

CHAPTER 1

1184100 MUH AMRI IRIANTO

March 8, 2021

SOAL CHAPTER 1 TEORI

1 TEORI UMUM

1.1 Definisi, Sejarah dan perkembangan Kecerdasan Buatan

1.1.1 Definisi AI

Kecerdasan buatan atau intelegensi artifisial dalam bahasa Inggris: Artificial Intelligence (AI) merupakan kecerdasan yang dimasukkan ke suatu sistem dan bisa diatur dalam konteks ilmiah. Terdapat beberapa pengertian kecerdasan buatan atau artificial intelligence (AI) dalam berbagai sudut pandang.

Kecerdasan buatan adalah usaha untuk memodelkan proses berfikir manusia dan mendesain mesin agar dapat menirukan perilaku manusia. Kecerdasan buatan juga suatu penelitian, aplikasi dan intruksi yang terkait dengan pemrograman komputer dalam melakukan suatu kegiatan menurut pandangan manusia.

1.1.2 Sejarah

Dimulai pada tahun 1940 pada era komputer elektronik benih-benih kecerdasan buatan ditanamkan oleh para filsuf klasik yang ingin menggambarkan proses berfikir manusia sebagai manipulasi dengan cara mekanis. Pada tahun 1940-an sebuah mesin yang didasarkan perangkat menginspirasi segelintir ilmuwan untuk mulai serius untuk membangun sebuah otak elektronik.

Istilah kecerdasan buatan pertama kali dikemukakan pada 1956 di konferensi Dartmouth, sejak itu kecerdasan buatan terus dikembangkan sebagai penelitian untuk teori-teori dan juga berkembangnya prinsip. Walaupun seperti itu istilah kecerdasan baru muncul pada tahun 1956 tetapi teori kecerdasan buatan sudah muncul sejak 1941.

1.1.3 Perkembangan

Zaman komputer Elektronik (1941)

Pada tahun 1941 alat penyimpan dan pemrosesan informasi ditemukan penemuan tersebut dinamakan komputer elektronik yang dikembangkan USA dan Jerman. Saat itu komputer melibatkan ribuan kabel untuk menjalankan suatu program.

Masa Awal Persiapan AI (1943-1956)

Tahun 1943 Warren McCulloch dan Walter Pitt mengemukakan tiga hal : Pengetahuan fisiologi dasar dan fungsi sel syaraf dalam otak, analisis tentang logika proposisi, dan teori komputasi turing mereka membuat model sel syaraf tiruan dimana setiap sel syaraf digambarkan dengan ON dan OFF menunjukkan bahwa setiap fungsi dapat dihitung dengan suatu sel syaraf dan semua dapat diimplementasikan secara logis

tahun 1950, Nobeert Wiener membuat penelitian mengenai prinsip-prinsip teori feedback. Contoh yang terkenal adalah thermostat. Penemuan ini juga merupakan awal dari perkembangan AI.

tahun 1956, John McCarthy meyakinkan Minsky, Claude Shannon dan Nathaniel Rochester untuk membantunya melakukan penelitian dalam bidang Otomata, Jaringan Syaraf dan pembelajaran inteligensi. Mereka mengerjakan proyek ini selama 2 bulan di Dartmouth. Hasilnya adalah program yang mampu berpikir non-numerik dan menyelesaikan masalah pemikiran, yang dinamakan Principia Mathematica. Hal ini menjadikan McCarthy disebut sebagai bapak kecerdasan buatan.

Awal Perkembangan AI (1952-1969)

Pada tahun-tahun pertama perkembangannya, kecerdasan buatan mengalami banyak kesuksesan. Diawali dengan kesuksesan Newell dan Simon dengan sebuah program yang disebut General Problem Solver.

Pada tahun 1958, McCarthy di MIT AI Lab Memo No.1 mendefinisikan bahasa pemrograman tingkat tinggi yaitu LISP, yang sekarang mendominasi pembuatan program-program kecerdasan buatan. Kemudian, McCarthy membuat program yang dinamakan Programs with Common Sense. Di dalam program tersebut, dibuat rancangan untuk menggunakan pengetahuan dalam mencari solusi.

Pada tahun 1959, Nathaniel Rochester dari IBM dan mahasiswa-mahasiswanya mengeluarkan program kecerdasan buatan yaitu Geometry Theorem Prover. Program ini dapat mengeluarkan suatu teorema menggunakan aksioma-aksioma yang ada.

Sistem Berbasis Pengetahuan (1969 - 1979)

Pengetahuan adalah kekuatan pendukung kecerdasan buatan. Hal ini dibuktikan dengan program yang dibuat oleh Ed Feigenbaum, Bruce Buchanan dan Joshua Lederberg yang membuat program untuk memecahkan masalah struktur molekul dari informasi yang didapatkan dari spectrometer massa.

