
MINISTÉRIO DA EDUCAÇÃO – MEC

SECRETARIA DE EDUCAÇÃO SUPERIOR – SESU

PROGRAMA DE EDUCAÇÃO TUTORIAL – PET

UNIVERSIDADE FEDERAL FLUMINENSE – UFF

ESCOLA DE ENGENHARIA – TCE

GRUPO PET DO CURSO DE ENG. DE TELECOMUNICAÇÕES – PET-TELE

Tutoriais PET-Tele

Um breve resumo dos estudos relativos a Inteligências Artificiais (Versão: A2023M01D27)

Autor: Amanda Souza Zírpolo (dez/2022)
Lucca Sabbatini Reid Rodrigues (abr/2022)

Tutor: Alexandre Santos de la Vega

Niterói – RJ
Janeiro / 2023

Sumário

| | | |
|----------|--|-----------|
| 1 | Introdução | 3 |
| 1.1 | Programa PET | 3 |
| 1.2 | Motivações | 3 |
| 1.3 | Objetivo | 3 |
| 1.4 | Resultados esperados | 3 |
| 2 | Inteligências Artificiais: conceitos básicos | 4 |
| 2.1 | Introdução | 4 |
| 2.2 | O conceito de Inteligência | 4 |
| 2.3 | O conceito de Inteligência Artificial | 4 |
| 2.4 | Interdisciplinaridade das Inteligências Artificiais | 5 |
| 3 | Inteligências Artificiais: classificações | 6 |
| 3.1 | Classificação quanto ao tipo de aprendizado das IAs | 6 |
| 3.2 | Classificação quanto à descrição das IAs | 7 |
| 4 | Aprendizado de máquinas | 8 |
| 4.1 | O que é o <i>machine learning</i> | 8 |
| 4.2 | Como funciona o <i>machine learning</i> | 8 |
| 4.3 | Qual a importância do <i>machine learning</i> | 9 |
| 5 | Algoritmos para aprendizado de máquinas | 10 |
| 5.1 | Aprendizado supervisionado | 10 |
| 5.1.1 | Algoritmos de classificação | 10 |
| 5.1.2 | Algoritmos de regressão | 11 |
| 5.2 | Aprendizado não supervisionado | 11 |
| 5.2.1 | Regras de associação | 11 |
| 5.2.2 | Redução de dimensionalidade | 12 |
| 5.3 | Aprendizado semi-supervisionado | 13 |
| 5.4 | Aprendizado por reforço | 13 |
| 6 | Python e <i>machine learning</i> | 14 |
| 6.1 | Visão geral da linguagem | 14 |
| 6.2 | Uso de <i>Python</i> em <i>machine learning</i> | 14 |
| 6.3 | Bibliotecas <i>Python</i> para <i>machine learning</i> | 15 |
| 6.3.1 | <i>Numpy</i> | 15 |
| 6.3.2 | <i>SciPy</i> | 15 |
| 6.3.3 | <i>Scikit-learn</i> | 15 |
| 6.3.4 | <i>Theano</i> | 15 |
| 6.3.5 | <i>TensorFlow</i> | 15 |

| | | |
|----------|-----------------------------------|-----------|
| 6.3.6 | <i>Keras</i> | 15 |
| 6.3.7 | <i>PyTorch</i> | 16 |
| 6.3.8 | <i>Pandas</i> | 16 |
| 6.3.9 | <i>Matplotlib</i> | 16 |
| 7 | Conclusão | 17 |
| | Referências bibliográficas | 18 |

Capítulo 1

Introdução

1.1 Programa PET

O Programa de Educação Tutorial (PET) [Pro], do Ministério da Educação (MEC), exige que os grupos PET desenvolvam atividades que contemplem, de forma indissociável, itens de Pesquisa, de Ensino e de Extensão. Além disso, os grupos devem estimular uma evolução positiva dos seus integrantes, dos demais alunos do seu curso de graduação, do próprio curso e da sua instituição.

Nesse sentido, o PET-Tele [PET] procura desenvolver atividades e/ou atender a demandas que cumpram tais exigências.

A seguir, são apresentadas as motivações, o objetivo e os resultados esperados para o trabalho em questão.

1.2 Motivações

No final da década de 2010, o grupo estudou a versão corrente da linguagem, produziu uma apostila e ofereceu um curso introdução a Python. Em seguida, os cursos de Computação e de Engenharia passaram a adotar a linguagem na primeira disciplina de ensino de programação, fazendo com que o grupo desviasse a sua atenção da linguagem.

Atualmente, devido à aplicação de Python em diversas áreas de interesse, tanto da Academia quanto do Mercado Corporativo, tais como: cálculo numérico, cálculo matricial, cálculos estatísticos, inteligência artificial, aprendizado de máquina e *big data*, o grupo visualizou uma forma de inovar, amadurecer e expandir seus conhecimentos relativos à linguagem.

