p-ISSN 2087-1716
e-ISSN 2548-7779

ILKOM Jurnal Ilmiah Volume 11 Nomor 3 Desember 2019
Terakreditasi peringkat 3 SK. No. 28/E/KPT/2019

ANALISIS PERFORMA METODE K-NEAREST NEIGHBOR UNTUK IDENTIFIKASI JENIS KACA

Mus Mulyadi Baharuddin[1], Tasrif Hasanuddin[2], Huzain Azis[3]
[1]musmulyadiofficial@gmail.com, [2]tasrif.hasanuddin@umi.ac.id, [3]huzain.azis@umi.ac.id
[1,2,3] Universitas Muslim Indonesia
[3]*corresponding author*

Abstrak

Saat ini industri membuat berbagai jenis barang yang memiliki bahan dasar kaca, diantaranya kaca mobil *float* , jendela bangunan *non float*, lampu, Toples, dan Peralatan Makan. Kaca–kaca tersebut memiliki bahan produksi yang sama, yang membedakan antara satu dan lainya adalah komposisi bahan produksinya. Algoritma K-Nearest Neighbor (KNN)) yang merupakan salah satu metode klasifikasi pada data mining dan juga menjadi algoritma *supervised learning* pada *machine learning* adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Penelitian ini mencakup pengukuran performa (akurasi, presisi, *recall* dan *f-measure*) metode KNN dengan berbagai macam nilai K pada objek 1000 data produksi jenis kaca yang diperoleh dari pusat dataset. dapat disimpulkan bahwa dengan menguji nilai K=3 hingga K=9 maka diperoleh nilai performa paling baik pada K=3, dimana tingkat akurasi mencapai 64%, presisi 63%, recall 71%, dan F-Measure sebesar 67%.
Kata Kunci : K-Nearest Neighbor, klasifikasi , *supervised learning*, *data mining*, *machine learning*

Abstract

Nowadays, the industry makes various types of goods that have glass-based materials, float car window panes, non-float building windows, lamps, jars, and tableware. These glasses have the same production material, the difference between one and the other is the composition of the production material. K-Nearest Neighbor (KNN) algorithm which is one of the classification methods in data mining and also a supervised learning algorithm in machine learning is a method for classifying objects based on learning data that is the closest distance to the object.. This study discusses the performance measurement (accuracy, precision, recall and f-measure) of the KNN method with a variety of values on 1000 glass type production data objects obtained from the central UCI Machine Learning Repository dataset. The conclusion of this research is the results of the value of K = 3 to K = 9, the best performance values obtained at K = 3, where the level of accuracy reaches 64%, 63% precision, 71% recall, and F-Measure of 67%.
Keywords: K-Nearest Neighbor, classification, supervised learning, data mining, machine learning

1. Pendahuluan

Perkembangan teknologi komputer sangat berkembang khususnya pada sistem pengenalan atau pengidentifikasian yang banyak digunakan sebagai objek riset dan pengembangan. Sistem ini menjadi salah satu riset yang populer dalam hal ini untuk mengidentifikasi suatu objek. Algoritma K-Nearest Neighbor (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek yang diuji. Algoritma K-NN sangatlah sederhana, bekerja berdasarkan jarak terpendek [1]. KNN akan mengelempokkan hasil perhitungan dengan data latih yang mempunyai kerabat terbanyak dalam nilai jangkauan yang ditentukan. Jarak antara data latih dan data uji dihitung menggunakan persamaan Euclidean [2]**.**

Data yang diperoleh dari *UCI Machine Learning Repository* yang dikelola oleh *Home Office Forensic Science Service* dan *Diagnostic Products Corporation* terlihat bahwa produksi jenis kaca memiliki bahan yang sama namun yang membedakannya adalah komposisi produksi, Adapun komposisi produksi yang dimaksud adalah RI (Refractive Index), Na (Sodium), Mg (Magnesium), Al (Aluminium), Si (Silicon), K (Potasium), Ca (Calcium), Ba (Barium), Dan Fe (Iron).

