento de um modelo, segundo o paradigma de Samuel estrito, o qual abordaremos em breve.

Mais recentemente surge o conceito de agente baseado em aprendizagem. Trata-se de dotar a máquina da capacidade de aprender durante a operação. Importante enfatizar que a ausência de aprendizagem durante a operação não descaracteriza o agente como inteligente (FACELI et al., 2021, pg.1).

1.3 O que é Machine Learning

Quando decidimos enfrentar um problema qualquer, com uso de uma estratégia computacional, precisamos analisar se este problema é solúvel. Para isso a mera complexidade não é fiel de balança. Existem problemas de altíssima complexidade, mas que podem ser particionados ou tratados, se aceita certa incerteza sobre a assertividade das soluções encontradas. Além desta limitação, para que um problema seja tratável, é preciso que as condições de contorno desta questão sejam computáveis, ou seja, estejam dentro da capacidade computacional disponível no momento em que se deseja codificar.

A limitação mais restritiva, porém, foi vaticinada por Samuel. Arthur Lee Samuel, cientista da computação americano, considerado por muitos o pai da IA moderna. Samuel enunciou o paradigma básico da computação segundo o qual um computador somente faz aquilo para o qual foi programado.

É possível, entretanto, sem romper com o paradigma de Samuel, permitir certa liberdade de evolução à máquina computacional. Esta liberdade é obtida pela criação de algoritmos capazes de analisar dados e tomar as decisões a partir desta análise e não diretamente do algoritmo. A estas técnicas chamamos de aprendizagem de máquina (ML - Machine Learning, em inglês).

Os algoritmos de ML surgiram como incrementos aos métodos clássicos de IA, mas hoje são consideradas um importante ramo da inteligência artificial.

TEMA 2: PRINCIPAIS LINHAS DE PESQUISA EM INTELIGÊNCIA ARTIFICIAL



Em nosso tema anterior comentamos, enquanto percorríamos a história da IA, a alternância sobre as abordagens na pesquisa de inteligência artificial. Ainda hoje as pesquisas em torno de IA não tem um rumo único. Há, naturalmente, abordagens mais concernentes com determinado tipo de problema que outras, permitindo a convivência harmoniosa e sinérgica das diversas linhas de pesquisa. A escolha da abordagem, entretanto, pode limitar as formas de aprendizagem possíveis, impedindo, por exemplo a reconfiguração autônoma do agente.

Vamos agora conhecer algumas destas abordagens.

2.1 IA Simbólica

Como já comentamos os esforços iniciais de reprodução parcial da mente humana foram frustrados pela da inexistência de capacidade computacional para processar os modelos neurais artificiais.

O modelo alternativo foi modelar matematicamente os problemas e não o processo mental genérico de solução de problemas. Surgiram, então, **algoritmos de solução de problemas dedicados, chamados de sistema baseados em conhecimento ou sistemas especialistas**.

Os sistemas especialistas são baseados em um “motor de inferência”, que **toma as decisões baseado em regras “se-então”**. Estes sistemas criam uma **árvore de decisão** com regras pré-definidas de navegação apreendidas diretamente de especialistas pelo arquiteto da solução computacional (FACELI et al., 2021).

Em sua forma inicial, tais sistemas eram estáticos com regras inseridas no momento da codificação. A agregação das atuais técnicas de ML, a estes sistemas, podem permitir certa evolução, ou mesmo permitir a definição autônoma da árvore de decisão. Vamos voltar a este assunto ainda nesta aula.

2.2 IA Conexionista

Conexionismo é o nome dado a primeira aproximação de IA, aquela de McCulloch e Pitts, na época de Turing, que baseia-se na tentativa de simulação computacional do processo biológico do raciocínio humano.



O termo conexionista surgiu do entendimento da inteligência pela óptica das conexões entre neurônios. Dito de outra forma, as conexões ou sinapses neurais, como são chamadas, são responsáveis pela memória e pela capacidade de interpretação racional. Desta conclusão surgem os modelos clássicos de redes neurais que aprendem pela alteração dos pesos sinápticos (ditos $w_{i,j}$) conforme ilustra a rede elementar de neurônios abaixo.

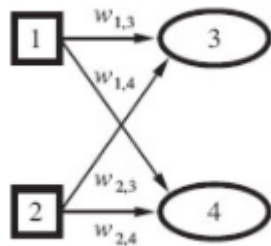


Figura 2 – Rede neural artificial e seus respectivos pesos sinápticos. (adaptado de Norvig, 2013, pg 623)

Nesta ilustração temos quatro neurônios que seguem o modelo de McCulloch e Pitts, interconectados por sinapses. A intensidade da sinapse é modelada pelos pesos sinápticos $w_{i,j}$. Parece confuso? Fique tranquilo que voltaremos a este tema “nervoso” em breve, ainda nesta aula

2.3 IA Evolucionária

Uma terceira aproximação para a criação de algoritmos de IA é a técnica evolucionária. Esta aproximação baseia-se na ideia de que tanto a mente humana como a natureza são resultados de um processo evolutivo. O processo tem um componente aleatório forte, associado à seleção natural. As aplicações práticas desta aproximação se dão principalmente pelos algoritmos genéticos (AGs).