1.3 Objetivo

O objetivo deste documento é organizar e disponibilizar um breve resumo dos estudos relativos a Inteligências Artificiais, ao Aprendizado de Máquina e ao uso da linguagem Python nessas áreas.

1.4 Resultados esperados

Ao final da leitura deste documento, é esperado que o leitor tenha adquirido conceitos básicos sobre Inteligências Artificiais e Aprendizado de Máquina, bem como noções de como utilizar a linguagem Python para programar nestas áreas.

Capítulo 2

Inteligências Artificiais: conceitos básicos

O objetivo desse capítulo é apresentar algumas ideias básicas sobre o conceito de inteligência, seja ela de origem natural ou artificial.

2.1 Introdução

Antes de começar a estudar e a entender o que é Inteligência Artificial, é necessário primeiro definir o conceito de inteligência. Tal conceito é essencial para o entendimento de toda teoria que existe por trás do estudo presente neste documento. Nesse sentido, naturalmente surgem algumas questões, tais como: O que é inteligência? O que diferencia inteligência natural e inteligência artificial? Quais são os requisitos para que se tenha inteligência? É possível replicar a mente humana de maneira artificial? Algumas destas questões são discutidas a seguir.

2.2 O conceito de Inteligência

De acordo com o dicionário de Oxford [Oxf], inteligência é:

1. *faculdade de conhecer, compreender e aprender.*
2. *capacidade de compreender e resolver novos problemas e conflitos e de adaptar-se a novas situações.*

A partir disso, é possível ver inteligência como a capacidade de evoluir e fazer escolhas diferentes, baseando-se em experiências passadas e no ambiente ao seu redor.

Contudo, essa é apenas uma das definições existentes. A inteligência também pode ser entendida como a capacidade de controle das atividades do corpo. Tal forma de inteligência é denominada inteligência inata. É ela que governa o crescimento do corpo humano ou o crescimento dos galhos e do tronco de uma árvore.

Isso já é o bastante para deixar claro que a inteligência não é um conceito exclusivo dos humanos, e pode ser explorada e replicada, pelo menos em alguns níveis, artificialmente.

2.3 O conceito de Inteligência Artificial

Antes de falar de Inteligência Artificial (IA), é necessário refletir sobre a inteligência de um computador. Computadores possuem inteligência? Se sim, de onde vem essa inteligência? Quão poderosa ela é?

Computadores são bastante diferentes de seres vivos portadores de inteligências inatas e naturais. Embora possuam inteligência, pode-se dizer que os computadores só possuem inteligência inata. E a origem dessa inteligência está no ser humano. Computadores e, consequentemente, suas inteligências, só são capazes de conhecer, compreender, aprender, analisar e se adaptar, baseando-se no que o ser humano ensina a ele. Um computador que foi programado e ensinado a automatizar um processo industrial, não faz ideia de como extrair emoções de um texto, e vice-versa.

Dito isso, uma das definições mais interessantes de IA é a de Inteligência Aumentada. Isso significa que uma IA pode ser vista como uma extensão da inteligência de um ser humano.

Do mesmo jeito que a *internet* e a *web* hoje servem como uma extensão da memória humana, quando queremos saber qual é a população de um país ou qual a data de algum evento histórico, as IAs servem para ajudar no processo de realização de tarefas que poderiam ser feitas por humanos, mas demandariam muito mais esforço e tempo.

Assim, do mesmo jeito que um humano pode ler e reler enciclopédias até reter uma enorme quantidade de informação, que talvez nunca seja utilizada e que pode se tornar desatualizada rapidamente, um ser humano também pode sentar em uma mesa, classificar e separar quantidades astronômicas de dados em um conjunto finito de categorias. Contudo, isso não é nem um pouco eficiente, pois é muito mais rápido pesquisar na *web* e afirmar com propriedade que, durante a escrita deste texto, o Brasil possuía 212.6 milhões de habitantes [THE], assim como é também muito mais eficiente reunir um conjunto de dados, um algoritmo de aprendizado de máquina e ensinar um computador a classificar qualquer quantidade de novos dados no lugar de uma pessoa.

As IAs proporcionam a possibilidade de terceirizar trabalhos árduos e massantes para um computador, de maneira confiável, monitorável, ajustável e com evolução automática, redirecionando o tempo que as pessoas gastariam em tais trabalhos para outras atividades, mais importantes e valorosas, como, por exemplo, pensar.

