AI Cheat Sheet

ALTOROS

AI Basics

ARTIFICIAL INTELLIGENCE (AI)
"The theory and development of computer systems able to perform tasks normally requiring human intelligence."

Oxford English Dictionary

BUSINESS USE CASES

1. AI and CUSTOMER SERVICE ENHANCEMENT:
Business value is generated through optimization of the "front-office" operations.

2. AI and PROCESSES OPTIMIZATION:
The value is generated through optimization of the "back-office" operations in order to reduce costs and improve compliance.

3. AI and INSIGHTS GENERATION:
New business value is created from the existing data by enabling better, more consistent, and faster decision making.

MACHINE LEARNING (ML)
ML is one of the AI approaches, which uses statistical techniques to give computer systems the ability to "learn" (i.e., progressively improve performance on a specific task) from some data, without being explicitly programmed.

DEEP LEARNING (DL)
DL is a machine learning method. It uses neural networks and allows us to train an algorithm to predict outputs, given a set of inputs. A neural network consists of an input layer, a hidden layer(s), and an output layer. The "deep" in deep learning refers to having more than one hidden layer of neurons in a neural network. Both supervised and unsupervised learning can be used for training.

TYPES OF ML ALGORITHMS

Supervised learning algorithms make predictions based on a set of examples. They are trained using labeled data sets that have inputs and expected outputs.

Semi-supervised learning learning algorithms use unlabeled examples together with a small amount of labeled data to improve the learning accuracy.

Unsupervised learning algorithms work with totally unlabeled data. They are designed to discover the intrinsic patterns that underlie the data, such as a clustering structure, a low-dimensional manifold, or a sparse tree and graph.

Reinforcement learning algorithms analyze and optimize the behavior of an agent based on the feedback from the environment. Machines try different scenarios to discover which actions yield the greatest reward, rather than being told which actions to take.

Supervised learning

ALGORITHMS

Regression [Linear, Polynomial, Nonparametric] — the algorithm is asked to predict a numerical value given some input: "How much money would a bank gain (lose) by lending to a certain client?"

Classification [Naive Bayes, k-NN,SVM, Random Forest, Neural Networks] — the algorithm is asked to specify which of k categories some input belongs to. "Will a client be able to pay his loan back?"

Learning to rank [HITS, SALSA, PageRank] — the algorithm is asked to rank (i.e., to produce a permutation of items in new, unseen lists) in a way that is similar to the rankings in the training data. "What are the top 10 world's safest banks?"

Forecasting [Trending, Time-Series Modeling, Neural Networks] — the algorithm is asked to generate predictions based on available data.

USE CASES

Ranking is used in bioinformatics, drug discovery, information retrieval, sentiment analysis, machine translation, and online advertising.

Classification is applied to e-mail spam filtering, bank customers loan pay back willingness prediction, cancer tumour cells identification, sentiment analysis, drugs classification, facial keypoints detection, and pedestrians detection in an automotive car driving.

Regression is employed for pricing optimization, modeling historical sales in order to determine a pricing strategy and predict demand for products that have not been sold before.

Supervised ML methods are leveraged to minimize the number of returns for online purchases.

DL methods are used for predicting the next purchases of customers in advance.

Forecasting models are the bread and butter for business intelligence.

OPEN-SOURCE FRAMEWORKS

R Data Science libraries: Caret, randomForest, nnet, dplyr

Python Data Science libraries: Scikit-learn, Scipy, NumPy, NLTK, Matplotlib, CatBoost, XGBoost, PyTorch, Caffe2, Theano, Keras, TensorFlow, OpenCV

Semi-supervised learning

ALGORITHMS

Pseudo labeling is an algorithm used for expanding training data sets. It requires some data to be labeled first, but then it uses this data in a conjunction with a large amount of unlabeled data to learn a model for a domain. It is compatible with almost all neural network models and training methods.

Generative models [VAE and GANs] are neural network models that can replicate the data distribution given as an input. This allows to generate "fake-but-realistic" data points from real data points.

USE CASES

Pseudo labeling is applicable to malware/fraud detection, document structure analysis, stock predictions, real-time diagnostics, NLP/speech recognition, and any other type of problems where small labeled data set size represents a constraint.