AGs são algoritmos de busca que procuram uma solução mais eficiente em um conjunto de soluções possíveis. Desta forma, tornam-se uma alternativa para o enfrentamento de problemas intratáveis. AGs contornam o problema de convívio com matemáticas não computáveis diretamente.

Escolha da roteirização de games e do planejamento da arquitetura de circuitos eletrônicos são aplicações práticas destas técnicas (Norvig, 2013).



Os AGs são, portanto, algoritmos estocásticos direcionando a busca de uma solução em regiões nas quais será mais provável encontrar uma solução ótima. AZEVEDO (2000, p. 37) afirma que tais algoritmos fazem uso de operadores probabilísticos e não de regras determinísticas, realizando uma busca exploratória, mas estruturada com base nas informações de “fitness” (ou adequação) das gerações de soluções anteriores.

Um exemplo de aplicação de algoritmo genético em C++ pode ser encontrado em <https://galib-c-plus-plus-genetic-algorithms-library.soft112.com/>.

2.4 IA Difusa

O tratamento de problemas de alta complexidade, mesmo daqueles inicialmente intratáveis, podem ser enfrentados se aceitarmos certa incerteza nos resultados. Weaver (1991) afirma que **complexidade e incerteza podem ser encarados como dimensões de um problema**. Se, em dado evento, tanto incerteza quanto complexidade forem autos a lógica determinística não será capaz de compreendê-lo. Neste caso, para permitir o entendimento, devemos aceitar a incerteza como componente natural do evento.

Yager & Jadeh (2012) afirmam que o raciocínio que pode conduzir à solução de um problema complexo, não é objetivo, mas, difuso. **Aplicações do raciocínio difuso são a lógica Fuzzi, que reputa a cada informação um grau de incerteza e a computação quântica**. Nesta última aplicação, a computação empresta os conceitos estocásticos da mecânica quântica. A informação só é objetivada quando a observamos. Antes deste fato, não temos certeza sobre ela. Se você ficou curioso sobre computação quântica, a leitura da obra de FALBRIARD & BROSSO (2022) será bastante interessante para você.

2.5 Comparação linhas de IA

Tomando por base o plano de Weaver podemos localizar a aplicabilidade de cada linha de IA, conforme a complexidade e incerteza do problema crescem. O gráfico abaixo ilustra esta relação.

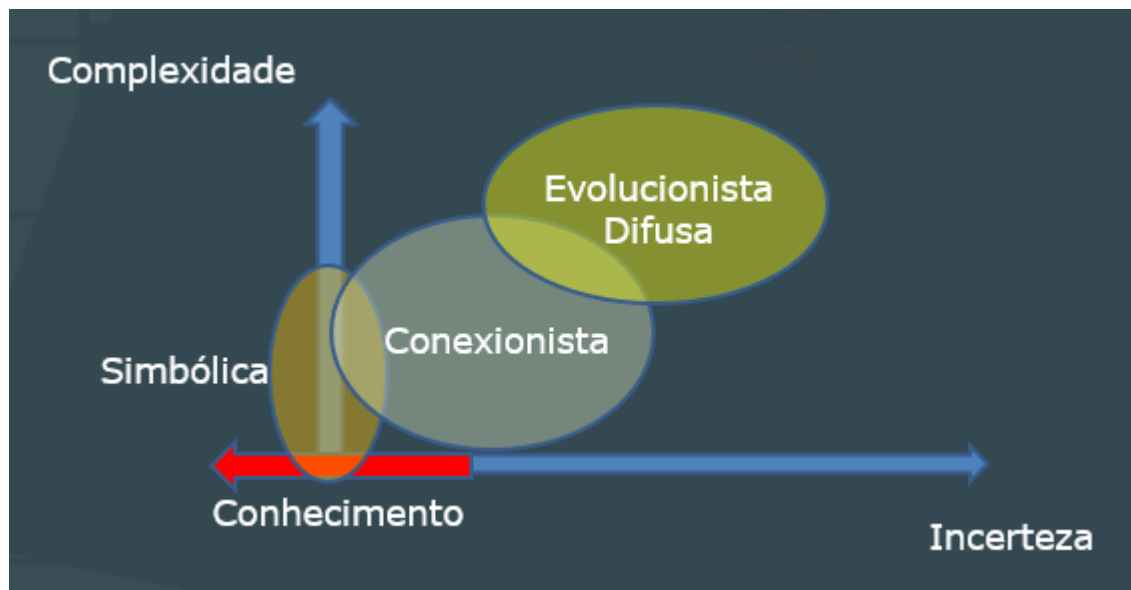


Figura 3: Plano de Weaver e as linhas de IA (autoral)

Observe que os sistemas especialistas estão próximos a origem do gráfico, porque demandam conhecimento prévio das soluções ao passo que, conforme a complexidade e incerteza crescem, tendemos ao uso de raciocínios estocásticos.

TEMA 3 – IA SIMBÓLICA E HEURÍSTICA

Como já introduzimos anteriormente, a IA simbólica (aquela dos sistemas especialistas, lembra?) busca codificar a modelagem matemática do problema. Vamos entender um pouco melhor esta ideia.