2.4 Interdisciplinaridade das Inteligências Artificiais

Mesmo sendo um assunto que é automaticamente relacionado com as áreas tecnológicas em geral e com a computação, a Inteligência Artificial é uma área extremamente interdisciplinar. Isso pode ser visto na seguinte lista, que apresenta algumas das principais áreas relacionadas aos estudos sobre IAs e seus respectivos papéis:

- **Ciência da Computação:** implementação das IAs no nível de *software*.
- **Engenharia Elétrica:** implementação das IAs no nível de *hardware*.
- **Matemática e Estatística:** desenvolvimento de métricas para análise de viabilidade de modelos.
- **Psicologia e Linguística:** entendimento do funcionamento da mente humana, que serve de base para o entendimento e o desenvolvimento de IAs.
- **Filosofia:** discussão de questões e de considerações éticas e morais, a serem empregadas na implementação de IAs.

Assim como no seu estudo, que envolve diversas áreas de conhecimento, as IAs podem ser aplicadas em diversos contextos. Elas podem servir de ajuda para qualquer área que possua atividades envolvidas com o reconhecimento e a diferenciação de informações. Pode-se dizer que sua aplicabilidade é quase ilimitada.

Capítulo 3

Inteligências Artificiais: classificações

Após uma breve introdução, para a familiarização dos conceitos que se encontram por trás das Inteligências Artificiais, é possível começar a classificar e diferenciar os tipos de IAs. Esse é o intuito deste capítulo.

3.1 Classificação quanto ao tipo de aprendizado das IAs

Uma das primeiras maneiras de se diferenciar e classificar uma IA é por meio do tipo de treinamento que gerou o modelo em questão.

O treinamento de uma IA pode ser classificado da seguinte forma:

1. **Aprendizado supervisionado:** realizado por meio da comparação da entrada e da saída de dados de um sistema, com valores esperados, por meio de exemplos. O programador humano insere as informações, ensinando à máquina o que é desejável e o que não é. Assim, o sistema aprende a realizar as comparações baseadas nos exemplos que lhe foram dados.
2. **Aprendizado não supervisionado:** a máquina opera por si só, sem nenhuma informação acerca do modelo de resposta desejado pelo programador. O aprendizado consiste em buscar padrões, semelhanças e/ou diferenças nos dados recebidos.
3. **Aprendizado semi-supervisionado:** nesse caso, a máquina consegue utilizar, ao mesmo tempo, os métodos de aprendizado supervisionado e não supervisionado. Embora o sistema receba alguns exemplos de entrada e saída de dados, a maior parte das informações que lhe são dadas não possuem o modelo desejado, obrigando a máquina a buscar esses padrões de forma não supervisionada. Das quatro classificações apresentadas, essa é a mais custosa.
4. **Aprendizado por reforço:** esse método de aprendizado consiste em um processo de **tentativa-e-erro**. É utilizado um sistema de recompensa e punição. A cada resultado desejável, a máquina ganha um tipo de ponto. Caso não seja o esperado, a máquina perde esse ponto. Assim, o computador é treinado para que consiga tomar as melhores decisões possíveis, focadas em obter um maior número de pontos.

3.2 Classificação quanto à descrição das IAs

Outra maneira possível de se classificar as IAs é através da sua descrição. A descrição de uma IA é baseada em sua força, seu alcance e suas aplicações. Nesse sentido, uma IA pode ser classificada de acordo com as seguintes categorias:

- **IA Fraca (ou Estreita):** são as IAs que são aplicadas a domínios específicos, tais como: tradutores de linguagem, assistentes virtuais e carros que se dirigem sozinhos. São IAs que são capazes de realizar muito bem algumas tarefas específicas, mas são incapazes de aprender a realizar novas tarefas.
- **IA Forte (ou Genérica):** são as IAs capazes de apresentar interação, operação e controle, sobre diversas tarefas independentes e distintas, mesmo que elas não tenham qualquer correlação entre si. Estas IAs conseguem resolver novos problemas, por meio do aprendizado autodidata de como realizar novas tarefas. As IAs Genéricas são combinações de várias estratégias para o aprendizado a partir da experiência e são capazes de operar no nível da inteligência humana.
- **Super IA ou IA Consciente:** são as IAs que possuem consciência, no nível humano. Logo, elas têm noção da sua existência. Não se tem notícia de que esse tipo de IA já tenha sido implementada e pode-se dizer que é muito pouco provável que algo parecido surgirá num futuro próximo, uma vez que o conceito de consciência é nebuloso e ainda não possui uma definição adequada.

Capítulo 4

Aprendizado de máquinas

Após a contextualização sobre o que são as Inteligências Artificiais, bem como algumas de suas classificações, é necessário o entendimento de como elas ocorrem. Para isso, realiza-se o estudo de aprendizado de máquinas, popularmente conhecido como *machine learning* (ML), que é uma subcategoria dentro do estudo das IAs. O objetivo desse capítulo consiste em trazer definições acerca desse tema.

