

Tugas

Kecerdasan Buatan



Rayhan Yuda Lesmana

1184007

Applied Bachelor of Informatics Engineering

Program Studi D4 Teknik Informatika

Applied Bachelor Program of Informatics Engineering

Politeknik Pos Indonesia

Bandung 2021

‘Jika Kamu tidak dapat menahan lelahnya belajar,
Maka kamu harus sanggup menahan perihnya Kebodohan.’
Imam Syafi’i

Acknowledgements

Pertama-tama kami panjatkan puji dan syukur kepada Allah SWT yang telah memberikan rahmat dan hidayah-Nya sehingga Buku Pedoman Tingkat Akhir ini dapat diselesaikan.

Abstract

Buku Pedoman ini dibuat dengan tujuan memberikan acuan, bagi mahasiswa Tingkat Akhir dan dosen Pembimbing. Pada intinya buku ini menjelaskan secara lengkap tentang Standar pengerjaan Intership dan Tugas Akhir di Program Studi D4 Teknik Informatika, dan juga mengatur mekanisme, teknik penulisan, serta penilaiannya. Dengan demikian diharapkan semua pihak yang terlibat dalam aktivitas Bimbingan Mahasiswa Tingkat Akhir berjalan lancar dan sesuai dengan standar.

Contents

1	Mengenal Kecerdasan Buatan dan Scikit-Learn	1
1.1	Teori	1
1.2	Instalasi	2
1.3	Penanganan Error	6
	Bibliography	8

List of Figures

Chapter 1

Mengenal Kecerdasan Buatan dan Scikit-Learn

1.1 Teori

Praktek teori penunjang yang dikerjakan :

1. Sejarah Perkembangan dan penjelasan Definisi *Artificial Intelligence*. Artificial Intelligence merupakan suatu ilmu pada bidang komputer yang mana dapat membuat suatu sistem yang cerdas untuk menyelesaikan suatu permasalahan. Pada akhir tahun 1955 istilah dari *Artificial Intelligence* pertama kali diciptakan. Pada awal kemunculnya kecerdasan buatan ini mengalami banyak kesuksesan, yang dimana diawali dengan kesuksesan Newell dan Simon dengan sebuah program yang disebut dengan *The Logic Theorist*, Program ini merepresentasikan masalah sebagai model pohon, lalu penyelesaiannya dengan memilih cabang yang akan menghasilkan kesimpulan terbenar. Program ini berdampak besar dan menjadi batu loncatan penting dalam mengembangkan bidang AI. Pada tahun 1956 John McCarthy dari Massachusetts Institute of Technology dianggap sebagai bapak AI, menyelenggarakan konferensi untuk menarik para ahli komputer bertemu, dengan nama kegiatan “The Dartmouth summer research project on artificial intelligence.”. Setelah itu pada tahun 1970, proyek DARPA (*Defence Advanced Research Project Agency*) berhasil menyelesaikan studi kasus mengenai pemetaan jalan. Selanjutnya awal abad ke 21 atau lebih tepatnya pada tahun 2003, DARPA juga berhasil menciptakan asisten pribadi cerdas. Sejak itulah kecerdasan buatan mulai mengalami perkembangan hingga saat ini menjadi program yang sangat kompleks dengan menerapkan algoritma dan *deep learning*. Oleh karena itu kecerdasan buatan mampu memberikan so-

lusi yang kompleks untuk berbagai macam permasalahan dengan inovasi yang bervariasi.

2. Definisi supervised learning, klasifikasi, regresi dan unsupervised learning. Data set, training set dan testing set

- Supervised Learning dan Unsupervised Learning

Supervised learning Adalah sebuah pembelajaran yang dapat memprediksi suatu pola yang mana pola tersebut sudah ada contoh datanya, jadi pola yang sudah ada tadi merupakan hasil dari data yang sudah ada contoh datanya *Unsupervised Learning* yaitu tidak menggunakan label dalam memprediksi targetnya melainkan dengan melihat kesamaan dari atribut yang dimiliki. Jika memiliki kesamaan dari atribut atau sifat data yang diekstrak maka akan dimasukkan kedalam satu kelompok (*clustering*). Sehingga dapat menimbulkan banyak kelompok.

- Klasifikasi dan Regresi

Adalah suatu teknik mengklasifikasikan atau mengelompokkan beberapa item yang belum ada label kedalam sebuah kelas distrit. Regresi adalah suatu teknik analisis untuk mendefinisikan suatu relasi diantara dua variabel atau lebih, berguna untuk menemukan fungsi model data dengan meminimalkan error atau selisih nilai prediksi dengan nilai sebenarnya.

- Data set, Training set dan Testing set

Adalah suatu objek yang menjelaskan sebuah data dengan ralasinya di memory. Lalu training set adalah bagian dari data set yang berfungsi membuat suatu algoritma sesuai tujuan yang di harapkan, sedangkan testing set yaitu bagian dari data set yang berfungsi untuk melihat ke akuratan dari sebuah data yang di tes.