3.1 Princípios de IA Simbólica

Os sistemas especialistas são baseados em um mecanismo de inferência ou de raciocínio. Este mecanismo pode possuir as informações necessárias a tomada de decisão ou utilizar um banco de dados associado, que as possua. Neste último caso, a máquina de inferência apenas possuirá a lógica de acesso às informações, presentes no BD. De qualquer forma, tanto a máquina de inferência quanto o eventual banco de dados, estão presentes desde a concepção do sistema.



Um sistema simbólico pode ter interações com o meio. Esta interação pode ser feita por meio de sensores, para adquirir dados e atuadores, que permitirão a ação do sistema sobre o meio.

A somatória de mecanismo de inferência, dados, sensores e atuadores compõe o **agente inteligente simbólico**. O agente, cujo modelo é descrito na figura abaixo, coleta informações e age sobre o meio, seguindo estritamente o paradigma de Samuel, ou seja, não há adaptação, todas as ações são pré definidas.

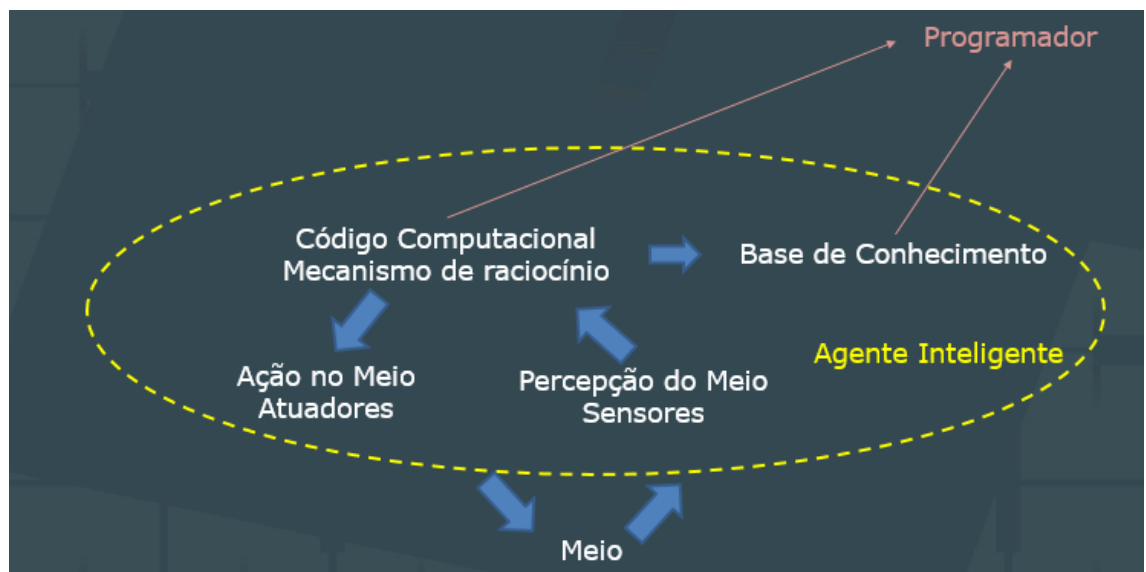


Figura 4: Sistema especialista (autoral)

O desenvolvimento de sistemas especialistas demandou a criação de um tipo de linguagem que oferece maior facilidade na criação da árvore de decisão típica destes sistemas. Desta forma, surge o paradigma de programação declarativo, antagonizando com o paradigma clássico imperativo de programação, ao qual se associam linguagens tradicionais como Java, C, C++, Javascript, Python, etc.



Figura 5: Paradigmas de programação (autoral)

O abandono gradativo dos sistemas especialistas fez com que linguagens declarativas, como Prolog, caíssem em também em desuso.

Alguns conceitos criados pelos sistemas de IA, da época áurea do simbolismo, ainda perseveram, principalmente na implementação de técnicas recentes de aprendizagem por reforço (Reinforcement Learning – RL) aplicáveis em robótica e algoritmos de jogos, por este motivo, vamos explora-los brevemente.

Para ilustrar nossa exposição, vamos tomar um exemplo clássico de IA, o quebra-cabeça de blocos deslizantes. Imagine uma caixa com cubos numerados, como representado abaixo, que lhe são entregues em uma ordem aleatória e precisam ser ordenados. Os blocos não podem ser removidos da caixa para que sejam ordenados, só é possível desliza-los para os lados, respeitando as paredes da caixa.



Figura 6: Quebra-cabeça de blocos deslizantes (autoral)

Chamamos de **abstrações** as regras que serão utilizadas para codificar o problema, são exemplos de abstrações:

- . a existência de 8 blocos



- . a existência de uma matriz de deslocamento 3X3
- . os movimentos permitidos
- . as limitações de movimentos pelos índices da matriz

Chamamos de **estado** cada disposição possível dos blocos. O **estado inicial** será a disposição dos blocos, quando recebemos o quebra-cabeças, ao passo que, **estado final**, será a disposição que desejamos obter.