Beberapa penelitian sebelumnya [3][4][5] mencoba mencari performa metode KNN dengan 1-3 nilai K yang berbeda, berdasarkan hal tersebut maka penulis mencoba melakukan penelitian mengenai performa metode KNN pada dataset bahan produksi kaca dengan melakukan identifikasi jenis kaca berdasarkan komposisi produksi kaca performa yang akan diukur yaitu akurasi, presisi, *recall* dan *f – measure* pada 7 macam nilai K yaitu K 3 hingga K 9.

DOI: https://doi.org/10.33096/ilkom.v11i3.489.269-274

p-ISSN 2087-1716
e-ISSN 2548-7779

ILKOM Jurnal Ilmiah Volume 11 Nomor 3 Desember 2019
Terakreditasi peringkat 3 SK. No. 28/E/KPT/2019

2. Metode

Data mining

Data mining adalah proses yang menggunakan statistik, matematika, kecerdasan buatan, dan *machine learning* untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat [4]. Data Mining didefinisikan sebagai proses penemuan pola dalam data. Berdasarkan tugasnya, data mining dikelompokkan menjadi deskripsi, estimasi, prediksi, klasifikasi, clustering dan asosiasi. Proses dalam tahap data mining terdiri dari tiga langkah Utama, yaitu data Preparation Pada langkah ini, data dipilih, dibersihkan, dan dilakukan preprocessed mengikuti pedoman dan knowledge dari ahli domain yang menangkap dan mengintegrasikan data internal dan eksternal ke dalam tinjauan organisasi secara menyeluruh. Penggunaan algoritma data mining dilakukan untuk menggali data yang terintegrasi untuk memudahkan identifikasi informasi bernilai [6]. Namun semakin besar data yang diolah maka semakin besar pula waktu prosesnya [7].

Machine Learning

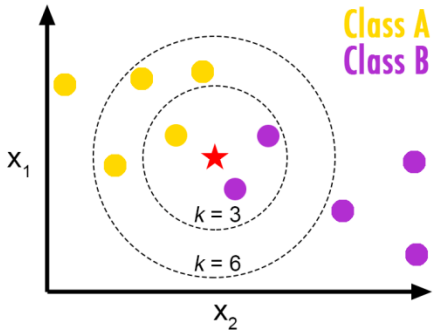
Machine Learning sering digambarkan sebagai pembelajaran dari pengalaman atau tanpa pengawasan dari manusia. Dalam masalah belajar yang diawasi, sebuah program memprediksi output untuk input dengan belajar dari pasangan input dan output berlabel; yaitu, program belajar dari contoh jawaban yang benar. Dalam tanpa pengawasan belajar, suatu program tidak belajar dari data berlabel [8]. *Machine Learning* mengeksplorasi studi dan konstruksi algoritma yang dapat belajar dari dan membuat prediksi pada data. Algoritma tersebut beroperasi dengan membangun model dari input contoh untuk membuat prediksi berbasis data atau keputusan, daripada mengikuti instruksi program yang benar-benar statis [9].

K-Nearest Neighbor

K-Nearest Neighbor (k-NN atau KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran (*neighbor*) yang jaraknya paling dekat dengan objek tersebut. Dekat atau jauhnya *neighbor* biasanya dihitung berdasarkan jarak *Euclidean*. diperlukan suatu sistem klasifikasi sebagai sebuah sistem yang mampu mencari informasi [10]. Metode KNN dibagi menjadi dua fase, yaitu pembelajaran (*training*) dan klasifikasi atau pengujian (*testing*). Pada fase pembelajaran, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk data yang akan diuji coba (yang klasifikasinya tidak diketahui). Jarak dari vektor yang baru ini terhadap seluruh vektor data pembelajaran dihitung, dan sejumlah *k* buah *neighbor* yang paling dekat diambil. Perhitungan jarak ketetanggan menggunakan algoritma *euclidien* seperti yang ditunjukkan pada persamaan 1.

$$\text{euc} = \sqrt{((a_1 - b_1)^2 + \cdots + (a\_n - b\_n)^2)} \qquad (1)$$

Dimana a = a1,a2, .. , an, dan b = b1, b2, .., bn mewakili n nilai atribut dari dua record. Untuk atribut dengan nilai kategori [5]. Sebuah titik akan diprediksi jenisnya berdasarkan pada klasifikasi terbanyak dari *neighbor* di sekitarnya,ilustrasiny dapat dilihat pada Gambar 1.