Kembalinya Jaringan Syaraf Tiruan (1986 - Sekarang)

Meskipun bidang ilmu komputer menolak jaringan syaraf tiruan setelah diterbitkannya buku "Perceptrons" karangan Minsky dan Papert, para ilmuwan masih mempelajari bidang ilmu tersebut dari sudut pandang yang lain, yaitu fisika. Para ahli fisika seperti Hopfield (1982) menggunakan teknik-teknik mekanika statistika untuk menganalisis sifat-sifat penyimpanan dan optimasi pada jaringan syaraf. Para ahli psikologi, David Rumelhart dan Geoff Hinton, melanjutkan penelitian mengenai model jaringan syaraf tiruan pada memori.

1.2 Definisi

1.2.1 supervise learning unsupervised learning

supervised learning merupakan proses pengelompokan data yang telah memiliki label dan akan dikelompokkan berdasarkan labelnya untuk mendapatkan label tertentu untuk melakukan proses training terlebih dahulu.

unsupervised learning merupakan proses pengelompokan data yang tidak memiliki label. sehingga kita bebas menentukan jumlah kelompok data yang dibuat

1.2.2 Klasifikasi dan Regresi

klasifikasi adalah proses menemukan atau menemukan model (fungsi) yang membantu dalam memisahkan data menjadi beberapa kelas kategorikal yang berarti data yang dikategorikan dalam label yang berbeda sesuai dengan parameter

regresi adalah proses menemukan model atau fungsi untuk membedakan data menjadi nilai riil kontinu alih-alih menggunakan kelas secara matematis, dengan masalah regresi, seseorang berusaha menemukan perkiraan fungsi dengan deviasi kesalahan minimum

1.2.3 Dataset, Training Set dan Testing Set

dataset istilah informan yang merujuk pada kumpulan data. secara umum dataset berisi lebih dari variabel dan menyangkut satu topik: itu mungkin menyangkut satu sampel

Training set adalah bagian dari data set untuk memprediksi atau menjalankan fungsi dari sebuah algoritma ML. agar mesin yang kita latih mencari korelasi sendiri atau belajar dengan pola data yang diberikan

Testing set adalah bagian dataset yang kita tes untuk melihat keakuratannya atau dengan kata lain performanya

2 IMPLEMENTASI

untuk pertama kita bisa melakukan instalasi scikit learn seperti gambar di bawah melalui link <https://scikit-learn.org/stable/tutorial/basic/tutorial.html> akan muncul juga tutorial mengenai penggunaan scikit learn

2.1 Instalasi Skit learn

kita membuka anaconda prompt terus kita bisa memasukkan

conda create -n sklearn-env seperti gambar di bawah

```
(base) C:\Users\user>conda create -n sklearn-env
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\aplikasi\Anaconda\envs\sklearn-env

Proceed ([y]/n)? y

Preparing transaction: done
Verifying transaction: done
Executing transaction: done
#
# To activate this environment, use
#
#     $ conda activate sklearn-env
#
# To deactivate an active environment, use
#
#     $ conda deactivate
```

setelah itu kita bisa melakukan instalasi seperti gambar dibawah untuk melihat apakah SCIKIT LEARN telah di instal

```
(sklearn-env) C:\Users\user>conda install -c conda-forge scikit-learn
Collecting package metadata (current_repodata.json): done
Solving environment: done

## Package Plan ##

  environment location: D:\aplikasi\Anaconda\envs\sklearn-env

added / updated specs:
- scikit-learn
```

Setelah itu,di harapkan melakuka pengecekan dengan conda seperti gambar dibawah

```
(sklearn-env) C:\Users\user>conda list
# packages in environment at D:\aplikasi\Anaconda\envs\sklearn-env:
#
# Name                                Version                                Build                                Channel
ca-certificates                      2020.12.5                              h5b45459_0                          conda-forge
certifi                              2020.12.5                              py39hcbf5309_1                      conda-forge
intel-openmp                         2020.3                                  h57928b3_311                        conda-forge
joblib                               1.0.1                                  pyhd8ed1ab_0                        conda-forge
libblas                              3.9.0                                  8_mkl                               conda-forge
libcblas                             3.9.0                                  8_mkl                               conda-forge
liblapack                            3.9.0                                  8_mkl                               conda-forge
m2w64-gcc-libgfortran                5.3.0                                  6                                    conda-forge
m2w64-gcc-libstdc++                  5.3.0                                  7                                    conda-forge
m2w64-gcc-libstdc++-core             5.3.0                                  7                                    conda-forge
m2w64-gmp                             6.1.0                                  2                                    conda-forge
m2w64-libwinpthread-git              5.0.0.4634.697f757                    2                                    conda-forge
mkl                                   2020.4                                  hb70f87d_311                        conda-forge
msys2-conda-epoch                    20160418                               1                                    conda-forge
numpy                                 1.20.1                                  py39h6635163_0                      conda-forge
openssl                               1.1.1j                                  h8ffe710_0                          conda-forge
pip                                   21.0.1                                  pyhd8ed1ab_0                        conda-forge
python                                3.9.2                                  h7840368_0_cpython                  conda-forge
python_abi                            3.9                                    1_cp39                              conda-forge
scikit-learn                         0.24.1                                  py39he931e04_0                      conda-forge
scipy                                 1.6.0                                  py39hc0c34ad_0                      conda-forge
setuptools                            49.6.0                                  py39hcbf5309_3                      conda-forge
```