O **custo do caminho**, entre dois estados quaisquer, será o número de movimentos realizado entre estes estados.

Toda vez que realizamos um movimento, precisamos compara-lo com o estado final para que saibamos se o objetivo (estado final) foi obtido, a esta comparação, chamamos de teste de objetivo.

3.2 Árvores de Busca

Já comentamos, que o motor de inferência de um sistema especialista cria uma árvore de decisão, para buscar a solução do problema. Vamos então exemplificar, para nosso quebra-cabeças, a árvore de busca criada, ilustrada parcialmente na figura abaixo.

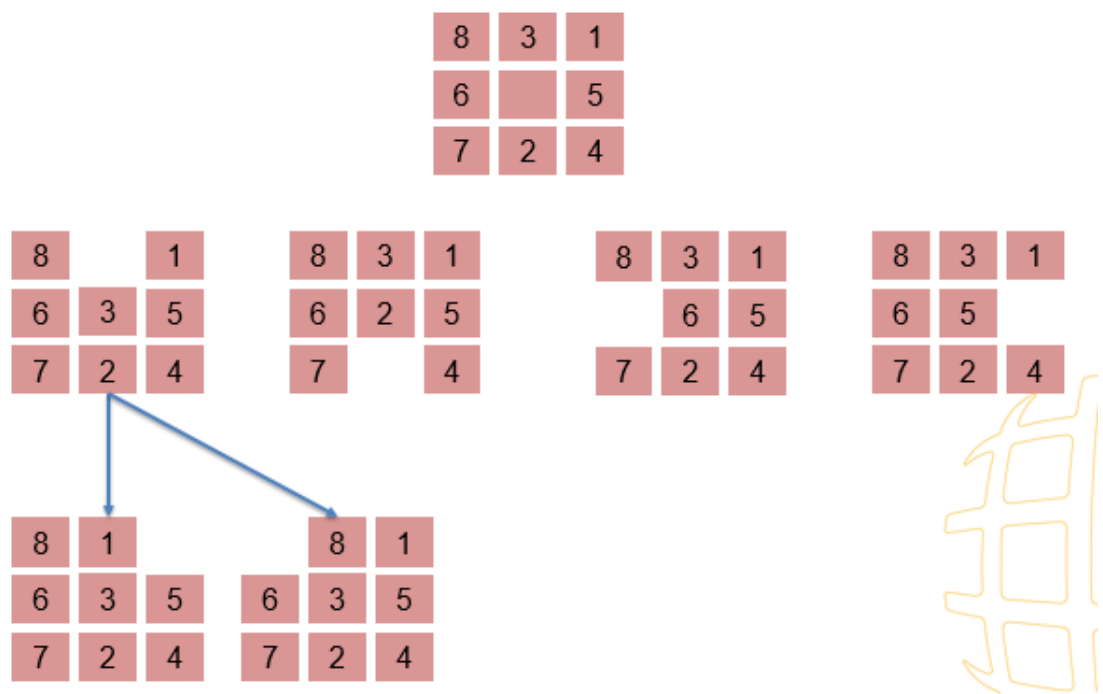


Figura 7: Arvore de busca parcial do quebra-cabeças de blocos deslizantes (autoral)



Nesta árvore o **nó raiz** corresponde ao estado inicial e os quatro ramos abaixo dele, representam todas os estados possíveis com custo de caminho unitário.

A árvore, da figura 7, está, obviamente incompleta, podemos seguir sua construção em várias sequências possíveis. Em largura, expandindo, a cada passo, todos os estados com custo unitário em todos os ramos, mas também podemos escolher uma estratégia de busca em profundidade, expandindo um único ramo até que se chegue ao objetivo ou a um estado já expandido anteriormente.

Estas estratégias de busca são ditas **buscas cegas**, por não utilizarem qualquer informação para orientar a expansão dos nós. O método alternativo será dito **busca informada ou busca heurística**.

3.3 Heurística

A **heurística é um algoritmo que permite avaliar se, ao expandirmos um nó, o fazemos no sentido da aproximação do objetivo (estado final)** ou se estamos nos afastando dele. A implementação de uma heurística não garante a maior eficiência de processamento, diretamente, mas coíbe extensas buscas em espaços que não conduzem ao objetivo. Desta forma há um ganho médio, quando realizamos diversas buscas, efetivo.

A grande questão é como encontrar uma heurística para o seu problema. De fato, fora do âmbito estrito de IA, heurística é definida como uma estratégia de decisão que propositalmente ignora parte de informação de forma a tornar possível a decisão de forma rápida. Prieditis et al. (1993) provaram que é possível obter uma heurística pelo relaxamento do problema, ou seja, **pela eliminação de alguma abstração**. Interessante não? Se você “relaxar” o problema uma heurística será criada. Vamos entender melhor.

Suponha que, no exemplo do quebra-cabeças, você elimine as abstrações que regem os movimentos, mas mantem as demais abstrações, como a necessidade de mover um bloco de cada vez e apenas uma casa por vez.