Gambar 1. Ilustrasi penggunaan nilai *k* pada metode KNN

Nilai k yang terbaik untuk KNN tergantung pada data. Secara umum, nilai k yang tinggi akan mengurangi efek noise pada klasifikasi, tetapi membuat batasan antara setiap klasifikasi menjadi lebih kabur. Nilai k yang bagus dapat dipilih dengan optimasi parameter, misalnya dengan menggunakan

DOI: https://doi.org/10.33096/ilkom.v11i3.489.269-274

p-ISSN 2087-1716
e-ISSN 2548-7779

ILKOM Jurnal Ilmiah Volume 11 Nomor 3 Desember 2019
Terakreditasi peringkat 3 SK. No. 28/E/KPT/2019

cross-validation. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain, k = 1) disebut algoritma nearest neighbor [3].

Pyhton

Python merupakan bahasa pemrograman yang berorientasi obyek dinamis, dapat digunakan untuk bermacam-macam pengembangan perangkat lunak. *Phyton* menyediakan dukungan yang kuat untuk integrasi dengan bahasa pemrograman lain dan alat-alat bantu lainnya. *Pyhton* hadir dengan pustakapustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari [11]. *Python* hadir dengan pustaka-pustaka standar yang dapat diperluas serta dapat dipelajari hanya dalam beberapa hari. Bahasa pemrograman yang interpretatif multiguna dengan filosofi perancangan yang berfokus pada tingkat keterbacaan kode. *Python* diklaim sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas, dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif [12].

Performa

Performa dari klasifikasi dapat dievaluasi dengan menghitung nilai akurasi, presisi, *recall* dan *f-measure*[13]. Dimana Akurasi didefinisikan sebagai tingkat kedekatan antara nilai prediksi dengan nilai aktual. rumus akurasi dipaparkan pada persamaan 2. Presisi didefinisikan sebagai rasio item relevan yang dipilih terhadap semua item yang terpilih. Presisi dapat diartikan sebagai kecocokan antara permintaan informasi dengan jawaban terhadap permintaan tersebut. rumus presisi ditunjukkan pada persamaan 3.

Recall didefinisikan sebagai rasio dari item relevan yang dipilih terhadap total jumlah item relevan yang tersedia. Rumus *Recall* diuraikan pada persamaan 4, dan *F-Measure* adalah *harmonic mean* antara nilai presisi dan *recall, F-measure* juga kadang disebut dengan nama F_1-Score. Rumus *F-Measure* dijabarkan pada persamaan 5.

$$AKURASI = \frac{(\mathrm{TP} + \mathrm{TN})}{(\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN})} \qquad (2)$$

$$PRESISI = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \qquad (3)$$

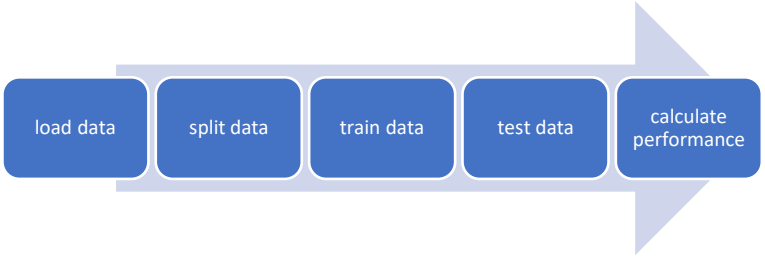
$$RECALL = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \qquad (4)$$

$$F - Measure = 2\frac{Presisi\ x\ Recall}{Presisi + Recall} \qquad (5)$$

Dimana TP adalah True Positif , TN adalah True Negative, FP adalah False Positif dan FN adalah *False Negative*.