1.2 Instalasi

1. Melakukan instalasi pada anaconda prompt dengan perintah " pip install -U scikit-learn".


```
Command Prompt
C:\Users\rayha>pip install scikit-learn
WARNING: Retrying (Retry(total=4, connect=None, read=None, redirect=None, status=None)) after connection broken by 'ConnectTimeoutError(<pip._vendor.urllib3
HTTPSConnection object at 0x000002328B9251F0>, 'Connection to pypi.org timed out. (connect timeout=15)')': /simple/scikit-learn/
Collecting scikit-learn
  Downloading scikit_learn-0.24.1-cp38-cp38-win_amd64.whl (6.9 MB)
    | 6.9 MB 409 kB/s
Collecting numpy>=1.13.3
  Downloading numpy-1.20.1-cp38-cp38-win_amd64.whl (13.7 MB)
    | 13.7 MB 344 kB/s
Collecting threadpoolctl>=2.0.0
  Using cached threadpoolctl-2.1.0-py3-none-any.whl (12 kB)
Collecting scipy>=0.19.1
  Downloading scipy-1.6.1-cp38-cp38-win_amd64.whl (32.7 MB)
    | 32.7 MB 262 kB/s
Collecting joblib>=0.11
  Downloading joblib-1.0.1-py3-none-any.whl (303 kB)
    | 303 kB 312 kB/s
Installing collected packages: numpy, threadpoolctl, scipy, joblib, scikit-learn
Successfully installed joblib-1.0.1 numpy-1.20.1 scikit-learn-0.24.1 scipy-1.6.1 threadpoolctl-2.1.0
C:\Users\rayha>
```

- Setelah itu masuk ke link website yang telah diberikan yaitu "<https://scikit-learn.org/stable/tutorial/basic/tutorial.html>".

```
1 from sklearn.linear_model import LogisticRegression
2 from sklearn import set_config
3
4
5 lr = LogisticRegression(penalty='l1')
6 print('Default representation:')
7 print(lr)
8 # LogisticRegression(C=1.0, class_weight=None, dual=False,
9 # fit_intercept=True,
10 # intercept_scaling=1, l1_ratio=None, max_iter
11 # =100,
12 # multi_class='auto', n_jobs=None, penalty='l1',
13 # random_state=None, solver='warn', tol=0.0001,
14 # verbose=0,
15 # warm_start=False)
16 set_config(print_changed_only=True)
17 print('\nWith changed_only option:')
18 print(lr)
19 # LogisticRegression(penalty='l1')
```

- Mencoba Loading example dataset

```
1 from sklearn import datasets
```

```

2 #mengimport class dataset dari scikit learn library
3 iris = datasets.load_iris()
4 #memuat dan memasukkan dataset iris ke variabel bernama iris
5 digits = datasets.load_digits()
6 #membuat dan memasukkan dataset digits ke variabel digits
7
8 print(digits.data)
9 #memberikan akses ke fitur yang dapat digunakan untuk
   mengklasifikasikan sampel digit dan menampilkan diconsole
10
11 digits.target
12 #memberikan informasi tentang data yang berhubungan atau juga dapat
   dijadikan sebagai label
13
14 digits.images[0]
15 #Data selalu berupa array 2D, shape (n.samples, n.features),
   meskipun data aslinya mungkin memiliki bentuk yang berbeda.

```

4. Mencoba Learning and predicting

```

1 from sklearn import svm
2 #perintah untuk mengimport class svm dari package sklearn
3
4 digits = datasets.load_digits()
5 #memasukkan dan memuat dataset digits ke variabel digits
6
7 clf = svm.SVC(gamma=0.001, C=100.)
8 #clf sebagai estimator/parameter, svm.SVC sebagai class, gamma
   sebagai parameter untuk menerapkan nilai secara manual
9
10 clf.fit(digits.data[:-1], digits.target[:-1])
11 #clf sebagai estimator/parameter, fit sebagai metode, digits.data
   sebagai item,[:-1] sebagai syntax python dan menampilkan
   outputnya
12
13
14 print(clf.predict(digits.data[-1:]))
15 #predict sebagai metode lainnya, digit.data sebagai item
   menampilkan outputnya

```

5. Mencoba Model Persistence

```

1 from sklearn import svm
2 #mengimport class svm dari scikit learn library
3 from sklearn import datasets
4 #mengimport class dataset dari scikit learn library
5
6 clf = svm.SVC(gamma=0.001, C=100.)
7 #memanggil class SVC dan menset argument constructor SVC serta
   ditampung di variabel clf
8 X, y = datasets.load_iris(return_X_y=True)
9 #meload datasets iris dan ditampung di variabel x untuk data
   sedangkan y untuk target