4.1 O que é o *machine learning*

Como mencionado anteriormente, ML é um campo de estudo importante dentro do universo da Inteligência Artificial, cujo intuito é ensinar máquinas a tomarem decisões que seriam naturalmente tomadas por seres humanos, baseadas, ou não, em modelos desejados pelos desenvolvedores, dependendo da técnica de aprendizado a ser utilizada.

O principal objetivo do ML é que as máquinas possam escolher as melhores opções sem que dependam totalmente da intervenção humana para tais realizações.

4.2 Como funciona o *machine learning*

Quando se trata de programação, um dos passos mais importantes é predefinir etapas e condições para que uma ação seja executada. Contudo, quando se inicia a construção de um algoritmo, nem sempre é possível saber qual será o resultado final desejado, pois, no meio do processo, há uma série de variáveis que podem ser alteradas continuamente até que se chegue no ponto final requerido.

Quando se trabalha com o processamento de grandes volumes de dados, buscar e organizar padrões, a fim de tomar alguma decisão, são tarefas que consomem uma quantidade muito grande de tempo para uma ação humana. Na tentativa de diminuir o tempo de processamento, empregam-se técnicas de aprendizado de máquina, onde computadores são treinados para padronizar esses volumes de dados e tomar decisões baseadas na mente humana.

Existem algoritmos que, quando aplicados computacionalmente, preparam as máquinas para realizarem as ações necessárias. Assim, esse tipo de processamento de dados gera suas saídas de forma mais rápida e precisa, mesmo com grandes volumes de informação.

4.3 Qual a importância do *machine learning*

Principalmente no universo da computação, o aprendizado de máquinas possibilitou a melhor utilização do tempo necessário para se realizar tarefas e, tão ou mais importante que isso, produzir resultados mais precisos.

Quando se fala do processamento de dados, isso está relacionado com o ponto crucial para a tomada de decisões em qualquer espaço, seja no ramo empresarial, no ramo da saúde, entre outros. Logo, quanto mais rápido for a produção de respostas para questões de problemas reais, mais rápida poderá ser a sua solução.

Capítulo 5

Algoritmos para aprendizado de máquinas

Como já foi citado anteriormente, o aprendizado de máquinas pode ser realizado de acordo com as seguintes técnicas: **aprendizado supervisionado**, **aprendizado não supervisionado**, **aprendizado semi-supervisionado** e **aprendizado por reforço**. Esse capítulo lista os principais algoritmos utilizados em cada técnica de aprendizado.

5.1 Aprendizado supervisionado

Os algoritmos de aprendizado supervisionado ainda podem ser subdivididos nas seguintes classes: algoritmos de classificação e algoritmos de regressão. A seguir, são listados os principais algoritmos e um breve resumo sobre cada um deles.

5.1.1 Algoritmos de classificação

Classificação é o processo que consiste em criar categorias (ou classe) para um determinado conjunto de dados, organizando-os em tais em classes. Os algoritmos mais utilizados são os seguintes:

- ***K-Nearest Neighbors (KNN)***: busca organizar cada amostra de um conjunto de dados em classes, usando a distância em relação aos vizinhos mais próximos como parâmetro. Se a maior parte dos vizinhos estiverem na mesma classe, a amostra que está sendo utilizada é armazenada na mesma categoria.
- ***Naive Bayes***: realiza análises de frequências e de como as variáveis de entrada e saída se relacionam. Assim, com base nas estatísticas, a máquina aprende sobre os dados e retorna as melhores previsões possíveis.
- ***Logistic Regression (regressão logística)***: realiza seus cálculos usando as noções da probabilidade e da estatística. Busca calcular a saída de uma variável dependente através da análise em conjunto de duas variáveis independentes. Geralmente seus resultados retornam **sim** ou **não**.
- ***Support Vector Machines (máquinas de vetores de suporte)***: busca uma linha de separação, chamada de hiperplano, entre os dados de duas classes distintas. A máquina seleciona dois pontos, um de cada conjunto de dados, e os analisa em relação aos pontos mais próximos do outro grupo. Assim, é possível inserir a nova informação em qual das classes fizer mais sentido.

- **Decision Trees (árvores de tomada de decisão):** criam nós que se relacionam seguindo uma hierarquia entre eles. Os nós do tipo raiz guardam as informações da base de dados fornecidas e os nós do tipo folha, que são os resultados vindos das análises de dados realizadas, guardam as classes que reagrupam os dados ou os valores que as buscas retornam.