Generative models are used for real-time visual processing, text-to-image generation, image-to-image translation, increasing image resolution, or predicting the next video frame.

OPEN-SOURCE FRAMEWORKS

TensorFlow, numPy, Scikit-learn

Reinforcement learning

ALGORITHMS

Q-learning — the algorithm is based on a mathematical optimization method known as dynamic programming. Given current states of a system, the algorithm finds an optimal policy (i.e., set of actions) that maximizes Q-value function.

State-Action-Reward-State-Action — the algorithm resembles Q-learning a lot, but learns Q-value based on the action performed by the current policy instead of the greedy policy.

Deep Q Network — the algorithm leverages a Neural Network to estimate the Q-value function. It resolves some limitations of the Q-learning algorithm.

Deep Deterministic Policy Gradient — the algorithm is designed for such problems as physical control tasks, where the action space is continuous.

USE CASES

Manufacturing. Robots use deep reinforcement learning to pick a device from one box and put it in a container. They learn from successful and failed attempts.

Inventory Management. RL algorithms reduce transit time for stocking and retrieving products in the warehouse to optimize space utilization and warehouse operations.

Delivery management. Reinforcement learning is used to solve the problems of operational research and logistics (e.g., the split delivery vehicle routing problem).

Finance sector. The Q-learning algorithm is able to learn an optimal stock market trading strategy with a single instruction: maximize the value of a portfolio.

OPEN-SOURCE FRAMEWORKS

RL-Glue, OpenAI Gym, RLPy, BURLAP

Unsupervised learning

ALGORITHMS

Clustering [k-means, Mean-Shift, Hierarchical, Fuzzy c-means] — the algorithm is asked to group the similar kind of data items by considering the most satisfied condition: all the items in the same group (called a cluster) are more similar to each other than to the items in the other groups (clusters).

Anomalies detection [Density-Based, SVM-Based, Clustering-Based] — the computer program sifts through a set of events or objects and flags some of them as unusual or atypical.

Dimensionality reduction [PCA, Singular Value Decomposition, LDA] — the algorithm is asked to reduce the number of random variables under consideration by obtaining a set of principal variables. Dimensionality reduction can be divided into feature selection and feature extraction.

Missing data imputation [mean imputation, k-NN] — the algorithm is given examples with some missing entries and is asked to provide the values of the missing entries.

Association rules learning [AIS, SETM, APRIORI, FP-GROWTH] — is a rule-based method for discovering interesting relations between variables in large databases. It is intended to identify strong rules discovered in databases using some measures of interestingness.

USE CASES

Unsupervised learning methods are used in healthcare and pharma for such tasks as human genetic clustering and genome sequence analysis. They are also widely used across all industries for customers segmentation, recommender systems, chatbots, topic modeling, anomalies detection, grouping of shopping items, search results grouping, etc.

OPEN-SOURCE FRAMEWORKS

R Data Science libraries: Caret, Rattle, e1071, nnet, dplyr

Python Data Science libraries: BigARTM, Tesseract, Scrapy, Scikit-learn, PyTorch, Caffe2, Theano, Keras, TensorFlow

Information sources:

Ian Goodfellow, Yoshua Bengio and Aaron Courville (2016) "Deep Learning", MIT Press

Andrew Burgess (2017) "The Executive Guide to Artificial Intelligence", Springer

Hui Li (2017) "Which machine learning algorithm should I use?", SAS Blog

Machine Learning with Python Cheat Sheet

ALTOROS

General-purpose machine learning

The `Auto_ml` framework is developed for automating a machine learning process and making it easier to get real-time predictions in production. It automates analytics, feature engineering, feature selection, model selection, data formatting, hyperparameter optimization, etc.

The `machine-learning` framework provides a web interface and an API for classification and regression. The support vector machines and support vector regression algorithms are available via the framework out of the box.

`XGBoost` implements machine learning algorithms under the Gradient Boosting technique. XGBoost provides a parallel tree boosting (also known as GBDT or GBM), which solves many data science problems in a fast and accurate manner.