Rancangan / Model

Rancangan atau model pada penelitian ini dibentuk sesuai kebutuhan proses metode KNN dimana dimulai dari mengumpulkan dataset, *split* atau membagi dataset, implementasi metode KNN hingga menghitung performa metode KNN pada pada dataset tersebut. alur model ditunjukkan pada Gambar 2.



Gambar 2. Alur rancangan penelitian

DOI: https://doi.org/10.33096/ilkom.v11i3.489.269-274

p-ISSN 2087-1716
e-ISSN 2548-7779

ILKOM Jurnal Ilmiah Volume 11 Nomor 3 Desember 2019
Terakreditasi peringkat 3 SK. No. 28/E/KPT/2019

Sample Dan Data

Data yang digunakan pada penelitian ini diperoleh dari *UCI Machine Learning Repository* yang dikelola oleh B. German dari *Central Research Establishment Home Office Forensic Science Service* dan Vina Spiehler, Ph.D., *dari DABFT Diagnostic Products Corporation.,* terlihat bahwa produksi jenis kaca memiliki bahan yang sama namun yang membedakannya adalah komposisi produksi, Adapun komposisi produksi yang dimaksud adalah RI *(Refractive Index)*, Na *(Sodium)*, Mg *(Magnesium)*, Al *(Aluminium)*, Si *(Silicon)*, K *(Potasium)*, Ca *(Calcium)*, Ba *(Barium)*, Dan Fe *(Iron)*. data tersebut di unggah pada tahun 1987 dan terus di update hingga tahun 2018.

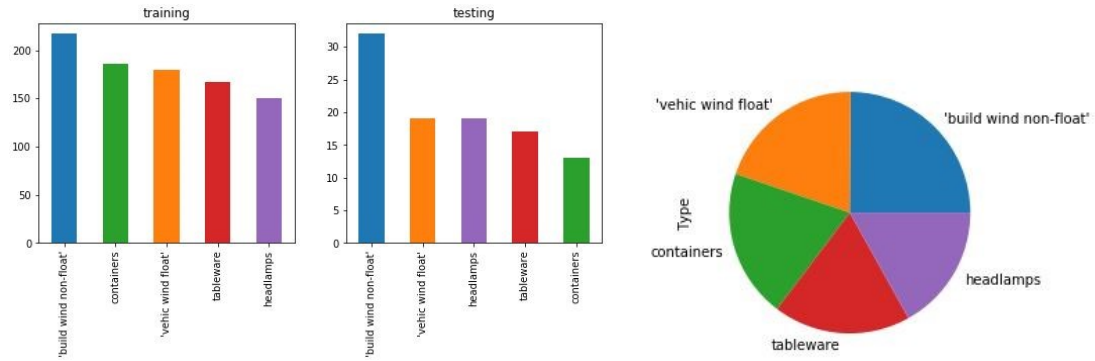
Teknik Analisis

Teknik analisis dilakukan dengan cara menghitung performa metode KNN dimana performa yang diukur yaitu akurasi, presisi, recall dan *f-measure* menggunakan persaman 2, 3 dan 4. Keseluruhan tahap penelitian ini menggunakan *scikit-learn library* sebagai *machine learning tools*.Tahap awal yang dilakukan adalah mengumpulkan data, data yang diperoleh sebanyak 1000 data Jenis Kaca. Tahap selanjutnya adalah melakukan split data dimana 90% digunakan sebagai data *training* dan 10% sebagai data *testing*. Uraian data tersebut di tunjukkan pada Table 1.

Tabel 1. Pembagian Dataset

Dataset	Jumlah	Class	Split
Data 1	1000	build wind non-float 249	90% Training
		Containers 199	10% Testing
		vehic wind float 199	
		Tableware 184	
		Headlamps 169	

Gambar 3 memvisualisasikan data dalam bentuk pie dan bar dataset, dimana pie chart menunjukkan perbandingan keseluruhan jumlah dataset dan bar chart menujukkan pembagian dataset *training dan testing*.