5.1.2 Algoritmos de regressão

Define-se por regressão a previsão da saída de uma variável dependente y a partir da análise de um conjunto de variáveis independentes x , por meio de equações de probabilidade e estatística. Os principais algoritmos de regressão são os seguintes:

- **Linear Regression (regressão linear):** consiste em realizar previsões do resultado de dados sem respostas, a partir de informações relacionadas e certas. Através de equações matemáticas lineares, a máquina busca encontrar o resultado das variáveis dependentes em função das variáveis independentes.
- **Polynomial Regression (regressão polinomial):** consiste em um método similar ao da regressão linear. Porém, as variáveis dependentes passam a ser modeladas em polinômios, cujo grau indica a relevância da informação. A regressão polinomial é utilizada quando os modelos lineares não devolvem o resultado esperado.
- **Principal Components Regression (regressão dos componentes principais):** busca reduzir o tamanho de um conjunto de dados, inserindo-o em classes de tamanhos menores, através de conceitos de estatística. A máquina seleciona um conjunto de dados variados e não relacionados e os aplica nos modelos de regressão citados acima.
- **Logistic Regression (regressão logística).**

5.2 Aprendizado não supervisionado

Os algoritmos de aprendizado não supervisionado ainda podem ser subdivididos nas seguintes classes: regras de associação e redução de dimensionalidade. A seguir, são listados os principais algoritmos e um breve resumo sobre cada um deles.

5.2.1 Regras de associação

O principal objetivo das regras de associação é descobrir as relações que definem grandes classes de dados. Os algoritmos que seguem essa proposta são os seguintes:

- **Apriori:** consiste em um sistema de “pontuação”. A máquina seleciona um conjunto grande de dados e realiza uma análise, ou então “pontua”, em relação a conjuntos menores, estabelecendo alguma ordem. As pontuações são usadas para criar classes de dados com aparições frequentes e inseri-las em um banco de dados maior, para que sejam coletados posteriormente.
- **Equivalence Class Clustering and bottom-up Lattice Traversal (ECLAT):** utiliza os mesmos princípios do algoritmo **Apriori**. Porém é mais eficiente, pois utiliza uma abordagem diferente, sendo capaz de realizar mais etapas. Consequentemente, é um algoritmo mais rápido.

- **BF-Growth:** também utiliza os mesmos princípios do algoritmo **Apriori**. Contudo, utiliza uma estrutura de dados em formato de árvore. Assim, é possível reduzir o número de acessos ao banco de dados, o que torna o algoritmo mais eficiente e menos custoso.

5.2.2 Redução de dimensionalidade

Ao estudar um conjunto de dados, é possível encontrar muitas variações de informações. Embora, em alguns casos, isso seja algo positivo, o volume excessivo de dados pode afetar a eficiência dos algoritmos.

Para que os algoritmos permaneçam eficientes, a redução de dimensionalidade implica em diminuir a quantidade de informações disponíveis, de forma que os algoritmos consigam processá-las sem perder a eficácia, preservando a integridade dos dados.

Neste caso, as principais técnicas utilizadas são as seguintes:

- **Missing Value Ratio:** consiste em remover as colunas de dados com muitos espaços vazios, pois não possuem informações úteis. É definido um limite de espaços vazios. As colunas que ultrapassam esse limite podem ser removidas. Isso torna-se menos custoso à máquina, pois reduz a demanda de tempo para realizar buscas não produtivas.
- **Low Variance Filter:** assim como na técnica anterior, colunas de dados com muitos espaços vazios não possuem informações relevantes. Logo, é estabelecido um limite de espaços vazios e, todas as colunas com a quantidade de espaços inferior a este limite são removidas.
- **High Correlation Filter:** colunas de dados com comportamentos muito parecidos carregam informações quase iguais. Assim, basta apenas uma das colunas para a máquina armazenar os dados relevantes. Essa técnica consiste em realizar o cálculo de um coeficiente de correlação entre as colunas e gerar um limite. Os pares de colunas cujo coeficiente de correlação ultrapasse o valor do limite são reduzidos a apenas uma coluna.
- **Random Forests / Ensemble Trees:** essa técnica é capaz de gerar um grande conjunto de dados, construído por árvores, com base em um atributo que lhe é dado. A partir disso, estatísticas de uso de cada atributo são usadas para encontrar a árvore que possui mais informações.
- **Principal Component Analysis (PCA):** consiste em transformar, de forma ortogonal, as coordenadas originais de um conjunto de dados em um novo conjunto de coordenadas, chamado de componentes principais. O primeiro componente principal tem a maior taxa de variação possível. Os componentes seguintes, também possuem a maior taxa de variação possível. Porém, há uma restrição: eles não podem ser relacionados com os componentes anteriores. Assim, é possível reduzir o volume de dados, enquanto se obtém o máximo de informação acerca das variações dos dados.
- **Backward Feature Elimination:** nessa técnica, em alguma iteração, o algoritmo de classificação é treinado com “x” características de entrada. Em seguida, é removido um recurso de entrada e o algoritmo é treinado novamente, porém com “x-1” características por “x” vezes. O recurso de entrada cuja remoção produziu o menor aumento na taxa de erro é removido, sobrando “x-1” recursos de entrada. O processo segue, sucessivamente, tratando “x” como “x!”. Ao selecionar-se o limite de erro máximo tolerável, define-se o menor número de recursos utilizados para atingir o desempenho de classificação, a partir do algoritmo escolhido.