`scikit-learn` is a Python module for machine learning built on top of the SciPy framework. The module encapsulates methods for enabling data preprocessing, classification, regression, clustering, model selection, etc.

`SimpleAI` is a library for solving search and statistical classification problems. The search module includes traditional and local search algorithms, constraint satisfaction problem algorithm, and interactive execution of search algorithms. The classification module of SimpleAI supports decision tree, Naive Bayes, and k-nearest neighbours classifiers.

`MLlib in Apache Spark` is a distributed machine learning library in Spark. Its goal is to make practical machine learning scalable and easy. It provides a set of common machine learning algorithms, as well as utilities for linear algebra, statistics, data handling, featurization, etc.

`Theano` is a numerical computation library for Python. It allows you to efficiently define, optimize, and evaluate mathematical expressions involving multi-dimensional arrays.

`TensorFlow` is an open-source software library for numerical computation using data flow graphs. Originally developed by the Google Brain team, TensorFlow allows to easily deploy computations across a variety of platforms (CPUs, GPUs, or TPUs), as well as on clusters of servers, mobile and edge devices, etc.It is widely used in a bundle with neural networks.

`Keras` is a high-level neural networks API, written in Python and capable of running on top of TensorFlow or Theano. It was developed with a focus on enabling fast experimentation.

`Caffe` is a deep learning framework that supports many different types of architectures geared towards image classification and image segmentation.

`Caffe2` is a lightweight, modular, and scalable deep learning framework. Based on the original Caffe, Caffe2 aims to provide an easy and straightforward way to experiment with deep learning and leverage community contributions of new models and algorithms.

`PyTorch` is a Python package that provides two high-level features: tensor computation (like NumPy) with strong GPU acceleration and deep neural networks.

`CatBoost` is a general purpose gradient boosting on decision trees library with categorical features support out of the box. It is an easy-to-install and well documented package. It supports CPU and GPU (even multi-GPU) computation.

Computer vision

`scikit-image` is a collection of algorithms for image processing in Python. It includes algorithms for segmentation, geometric transformations, color space manipulation, analysis, filtering, morphology, feature detection, etc.

`OpenCV` is a computer vision framework designed for computational efficiency with a strong focus on real-time applications. Usage ranges from interactive art to mines inspection and advanced robotics.

`SimpleCV` is a framework that gives access to several high-powered computer vision libraries, such as OpenCV. To use the framework, you don't need to first learn bit depths, file formats, color spaces, buffer management, eigenvalues, and matrix versus bitmap storage.

`OpenFace` is a Python and Torch implementation of face recognition with deep neural networks.

The `face_recognition` framework allows for recognizing and manipulating faces from Python or from the command line."

`Dockerface` is a Docker-based solution for face detection using Faster R-CNN.

`Detectron` is a software system by Facebook AI Research that implements state-of-the-art object detection algorithms, including Mask R-CNN. It is written in Python and powered by the Caffe2 framework.

Natural language processing

`NLTK` (the Natural Language Toolkit) is a suite of open-source Python modules, data sets, and tutorials supporting research and development in natural language processing.

`TextBlob` is a Python library for processing textual data. It provides a simple API for diving into common natural language processing tasks, such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, etc.

`PyNLPl` is a library for natural language processing that contains various modules useful for a variety of natural language processing tasks, such as extraction of n-grams and frequency lists or building simple language models.

`Polyglot` is a multilingual text processing toolkit. It supports language detection (196 languages), tokenization (165 languages), named entity recognition (40 languages), part-of-speech tagging (16 languages), sentiment analysis (136 languages), and other features.

`Fuzzy Wuzzy` is a fuzzy string matching implementation in Python. The algorithm uses Levenshtein Distance to calculate the differences between sequences.

`jellyfish` is a Python library for approximate and phonetic matching of strings.

Topic modeling

`BigARTM` is a powerful tool for topic modeling. Additive regularization of topic models is the innovative approach lying at the core of the BigARTM library. The solution helps to build multi-objective models by adding the weighted sums of regularizers to the optimization criterion. BigARTM supports different features, including sparsing, smoothing, topics decorrelation, etc.

`Gensim` is a Python library for topic modelling, document indexing, and similarity retrieval with large corpora.