```

```

10
11 clf.fit(X, y)
12 #memanggil method fit untuk melakukan training data dengan argumen
    data dan target dari database iris
13
14 import pickle
15 #mengimport pickle (agar dapat terbaca)
16 s = pickle.dumps(clf)
17 #memanggil method dumps dengan argumen clf dan ditampung pada
    variabel s
18 clf2 = pickle.loads(s)
19 #memanggil method loads dengan argumen s dan ditampung di variabel
    clf2
20 clf2.predict(X[0:1])
21 #menampilkan hasil dari method predict dengan argumen data variabel
    x
22
23 from joblib import dump, load
24 #mengimport dump dan load dari library joblib
25 dump(clf, '1184007.joblib')
26 #memanggil method dumps dengan argumen clf dari nama file joblib
27 clf3 = load('1184007.joblib')
28 #memanggil method load dengan argumen nama file joblibnya
29 print(clf3.predict(X[0:1]))
30 #menampilkan hasil dari method predict dengan argumen data variabel

```

6. Mencoba Conventions

```

1 #Type casting
2 import numpy as np
3 from sklearn import random_projection
4
5 rng = np.random.RandomState(0)
6 X = rng.rand(10, 2000)
7 X = np.array(X, dtype='float32')
8 print(X.dtype)
9
10
11 transformer = random_projection.GaussianRandomProjection()
12 X_new = transformer.fit_transform(X)
13 print(X_new.dtype)
14
15 from sklearn import datasets
16 from sklearn.svm import SVC
17 iris = datasets.load_iris()
18 clf = SVC(gamma=0.001, C=100.)
19 clf.fit(iris.data, iris.target)
20 print(list(clf.predict(iris.data[:3])))
21 clf.fit(iris.data, iris.target_names[iris.target])
22 print(list(clf.predict(iris.data[:3])))
23
24 #refitting and updating parameters
25 import numpy as np
26 from sklearn.datasets import load_iris

```

```

27 from sklearn.svm import SVC
28 X, y = load_iris(return_X_y=True)
29 clf = SVC(gamma=0.001, C=100.)
30 clf.set_params(kernel='linear').fit(X, y)
31 clf.set_params(kernel='rbf').fit(X, y)
32 print(clf.predict(X[:5]))
33
34 #multiclass vs multilabel fitting
35 from sklearn.svm import SVC
36 from sklearn.multiclass import OneVsRestClassifier
37 from sklearn.preprocessing import LabelBinarizer
38
39 X = [[1, 2], [2, 4], [4, 5], [3, 2], [3, 1]]
40 y = [0, 0, 1, 1, 2]
41
42 classif = OneVsRestClassifier(estimator=SVC(random_state=0, gamma
    =0.001, C=100.))
43 print(classif.fit(X, y).predict(X))
44 y = LabelBinarizer().fit_transform(y)
45 print(classif.fit(X, y).predict(X))
46
47 from sklearn.preprocessing import MultiLabelBinarizer
48 y = [[0, 1], [0, 2], [1, 3], [0, 2, 3], [2, 4]]
49 y = MultiLabelBinarizer().fit_transform(y)
50 print(classif.fit(X, y).predict(X))

```

1.3 Penanganan Error

1. Screenshot Error

```

In [8]: runfile('F:/Kecerdasan Buatan/Chapter1/modelpersistence.py', wdir='F:/Kecerdasan Buatan/Chapter1')
Traceback (most recent call last):
  File "C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages\IPython\core\interactiveshell.py",
line 3325, in run_code
    exec(code_obj, self.user_global_ns, self.user_ns)
  File "<ipython-input-8-33aa20b28124>", line 1, in <module>
    runfile('F:/Kecerdasan Buatan/Chapter1/modelpersistence.py', wdir='F:/Kecerdasan Buatan/Chapter1')
  File "C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 827, in runfile
    execfile(filename, namespace)
  File "C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages\spyder_kernels\customize
\spydercustomize.py", line 110, in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)
  File "F:/Kecerdasan Buatan/Chapter1/modelpersistence.py", line 11
    clf2 = pickle.loads(s) #memanggil method loads dengan argumen s dan ditampung di variabel clf2
    ^
IndentationError: unexpected indent

In [9]:

```

```

In [6]: runfile('F:/Kecerdasan Buatan/Chapter1/dataset.py', wdir='F:/Kecerdasan Buatan/Chapter1')
Traceback (most recent call last):

  File "<ipython-input-6-cf0b001d1260>", line 1, in <module>
    runfile('F:/Kecerdasan Buatan/Chapter1/dataset.py', wdir='F:/Kecerdasan Buatan/Chapter1')

  File "C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages\spyder_kernels\customize\spydercustomize.py", line 827,
in runfile
    execfile(filename, namespace)

  File "C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages\spyder_kernels\customize\spydercustomize.py", line 110,
in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)

  File "F:/Kecerdasan Buatan/Chapter1/dataset.py", line 1, in <module>
    from sklearn import dataset

ImportError: cannot import name 'dataset' from 'sklearn' (C:\Users\rayha\AppData\Local\Continuum\anaconda3\lib\site-packages
\sklearn\__init__.py)

In [7]:

```

2. Tuliskan kode error dan jenis error

IndentationError (Muncul saat ada indentasi yang salah).

ImportError (Muncul saat modul yang hendak diimpor tidak ditemukan)

3. Solusi dari error tersebut

Kita harus lebih teliti lagi dan fokus pada saat encoding agar tidak terjadi identasi dan menuliskan nama file / nama import modul agar tidak terjadi error.

Bibliography