- **Forward Feature Construction:** é o processo inverso da técnica acima. O início se dá com apenas um recurso. Progressivamente, é adicionado um recurso por vez, procurando produzir o maior aumento de desempenho.

5.3 Aprendizado semi-supervisionado

No modelo de aprendizado semi-supervisionado, utilizam-se os algoritmos de aprendizado supervisionado em conjunto com os algoritmos e as técnicas de aprendizado não supervisionado.

5.4 Aprendizado por reforço

No modelo de aprendizado por reforço, o método de ensino consiste em tentativa-e-erro, onde o programador penaliza ou recompensa a inteligência artificial. A aprendizagem ocorre por meio de uma rede neural, que só se comunica dessa forma e armazena as informações que aprendeu em um banco de dados. Para que essa rede neural seja operada, existe um algoritmo específico, chamado **Back Propagation**. Este algoritmo pode ser descrito da seguinte forma:

Baseando-se no cálculo do erro ocorrido na saída da rede neural, que representa a punição que a IA deve receber, deve-se recalcular o valor dos pesos de um vetor “x”, presente na última camada de neurônios, e, assim, sucessivamente, chegar até a camada de entrada da rede, em uma propagação de trás para a frente (Back Propagation). A partir do momento que esses valores são atualizados realizando a retropropagação, é possível calcular o erro obtido pela rede. Logo, calcula-se o erro entre o que a IA acha que é e o que realmente é, começando da última camada e indo até a primeira, até que a taxa de erro diminua e a IA aprenda o que é de fato.

Capítulo 6

Python e *machine learning*

Ao começar a idealizar o projeto de uma Inteligência Artificial, é necessário planejar qual será a melhor forma de implementação. Nos dias de hoje, os projetistas de IAs têm optado por usar a linguagem de programação *Python*. Esse capítulo busca explicar o porquê de se utilizar essa linguagem, como funciona a sua aplicação no aprendizado de máquinas e quais são as principais bibliotecas empregadas.

6.1 Visão geral da linguagem

A linguagem de programação *Python* é comumente utilizada no mundo todo e aplicável nas mais diversas áreas da computação. Ela pode ser descrita como uma linguagem de alto nível, interpretada, interativa e orientada a objetos. Uma de suas facilidades consiste em possuir fácil sintaxe, cujas palavras-chave são escritas em inglês, o que torna seu uso mais simples quando comparada a outras linguagens, tais como *C* e *Java*.

Python pode ser aplicada em diversas áreas, tais como: desenvolvimento *web*, automação com *scripts*, desenvolvimento e automação de testes em *software*, ciência de dados e, por fim, inteligências artificiais e *machine learning*.

A linguagem em questão possui diversos benefícios e, conseqüentemente, seu uso é global. Além de um aprendizado fácil e rápido, ela possui uma gama de bibliotecas padrões, é portátil (o que significa que pode ser usada em diversos ambientes e sistemas), extensível, escalável e facilmente integrada com outras linguagens de programação.

6.2 Uso de *Python* em *machine learning*

O uso de *Python* no aprendizado de máquinas dá-se por quatro motivos principais, que são: a simplicidade, a legibilidade, as poderosas bibliotecas e os ambientes de desenvolvimento integrado (IDEs).

As questões da simplicidade e da legibilidade são importantes pelo simples motivo de que, quanto menos o programador precisar se preocupar com a escrita do código, mais tempo é disponibilizado para que o mesmo possa se preocupar com a busca de soluções aplicáveis. Assim, é possível encontrar as melhores soluções, com mais foco e clareza na resolução dos problemas.

Já as bibliotecas *Python* são extremamente úteis no processo das IAs e do aprendizado de máquina, pois fornecem diversas funcionalidades adicionais no âmbito da linguagem. Ao todo, existem, hoje, cerca de 218.000 pacotes *Python* para serem utilizados nas mais diversas áreas e, boa parte delas, são aplicáveis no desenvolvimento de ML.

Por fim, seus ambientes de desenvolvimento integrado (IDEs) abrem portas para que a construção dos códigos seja um processo mais simples, pois os mesmos desfrutam de uma série de configurações, as quais tornam o dia a dia dos programadores o mais confortável possível.