Feito isso, poder-se-ia mover as peças livremente até seu objetivo. Se durante este processo, contarmos o número de movimentos necessários, a partir de um nó, até se chegar ao objetivo, teremos uma heurística (chamada distância



de Manhattan) que servirá para estimar o custo total de uma busca em profundidade a partir daquele nó. Tente fazer este exercício e verifique como esta heurística é efetiva como indicadora do caminho a seguir.

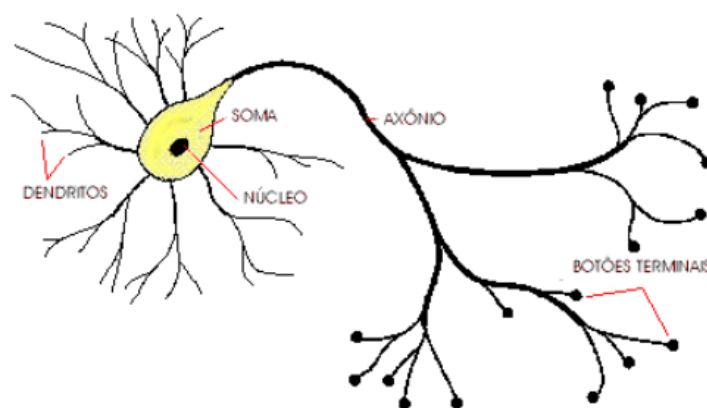
TEMA 4 – IA CONEXIONISTA E O TREINAMENTO NEURAL

Um agente inteligente, para que possa ser assim chamado, precisa adquirir conhecimento, este pode ser inserido diretamente pelo criador ou por um processo de aprendizagem. A IA, pela aproximação conexionista, tem por estrutura (equivalente ao motor de indução, que vimos no tema anterior) uma rede de neurônios artificiais. O conhecimento será adquirido pelo treinamento destes neurônios.

4.1 Neurônio e Perceptron

A unidade básica de uma RNA é o neurônio. O neurônio de McCulloch e Pits, sobre o qual já comentamos, é uma síntese matemática de um neurônio humano e sobre este modelo elementar as redes neurais artificiais serão construídas.

O neurônio biológico representado abaixo possui uma região proximal ao núcleo composta por dendritos, filamentos celulares que operam como sensores.



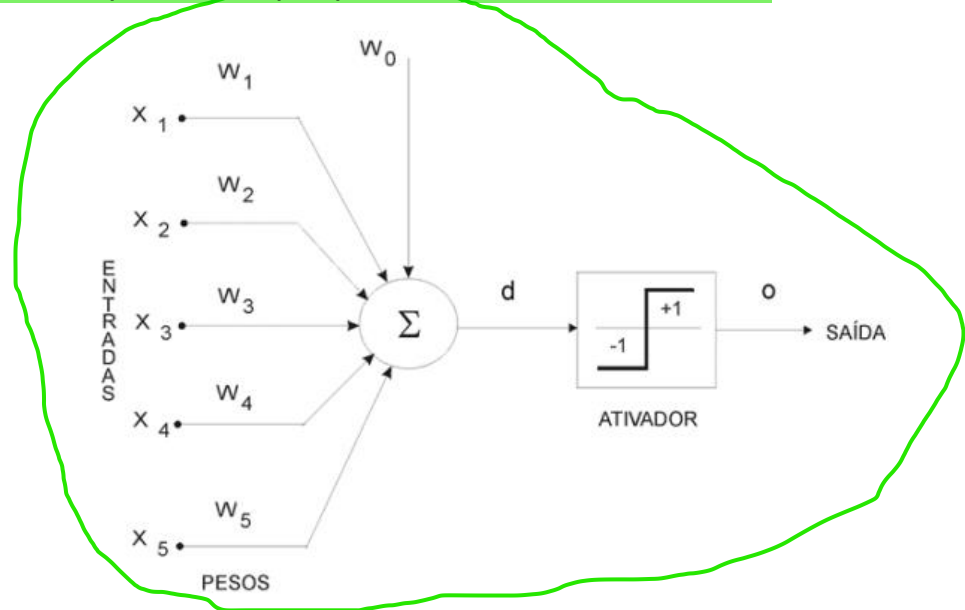
Fonte: adaptado de Medeiros (2007).

Na região distal ao núcleo, outros filamentos formam a rede axônica que transmite a “computação” realizada no núcleo a partir dos sinais captados pelos dendritos.



Computada a informação dos filamentos receptores, pelo núcleo, ela será transmitida pela rede axônica e captada pelos dendritos de neurônios próximos, pela conexão sináptica ou sinapse entre eles. A intensidade do estímulo captado pelo núcleo depende da proximidade entre axônio e dendrito.

Para reproduzir este comportamento McCulloch e Pits propuseram um modelo matemático (figura abaixo), atualmente conhecido por **Perceptron**, que transforma o núcleo em um somador acrescido de um operador matemático, dito **função de ativação**. A modelagem da intensidade das sinapses é feita pela inserção de um peso sináptico “w” o qual pondera a soma dos estímulos.



Fonte: adaptado de Medeiros (2007).