Gambar 3. Visualisai Pie 1000 Dataset berdasarkan type

Tahap selanjutnya adalah menerapkan metode KNN menggunakan data *testing* dan data *training* yang telah dipersiapkan sebelumnya. Pada tahap terakhir dilakukan perhitungan performa dari seluruh data *testing* dengan berbagai simulasi ketetanggaan pada metode KNN. Tabel 2 menunjukkan beberapa perintah utama yang digunakan menggunakan *scikit-learn library*.

Tabel 2 *Source code* implementasi metode KNN

Ket	Source Code
Load	dataset=pd.read_csv(' Dataset_kaca1.csv')
Split	x = dataset.iloc[:, 0:9]
	y = dataset.iloc[:, 9]
	x_train, x_test, y_train, y_test = train_test_split(x,y, random_state = 0, test_size=0.1)
Train	classifier = KNeighborsClassifier(n_neighbors=3,p=5,metric='euclidean')
	classifier.fit (x_train, y_train)

DOI: https://doi.org/10.33096/ilkom.v11i3.489.269-274

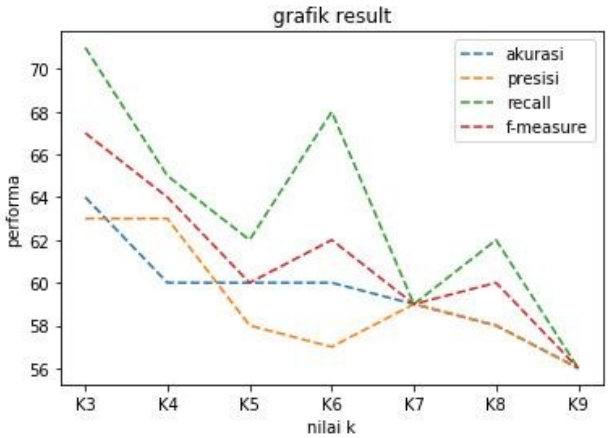
Ket	Source Code
Test Result	y_pred = classifier.predict(x_test) cm = confusion_matrix(y_test, y_pred) accuracy_score(y_test, y_pred) precision_score(y_test, y_pred, average = None) recall_score(y_test, y_pred, average = None)) f1_score(y_test, y_pred, average=None)

3. Hasil dan Pembahasan

Kesimpulan pengujian dilakukan setelah proses metode KNN dilakukan, Pengujian performa pada penelitian ini disimulasikan pada berbagai nilai ketetanggan metode KNN dengan nilai K=3 hingga nilai K=9. Hasil uji performa metode KNN di tunjukkan pada Table 3 dan Gambar 4.

Tabel 3 Hasil uji performa metode KNN

No	K	Akurasi	Presisi	Recall	F-Measure
1	3	0.64	0.63888889	0.71875	0.67647059
2	4	0.6	0.63636364	0.65625	0.64615385
3	5	0.6	0.58823529	0.625	0.60606061
4	6	0.6	0.57894737	0.6875	0.62857143
5	7	0.59	0.59375	0.59375	0.59375
6	8	0.58	0.58823529	0.625	0.60606061
7	9	0.56	0.5625	0.5625	0.5625



Gambar 4. Grafik Performa perhitungan Dataset

Berdasarkan data tersebut nilai akurasi tidak cukup baik untuk diterapkan ke tahap Decision Support System (DSS), DSS sebagai interatif aplkasi [14], dan perlu megukur usability aplikasi. Karena Usability adalah kunci keberhasilan manajemen software[15].

4. Kesimpulan dan Saran

Berdasarkan hasil penelitian ini maka penulis dapat menarik beberapa kesimpulan, yaitu dengan menyimulasikan knn dengan nilai K=3 hingga K=9 pada dataset data produksi jenis kaca maka diperoleh nilai performa paling baik pada k=3, dimana tingkat akurasi mencapai 64%, presisi 63%, recall 71%, dan F-Measure sebanyak 67%.