6.3 Bibliotecas *Python* para *machine learning*

As bibliotecas são o maior atrativo de *Python* para seus usuários e, portanto, para os desenvolvedores de IAs. As principais bibliotecas utilizadas no aprendizado de máquinas são listadas nas subseções abaixo.

6.3.1 *Numpy*

A biblioteca *Numpy* é utilizada para o processamento de grandes matrizes multidimensionais, por meio de uma coleção de funções matemáticas de alto nível. Por isso, é utilizada para realizar os cálculos científicos fundamentais que existem em *machine learning*.

6.3.2 *SciPy*

SciPy é uma biblioteca importante, pois contém diversos módulos para otimização, álgebra linear, integração e estatística. Além disso, é utilizada para a manipulação de imagens.

6.3.3 *Scikit-learn*

Essa é uma das bibliotecas de ML mais populares para a construção de algoritmos clássicos. É construída a partir da junção de duas bibliotecas básicas, *Numpy* e *SciPy*. É uma ferramenta utilizada para mineração e análise de dados, oferecendo suporte a algoritmos de aprendizado supervisionados e não supervisionados.

6.3.4 *Theano*

A biblioteca *Theano* é utilizada para definir, avaliar e otimizar expressões matemáticas que envolvem matrizes multidimensionais, o que é obtido a partir da utilização da CPU e da GPU. Seu uso se dá em testes de unidade e autoverificação, para detectar e diagnosticar vários tipos de possíveis erros.

6.3.5 *TensorFlow*

É uma biblioteca de código aberto, utilizada para computação numérica de alto desempenho. A *TensorFlow* é uma estrutura que envolve a definição e a execução de cálculos envolvendo tensores, a fim de treinar e executar redes neurais profundas, que podem ser usadas para o desenvolvimento de aplicativos de Inteligência Artificial.

6.3.6 *Keras*

A biblioteca *Keras* pode ser definida como uma API de redes neurais de alto nível, que pode ser executada em *TensorFlow*, *CNTK* ou *Theano*. Ela permite uma prototipagem fácil e rápida.

6.3.7 *PyTorch*

Assim como a *TensorFlow*, a *PyTorch* é uma biblioteca de código aberto, baseada em uma outra biblioteca, chamada *Torch*. Ela possui uma gama de ferramentas, que oferecem suporte às seguintes atividades: visão por computador, processamento de linguagem natural (PNL) e outros programas de ML. Com ela, os desenvolvedores podem realizar cálculos em tensores com a aceleração de GPU e criar gráficos computacionais.

6.3.8 *Pandas*

Pandas é uma biblioteca utilizada para extrair e preparar dados. A biblioteca fornece estruturas de dados de alto nível para realizar a análise dos dados e fornece muitos métodos embutidos para tatear, combinar e filtrar dados.

6.3.9 *Matplotlib*

A *Matplotlib* é utilizada para a visualização de dados, possibilitando que o programador encontre padrões nas informações disponíveis. É uma biblioteca de plotagem 2D, usada para criar gráficos e plotagens, fornecendo diversos tipos de visualizações, dentre elas: histogramas, gráficos de erro e gráficos de barra.

Capítulo 7

Conclusão

Nos dias atuais, a tecnologia vem proporcionando o avanço de várias áreas da sociedade, dentre elas: a saúde, a educação, o mundo dos negócios e, principalmente, as vidas pessoais.

O uso da Inteligência Artificial atrelado a funções diárias abriu portas para que diversas atividades sejam realizadas de forma autônoma à intervenção humana, a fim de que o tempo resultante da não realização dessas atividades seja utilizado para outras tarefas, consideradas mais importantes.

Baseado nos conhecimentos teóricos presentes neste documento, é possível compreender como funcionam os mecanismos utilizados para construir uma Inteligência Artificial, desde os conceitos básicos para a elaboração do projeto até a linguagem de programação escolhida para o desenvolvimento de tais objetivos.