Na versão original, proposta pelos neurocientistas, a função de ativação seria basicamente uma “degrau unitário” que **externaria o valor +1, como saída do neurônio, se e somente se a somatória ponderal das entradas fosse igual ou maior que zero. Caso a somatória resultasse em um valor negativo, a saída seria -1 (ou zero dependendo da implementação).**

Atualmente as funções de ativação não se limitam unicamente a degrau, há neurônios que utilizam funções não lineares e lineares deslocadas da origem como a ReLu.

Sobre este tema sugerimos o aprofundamento pela leitura dos primeiros capítulos da obra de Norvig (2013) referenciada ao final deste capítulo.

4.2 RNA



O cérebro humano é formado por alguns milhões de neurônios conectados por trilhões de sinapses. A IA conexionista, buscando reproduzir o comportamento inteligente, associou os Perceptrons em redes neurais artificiais (RNAs) conforme ilustrado abaixo.

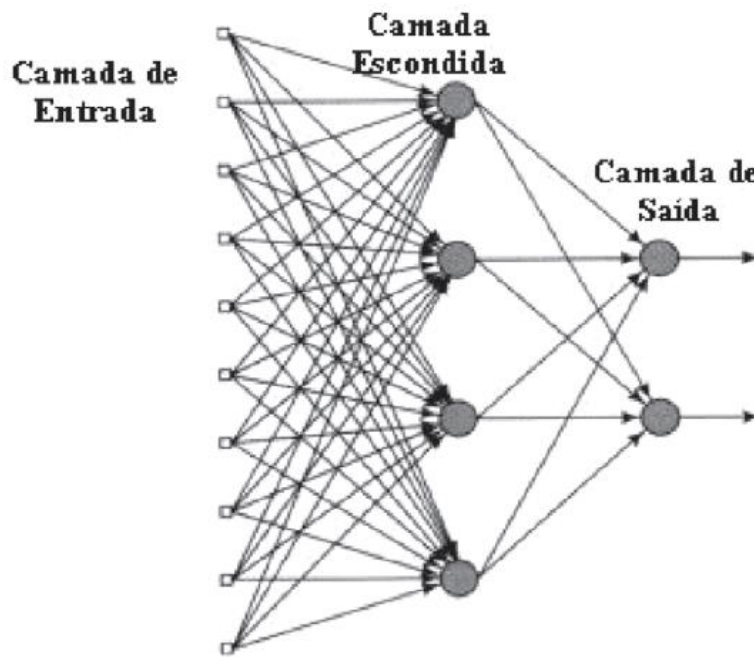


Figura 10: RNA (SANTOS et al., 2005)

Observe, na figura, que os neurônios da camada de entrada estão conectados aos neurônios da camada dita escondida, ou oculta. Os neurônios ocultos, por sua vez, se conectam com os dois da camada de saída, cada uma destas conexões está associada a um peso sináptico que simula a proximidade biológica, antes comentada.

O aprendizado de uma RNA é obtido pela seleção correta destes pesos, ou seja, por um algoritmo que ajuste os pesos para que a saída da rede seja aquela que desejamos. O peso sináptico é, por este motivo dito parâmetro livre da RNA, seu ajuste permite “intensificar” ou “deprimir” a intensidade da conexão entre dois neurônios.

A este processo de ajuste denominamos treinamento da rede neural. Após o treinamento, a RNA estará pronta para operar e enfrentará, neste processo, situações inusitadas. Assim como a rede biológica, a RNA será capaz



de generalizar o conhecimento aprendido, tomando uma decisão provavelmente acertada.

4.3 Treinamento Neural

Podemos imaginar que uma “boa” rede neural será composta por uma enorme quantidade de neurônios, mas Norvig (2013, pg. 629) afirma que “se escolhermos uma rede muito grande, ela será capaz de memorizar todos os exemplos, formando uma tabela grande de pesquisa, mas não generalizará bem, necessariamente, as entradas que nunca foram vistas antes”. Assim devemos, neste ponto, abandonar a ideia intuitiva de redes imensas, mesmo porque o custo computacional para treina-las as converterá em um problema intratável computacionalmente.

Há basicamente dois tipos de treinamento. O primeiro é implementado quando sabemos, para uma coleção determinada de eventos (entradas), a saída numérica que desejamos. A este processo denominamos treinamento supervisionado. Se desconhecemos a saída o treinamento é dito não supervisionado.

O **treinamento supervisionado** talvez seja o método mais intuitivo de aprendizagem. Neste modelo devemos selecionar um conjunto (x) de dados de entrada (amostras), cuja saída desejada (d) da RNA, conheçamos previamente. O processo se inicia pela inserção dos dados de entrada na RNA e a sequente coleta da saída (y) obtida. Como (d) é conhecida podemos determinar se ocorreu ou não erro (e). O valor do erro para a n -ésima amostra será:

$$e(n) = d(n) - y(n)$$

Os pesos sinápticos, então poderão ser corrigidos em função do erro, como ilustrado abaixo.