`topik` is a topic modeling toolbox, which provides a full-suite and high-level interface for anyone interested in applying topic modeling. It includes a bunch of utilities beyond statistical modeling algorithms.

Chatbots

`End-to-end-negotiator` is a PyTorch implementation of research paper "Deal or No Deal? End-to-End Learning for Negotiation Dialogues" by Facebook AI Research. The code trains neural networks to hold negotiations in natural language and enables reinforcement learning self-play and rollout-based planning.

`DeepPavlov` is an open-source library for building end-to-end dialog systems and training chatbots built on TensorFlow and Keras.

`awesome-bots` is a GitHub repository with a collection of materials dedicated to chatbots.

Reinforcement learning

`DeepMind Lab` is a first-person 3D game platform designed for research and development of general artificial intelligence and machine learning systems. DeepMind Lab can be used to study how autonomous artificial agents may learn complex tasks in large, partially observed, and visually diverse worlds.

`OpenAI Baselines` is a set of high-quality implementations of reinforcement learning algorithms.

`OpenAI Gym` is a toolkit for developing and comparing reinforcement learning algorithms.

`RLPy` is a framework for conducting sequential decision-making experiments. The current focus of this project is on value-function-based reinforcement learning.

`universe` is a software platform for measuring and training an AI's general intelligence across the world's supply of games, websites, and other applications.

Data analysis and data visualization

`Apache Spark` is a fast and general cluster computing system for big data. It provides high-level APIs in Python and an optimized engine that supports general computation graphs for data analysis.

`NumPy` is a fundamental package needed for scientific computing with Python.

`SciPy` is open-source software for mathematics, science, and engineering. It includes modules for statistics, optimization, integration, linear algebra, Fourier transforms, signal and image processing, ODE solvers, etc.

`Pandas` is a library providing high-performance, easy-to-use data structures and data analysis tools for the Python language.

`PyMC` is a Python module that implements the Bayesian statistical models and fitting algorithms, including the Markov chain Monte Carlo methods. Its flexibility and extensibility make it applicable to a large variety of problems. Along with core sampling functionality, PyMC includes methods for summarizing output, plotting, goodness-of-fit, and convergence diagnostics.

`statsmodels` is a package for statistical modeling and econometrics in Python. It provides a complement to SciPy for statistical computations, including descriptive statistics and estimation, as well as inference for statistical models.

`Matplotlib` is a Python 2D plotting library, which produces publication-quality figures in a variety of hard copy formats and interactive environments across platforms.

`ggplot` is a plotting system for Python built for making professional looking plots quickly and with a minimum of code.

`scikit-plot` is a visualization library for quick and easy generation of common plots in data analysis and machine learning.

Other projects

The `deepdream` repository contains IPython Notebook with sample code, complementing Google Research blog post about the neural network art.

`NeuralTalk2` is an efficient image captioning code based on recurrent neural networks.

`Kaggle-cifar` contains code for the CIFAR-10 Kaggle competition on image recognition. It uses a cuda-convnet architecture.

The `Lime` project is about explaining what machine learning classifiers (or models) are doing. At the moment, it supports explaining individual predictions for text classifiers or classifiers that act on tables (e.g., the NumPy arrays of numerical or categorical data) or images. The project aims at helping users to understand and interact meaningfully with machine learning.

`DeepJ` is a deep learning model for style-specific music generation.

`deep-neuroevolution` is a GitHub repository containing implementation of the neuroevolution approach, where neural networks are optimized through the evolutionary algorithms. It is an effective method to train deep neural networks for reinforcement learning tasks.

Free online books

Understanding Machine Learning: From Theory to Algorithms by Shai Shalev-Shwartz and Shai Ben-David (2014)

Natural Language Processing with Python by Steven Bird, Ewan Klein, and Edward Loper (2009)

Deep Learning by Yoshua Bengio, Ian Goodfellow, and Aaron Courville (2015)

Neural Networks and Deep Learning by Michael Nielsen (2014)

Deep Learning by Microsoft Research (2013)

Deep Learning in Neural Networks: An Overview by Jurgen Schmidhuber (2014)