Berdasarkan kesimpulan diatas, maka penulis menyarakan beberapa hal agar kiranya dapat dijadikan bahan pertimbangan untuk penelitian selanjutnya (1) Jumlah data yang lebih besar atau metode yang berbeda dapat digunkan pada penelitian selanjutnya agar dapat menghitung nilai performa yang baru. (2) Penelitian selanjutnya dapat mencoba *cross validation* untuk mencari nilai performa yang baru dengan cara melakukan berbagai simulai data.

DOI: https://doi.org/10.33096/ilkom.v11i3.489.269-274

Daftar Pustaka

[1] R. Adi, "Implementasi Algoritma K-Nearest Neighbor Untuk Identifikasi Implementasi Algoritma K-Nearest Neighbor Untuk Identifikasi Kualitas Air (Studi Kasus : Pdam Kota Surakarta)," no. April, 2018.
[2] A. Apriansyah, Ilhamsyah, and T. Rismawan, "Prototype Kunci Otomatis Pada Pintu Berdasarkan Suara Pengguna Menggunakan Metode KNN (K-Nearest Neighbor)," *J. Coding, Sist. Komput. Untan*, vol. 04, no. 1, pp. 45–56, 2016.
[3] I. A. A. Angreni, S. A. Adisasmita, and M. I. Ramli, "Pengaruh Nilai K Pada Metode K-NEAREST NEIGHBOR (KNN) Terhadap Tingkat Akurasi Identifikasi Kerusakan Jalan," vol. 7, no. 2, pp. 63–70, 2018.
[4] A. A. Karim, H. Azis, and Y. Salim, "Kinerja Metode C4.5 dalam Penyaluran Bantuan Dana Bencana 1," vol. 3, no. 2, pp. 84–87, 2018.
[5] M. Lestari, "Penerapan Algoritma Klasifikasi Nearest Neighbor (K-NN) Untuk Mendeteksi Penyakit Jantung," *Fakt. Exacta*, vol. 7, no. September 2010, pp. 366–371, 2014.
[6] H. Leidiyana, "Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bemotor," *J. Penelit. Ilmu Komputer, Syst. Embed. Log.*, vol. 1, no. 1, pp. 65–76, 2013.
[7] N. Fadhillah, H. Azis, and D. Lantara, "Validasi Pencarian Kata Kunci Menggunakan Algoritma Levenshtein Distance Berdasarkan Metode Approximate String Matching," vol. 3, no. 2, pp. 3–7, 2018.
[8] Gavin Hackeling 2014, *Mastering Machine Learning with scikit-learn*. .
[9] M. amayr. & Stephane.ploix, "Machine Learning with Python and Scikit-Learn," 2015.
[10] A. Fitria, Muslim, and H. Azis, "Analisis Kinerja Sistem Klasifikasi Skripsi menggunakan Metode Naïve Bayes Classifier," vol. 3, no. 2, pp. 102–106, 2018.
[11] B. Santoso, "Bahasa Pemrograman Python di Platform GNU/LINUX," pp. 1–9, 2016.
[12] K. R. R, A. Rahmansyah, W. Darwin, and A. R. Box, "Penggunaan Bahasa Pemrograman Python Sebagai Pusat Kendali Pada Robot 10-D," pp. 23–26, 2017.
[13] C. A. Ul Hassan, M. S. Khan, and M. A. Shah, "Comparison of Machine Learning Algorithms in Data classification," *2018 24th Int. Conf. Autom. Comput.*, no. September, pp. 1–6, 2019.
[14] L. Nurhayati and H. Azis, "Perancangan Sistem Pendukung Keputusan Untuk Proses Kenaikan Jabatan Struktural Pada Biro Kepegawaian," pp. 6–7, 2016.
[15] N. Puspitasari, V. N. Vadilla, U. Hairah, H. Azis, M. Wati, and E. Budiman, "Usability Study of Student Academic Portal from a User ' s Perspective," *2018 2nd East Indones. Conf. Comput. Inf. Technol.*, pp. 108–113, 2018.