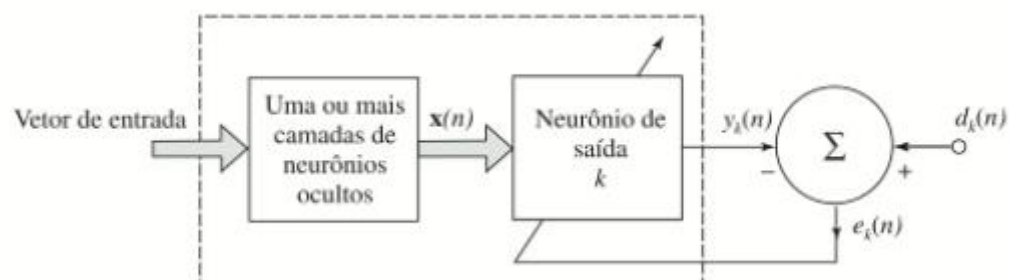




Figura 11 – Treinamento por correção de erro em RNAs (Haykin (2011, pg. 77)

Uma técnica bastante usual para a alteração dos parâmetros livres em função do erro da rede, é a regra de Widrow-Hoff (ou delta). Neste algoritmo, selecionamos uma velocidade (ou taxa) de aprendizagem e corrigimos os pesos por um gradiente, composto pelo produto do erro, taxa de aprendizagem e vetor de entrada.

Este é um processo iterativo e a cada passagem do conjunto de amostras pela RNA (denominado época de treinamento) o valor do erro total da rede será reduzido.

Há ainda o treinamento não supervisionado, neste caso não é possível conhecer a saída da RNA. Talvez você esteja se perguntando sobre a serventia deste treinamento.

De fato, boa parte do conhecimento adquirido por você, em sua vida, é obtido por treinamento não supervisionado.

Neste caso o treinamento se baseia sobre a natural classificação dos eventos. Quando observamos um conjunto de eventos, naturalmente os classificamos em categorias, ou seja, a natureza auto-organiza os eventos (Di Biasi, 2014).

As RNAs, por modelarem o comportamento natural biológico cerebral, importam também essa capacidade classificatória.

Dito de outra forma, uma RNA, por treinamento não supervisionado, será capaz de classificar as entradas mesmo que nenhum dado delas se conheça previamente.

TEMA 5 – ALGUMAS RNA COMERCIAIS DISPONÍVEIS

Estão disponíveis na internet algumas aplicações que permitem o uso de redes neurais artificiais com variados graus de dificuldade. Estão disponíveis redes gratuitas e remuneradas, vamos agora conhecer algumas.

5.1 RNA Azure



A Azure é a plataforma de nuvem da Microsoft. Disponibiliza servidores e serviços de nuvem. Entre estes serviços estão aqueles relacionados a IA.

A MS disponibiliza também cursos de certificação, voltados ao uso de IA, com o uso da plataforma Azure como serviço (PaaS), a exemplo da certificação AI-100 – Designing and Implementing an Azure AI Solution.

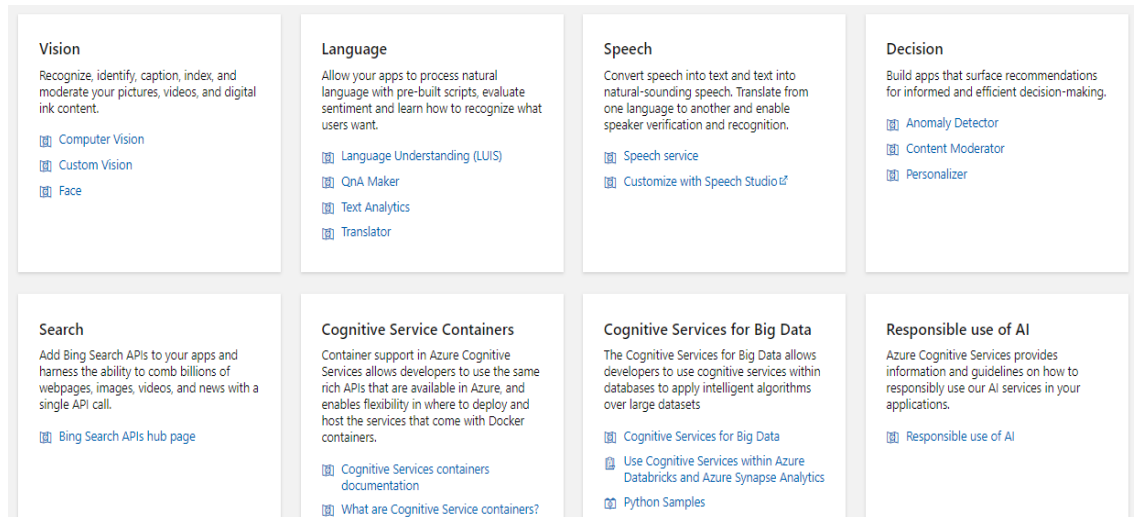


Figura 12: Serviços de IA no modelo PaaS da MS Azure.

Na plataforma há também datasets populados que permitem o teste dos serviços de IA como visão computacional, reconhecimento de fala e serviços cognitivos para big data.

5.2 Tensor Flow Google

O Tensor Flow é de fato uma biblioteca com recursos de IA conexcionista, criada pela Google, que pode ser utilizada livremente.