2.2 Loading an example dataset

pada halaman ini akan dilakukan test example data set untuk mengambil data pada scikit learn kita masukkan kode untuk melakukan test :

```
from sklearn import datasets
iris = datasets.load_iris()
digits = datasets.load_digits()
print(digits.data)
```

maka hasilnya akan terlihat seperti gambar dibawah

```
In [11]: runfile('E:/semester 6/kecerdasan buatan/Chapter 1/
Contoh 1.py', wdir='E:/semester 6/kecerdasan buatan/Chapter 1')
[[ 0.  0.  5. ...  0.  0.  0.]
 [ 0.  0.  0. ... 10.  0.  0.]
 [ 0.  0.  0. ... 16.  9.  0.]
 ...
 [ 0.  0.  1. ...  6.  0.  0.]
 [ 0.  0.  2. ... 12.  0.  0.]
 [ 0.  0. 10. ... 12.  1.  0.]]
```

2.3 Learning and predicting

pada percobaan kali ini kita akan menggunakan learning and predicting untuk melakukan prediksi dan penggunaan learning langsung seperti contoh dibawah

```
from sklearn import datasets
iris = datasets.load_iris()
digits = datasets.load_digits()
from sklearn import svm
clf = svm.SVC(gamma=0.001, C=100)
clf.fit(digits.data[:-1], digits.target[:-1])
hasil = clf.predict(digits.data[-1:])
print(hasil)
```

maka hasilnya akan seperti gambar dibawah:

```
In [18]: runfile('E:/semester 6/kecerdasan buatan/Chapter 1/contoh 2.py',
wdir='E:/semester 6/kecerdasan buatan/Chapter 1')
[8]
```

2.4 Model persistence

pada tahap ini akan menggunakan model persistence untuk berikut kita bisa melihat seperti contoh dibawah untuk melakukan uji coba

```
from sklearn import svm
from sklearn import datasets
clf = svm.SVC()
X, y = datasets.load_iris(return_X_y=True)
clf.fit(X, y)
```

```
#pickle
import pickle
s = pickle.dumps(clf)
clf2 = pickle.loads(s)
clf2.predict(X[0:1])
print(y[0])

#joblib
from joblib import dump, load
dump(clf, '1184100.joblib')
clf3 = load('1184100.joblib')
print(clf3.predict(X[0:1]))
```

maka hasilnya akan seperti gambar dibawah

```
In [22]: runfile('E:/semester 6/kecerdasan buatan/Chapter 1/contoh
3.py', wdir='E:/semester 6/kecerdasan buatan/Chapter 1')
0
[0]
```

2.5 Conventions

scikit-learn estimator mengikuti aturan tertentu untuk membuat perilakunya lebih prediktif, seperti contoh dibawah:

```
import numpy as np
from sklearn import random_projection

rng = np.random.RandomState(0)
X = rng.rand(10, 2000)
X = np.array(X, dtype='float32')
X.dtype

transformer = random_projection.GaussianRandomProjection()
X_new = transformer.fit_transform(X)
X_new.dtype
print(X_new)
```

maka hasilnya akan seperti gambar dibawah

```
In [23]: runfile('E:/semester 6/kecerdasan buatan/Chapter 1/Contoh
4.py', wdir='E:/semester 6/kecerdasan buatan/Chapter 1')
[[-0.47921741 -2.20104732 -0.94977111 ... 0.9104148 -0.39463488
-0.34533761]
 [ 0.21270861 -1.52596949 -0.51392087 ... 0.34580548 -0.49168487
-0.51148108]
 [-0.29692422 -1.76077448 -0.9435932 ... 0.7234875 -0.45175262
-0.80233693]
 ...
 [-0.22021783 -1.98258364 -0.41770065 ... 0.88338713 -0.34002892
-0.22582745]
 [-0.42749494 -1.60303494 -0.35088177 ... 0.49354972 -0.17173604
-0.18213682]
 [-0.14597102 -1.73993241 -0.87961557 ... 0.70398457 -0.24599354
-0.72208925]]
```

3 Penanganan Errorr

1. pada eororr petama ditunjukkan found input yang salah pada inpun foun variabel

```
File "D:\aplikasi\Anaconda\lib\site-packages\sklearn\utils
\validation.py", line 255, in check_consistent_length
    raise ValueError("Found input variables with inconsistent numbers
of"
```

```
ValueError: Found input variables with inconsistent numbers of
samples: [1796, 1]
```

2. erorr kedua di tunjukaan pada input float sehingga varibale tidak dapat terbaca dan terjadinya errorr

```
File "E:\semester 6\kecerdasan buatan\Chapter 1\Contoh 4.py", line
17, in <module>
    X_new = transformer.fit_transform()
```

```
TypeError: fit_transform() missing 1 required positional argument:
'X'
```