Referências bibliográficas

- [Ace] Acervo Lima. Disponível em: “<https://acervolima.com/ml-algoritmo-eclat/>”. Acesso em: 27/01/2023.
- [Alu] Alura. Disponível em: “<https://cursos.alura.com.br/forum/topico-como-funciona-o-algoritmo-svm-128139>”. Acesso em: 27/01/2023.
- [AWSa] AWS – Amazon. Disponível em: “<https://aws.amazon.com/pt/what-is/linear-regression/>”. Acesso em: 27/01/2023.
- [AWSb] AWS – Amazon. Disponível em: “<https://aws.amazon.com/pt/what-is/logistic-regression/>”. Acesso em: 27/01/2023.
- [AWSc] AWS – Amazon. Disponível em: “<https://aws.amazon.com/pt/what-is/machine-learning/>”. Acesso em: 27/01/2023.
- [AWSd] AWS – Amazon. Disponível em: “<https://aws.amazon.com/pt/what-is/neural-network/>”. Acesso em: 27/01/2023.
- [Cet] Cetax. Disponível em: “<https://cetax.com.br/machine-learning/>”. Acesso em: 27/01/2023.
- [DEC] DECOM – UFOP. Disponível em: “<http://www2.decom.ufop.br/terralab/tipos-de-aprendizado-de-maquina-e-algumas-aplicacoes/>”. Acesso em: 27/01/2023.
- [Def] Definirtec. Disponível em: “<https://definirtec.com/algoritmo-apriori/>”. Acesso em: 27/01/2023.
- [Did] Didatica. Disponível em: “<https://didatica.tech/o-que-e-e-como-funciona-o-algoritmo-knn/>”. Acesso em: 27/01/2023.
- [E-d] E-disciplinas – USP. Disponível em: “https://edisciplinas.usp.br/pluginfile.php/5775889/mod_resource/content/2/Aula03_Regression.pdf”. Acesso em: 27/01/2023.
- [Gee] Geeks for Geeks. Disponível em: “<https://www.geeksforgeeks.org/principal-component-regression-pcr/>”. Acesso em: 27/01/2023.
- [Ima] Imasters. Disponível em: “<https://imasters.com.br/back-end/o-que-e-machine-learning-e-como-funciona>”. Acesso em: 27/01/2023.
- [LAM] LAMFO – UNB. Disponível em: “<https://lamfo-unb.github.io/2017/07/27/tres-tipos-am/>”. Acesso em: 27/01/2023.

- [Leu] Leucotron. Disponível em: “[https://blog.leucotron.com.br/machine-learning-e-inteligencia-artificial/#:~:text=Machine%20learning%20\(ML\)%20%E2%80%94%20ou,a%20ajuda%20expl%C3%ADcita%20de%20humanos.](https://blog.leucotron.com.br/machine-learning-e-inteligencia-artificial/#:~:text=Machine%20learning%20(ML)%20%E2%80%94%20ou,a%20ajuda%20expl%C3%ADcita%20de%20humanos.)”. Acesso em: 27/01/2023.
- [Max] Maxwell. Disponível em: “<https://www.maxwell.vrac.puc-rio.br/57547/57547.PDF>”. Acesso em: 27/01/2023.
- [Med] Medium. Disponível em: “<https://medium.com/ensina-ai/redes-neurais-perceptron-multicamadas-e-o-algoritmo-backpropagation-eaf89778f5>”. Acesso em: 27/01/2023.
- [Oxf] Oxford languages: Web dictionary. Disponível em: “<https://languages.oup.com/google-dictionary-en/>”. Acesso em: 27/01/2023.
- [PET] Grupo PET-Tele. Disponível em: “<http://www.telecom.uff.br/pet>”. Acesso em: 27/01/2023.
- [Pro] Programa de Educação Tutorial - PET. Disponível em: “http://portal.mec.gov.br/index.php?option=com_content&view=article&id=12223&ativo=481&Itemid=480”. Acesso em: 27/01/2023.
- [Repa] Repositório UFSCAR. Disponível em: “<https://repositorio.ufscar.br/bitstream/handle/ufscar/332/DissLCP.pdf?sequence=1&isAllowed=y>”. Acesso em: 27/01/2023.
- [Repb] Repositório UFU. Disponível em: “<https://repositorio.ufu.br/bitstream/123456789/25264/3/Gest%C3%A3oConhecimentoUsando.pdf>”. Acesso em: 27/01/2023.
- [SAS] SAS. Disponível em: “https://www.sas.com/pt_br/insights/analytics/machine-learning.html”. Acesso em: 27/01/2023.
- [Soma] Somostera. Disponível em: “<https://blog.somostera.com/data-science/arvores-de-decisao>”. Acesso em: 27/01/2023.
- [Somb] Somostera. Disponível em: “<https://blog.somostera.com/data-science/naive-bayes>”. Acesso em: 27/01/2023.
- [THE] THE WORLD BANK. Disponível em: “<https://datatopics.worldbank.org/world-development-indicators>”. Acesso em: 27/01/2023.
- [Tow] Towards Data Science. Disponível em: “<https://towardsdatascience.com/the-eclat-algorithm-8ae3276d2d17>”. Acesso em: 27/01/2023.
- [Wor] Wordpress. Disponível em: “<https://mineracaodedados.wordpress.com/2015/06/13/7-tecnicas-para-reducao-da-dimensionalidade/>”. Acesso em: 27/01/2023.
- [Zen] Zendesk. Disponível em: “<https://www.zendesk.com.br/blog/machine-learning/>”. Acesso em: 27/01/2023.