Esta biblioteca se tornou popular em função dos bons resultados obtidos em sua combinação com a linguagem Python, já famosa pelas facilidades ligadas a IA. Há ainda, no pacote Google, uma biblioteca que facilita o uso do Tensor Flow, chamada de Keras. A extrema facilidade para criação e treinamento de uma RNA elementar pode ser vista no trecho de código abaixo.



```
model = keras.Sequential([keras.layers.Dense(units=1, input_shape=[1])])
model.compile(optimizer='sgd', loss='mean_squared_error')

xs = np.array([-1.0, 0.0, 1.0, 2.0, 3.0, 4.0], dtype=float)
ys = np.array([-3.0, -1.0, 1.0, 3.0, 5.0, 7.0], dtype=float)

model.fit(xs, ys, epochs=500)

print(model.predict([10.0]))
```

Figura 13: : Google ML de zero a 100 em
<https://www.youtube.com/watch?v=t5z5lyrb-7s>

5.3 Watson IBM

A IA Watson, criada pela IBM, é comercializada associada a uma consultoria de aplicação. Trata-se de uma plataforma bastante sofisticada com foco em soluções para grandes empresas. Há documentação interessante, disponível no site da empresa, sobre a tecnologia empregada no desenvolvimento dos algoritmos.

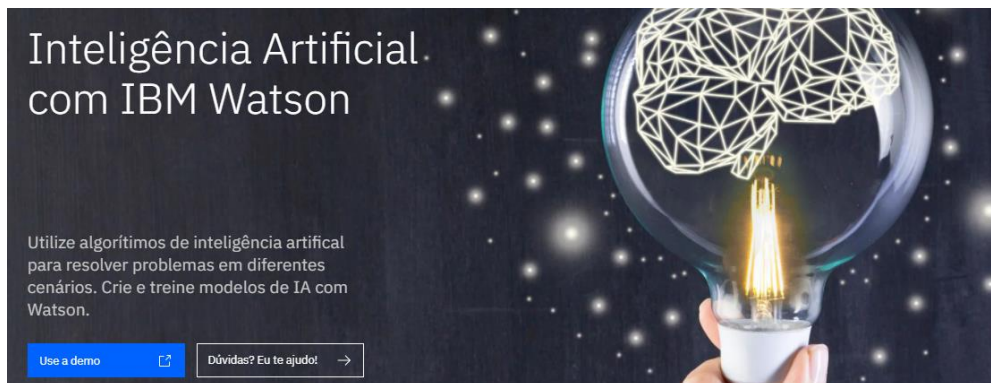


Figura 14: : IBM Watson em <https://www.ibm.com/>



FINALIZANDO

Neste capítulo buscamos construir conceitos básicos de IA, das abordagens e ferramentas existentes, descrevendo algumas das técnicas possíveis de implementação de processos de treinamento, para cada linha de IA, bem como algumas RNAs e plataformas disponíveis na Internet. São apenas vislumbres da história e das possibilidades que IA nos apresenta, na intenção de despertar o interesse pelo estudo desta empolgante área do conhecimento.

REFERÊNCIAS

AZEVEDO, F. M.; BRASIL, L. M.; OLIVEIRA, R. C. L. Redes Neurais com Aplicações em Controle e em Sistemas Especialistas. Florianópolis: Visualbooks, 2000.

DAMÁSIO, António. **O erro de Descartes: emoção, razão e o cérebro humano**. Editora Companhia das Letras, 2012.

DI BIASE, Francisco. Sistemas auto-organizadores físicos, biológicos, sociais e empresariais. **International Journal of Knowledge Engineering and Management (IJKEM)**, v. 2, n. 2, p. 123-146, 2013.

FALBRIARD, C.; BROSSO, I. Computacao Quantica. [Digite o Local da Editora]: Editora Alta Books, 2020. 9786555201529. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9786555201529/>. MB

FACELI, Katti, et al. Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina. Disponível em: Minha Biblioteca, (2nd edição). Grupo GEN, 2021. MB

HAYKIN, Simon. Redes Neurais. [Digite o Local da Editora]: Grupo A, 2011. 9788577800865. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577800865/>. Acesso em: 21 mar. 2022.

PRIEDITIS, Armand; JANAKIRAMAN, Bhaskar; JANAKIRAMAN, B. Generating Effective Admissible Heuristics by Abstraction and Reconstitution. In: **AAAI**. 1993. p. 743-748.

NORVIG, Peter. Inteligência Artificial. Disponível em: Minha Biblioteca, (3rd edição). Grupo GEN, 2013. MB



YAGER, Ronald R.; ZADEH, Lotfi A. (Ed.). An introduction to fuzzy logic applications in intelligent systems. Springer Science & Business Media, 2012.

WEAVER, Warren. Science and complexity. In: Facets of systems science. Springer, Boston, MA, 1991. p. 449-456.

MEDEIROS, L. F. Redes Neurais em Delphi. 2. ed. Florianópolis: Visualbooks, 2007.

SANTOS, Alcione Miranda dos et al. Usando redes neurais artificiais e regressão logística na predição da hepatite A. Revista Brasileira de Epidemiologia, v. 8, p. 117-126, 2005.