**Pengenalan dan penggunaan algoritma K-NN (K-Nearest Neigbor)**

Oleh:

|  |  |
| --- | --- |
| 1174073 | AINUL FILIANI |
| 1174070 | ARRIZAL FURQONA GIFARY |
| 1174079 | CHANDRA KIRANA POETRA |
| 1174075 | SEKAR JASMINE KHOIRUNNISA |

POLITEKNIK POS INDONESIA

chandrakiranapoetracendana@gmail.com

## A. Pendahuluan

Algoritma K-Nearest Neighbor (KNN) adalah salah satu metode yang menerapkan algoritma supervised (Mustakim, 2016), yang mana algoritma tersebut terbagi atas dua jenis yaitu supervised learning dengan unsupervised learning. Algoritma supervised learning bertujuan untuk mendapatkan pola baru sedangkan unsupervised learning untuk mendapatkan pola dalam sebuah data. Ketepatan algoritma KNN ditentukan oleh ada dan tidak adanya data yang tidak relevan, atau jika bobot fitur tersebut setara dengan relevansinya terhadap klasifikasi . Algoritma KNN adalah salah satu metode yang digunakan untuk analisis klasifikasi, namun beberapa dekade terakhir metode KNN juga digunakan untuk prediksi. Mencari jarak terdekat antara data yang akan dievaluasi dengan k tetangga (neighbor) terdekatnya dalam data pelatihan. Ruang ini dibagi menjadi kelompok-kelompok berdasarkan klasifikasi data training. Sebuah titik pada tempat ini ditandai dengan kelas c, dimana jika kelas c merupakan klasifikasi yang sangat banyak ditemui pada k buah tetangga terdekat titik tersebut. Untuk sistem kerja KNN data training di proyeksikan ke tempat berdimensi banyak, yang mana dari masing-masing dimensi mempersentasikan fitur dari data (Andi, 2017).

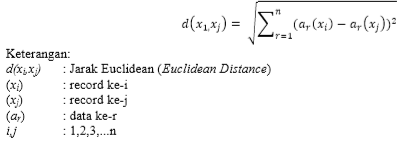
## B. Pengenalan K-Nearest Neighbor beserta implementasinya

K-Nearest Neighbor (KNN) adalah salah satu metode paling sederhana untuk memecahkan masalah klasifikasi (Adeniyi, Wei, & Yongquan, 2016). Algoritme ini sering digunakan untuk klasifikasi teks dan data (Samuel, Delima, & Rachmat, 2014). Pada metode ini dilakukan klasifikasi terhadap obyek berdasarkan data yang jaraknya paling dekat dengan obyek tersebut (Hardiyanto & Rahutomo, 2016). Klasifikasi teks menggunakan metode KNN akan menghasilkan nilai yang lebih optimal jika menggunakan rumus cosine similarity untuk pembobotan tiap-tiap kata pada dokumen teks yang akan diproses. Sebelum menghitung nilai (Nurjanah, Perdana and Fauzi, 2017)

Cosine similarity, harus melakukan tahapan dalam proses pembobotan kata yaitu tf, df, idf, tf-idf yang terdapat pada rumus persamaan 1 sampai persamaan 3. Setelah pembobotan kata selesai, selanjutnya yaitu menghitung kemiripan antar dokumen menggunakan rumus cosine similarity. Persamaan dari cosine similarity ditunjukkan pada Persamaan 4. (Nurjanah, Perdana and Fauzi, 2017)



**Gambar 1.2:** Rumus Euclidian



**Gambar 1.3:** Rumus Euclidian

Untuk menerapakan rumus KNN, setidaknya ada beberapa tahapan yang harus dilakukan untuk mendapatkan hasil hitungan akurat, berikut penerapan rumus K-nearest neighbor seperti yang diimplementasikan oleh (Suntoro, 2014)

1. Tahap pertama adalah membuat data training dan juga data testing yang akan kita coba olah, tabel dibawah ini akan digunakan untuk mengilustrasikan penggunaan rumus KNN ini.

|  |  |  |
| --- | --- | --- |
| X1 | X2 | Status |
| 8 | 4 | Baik |
| 4 | 5 | Jelek |
| 4 | 6 | Jelek |
| 7 | 7 | Baik |
| 5 | 6 | Jelek |
| 6 | 5 | Baik |

Tabel 1 : Data training yang akan dipakai untuk ilustrasi

|  |  |  |
| --- | --- | --- |
| X1 | X2 | Status |
| 7 | 4 | ? |

Tabel 2: Data testing yang akan dipakai untuk ilustrasi

1. Lalu tahap kedua adalah menentukan nilai parameter k, untuk kasus ini kita set K = 3
2. Tahap ketiga adalah menghitung jarak antara data training dan juga data testing dengan rumus Euclidean distance

|  |  |
| --- | --- |
| No | Proses |
| 1 | Ed1 = (8 − 7)2 + (4 − 4)2 = 1 |
| 2 | Ed2 = (4 − 7)2 + (5 − 4)2 = 10 |
| 3 | Ed3 = (4 − 7)2 + (6 − 4)2 = 13 |
| 4 | Ed4 = (7 − 7)2 + (7 − 4)2 = 9 |
| 5 | Ed5 = (5 − 7)2 + (6 − 4)2 = 8 |
| 6 | Ed6 = (6 − 7)2 + (5 − 4)2 = 2 |

Tabel 3: Proses perhitngan Euclidean distance

|  |  |  |  |
| --- | --- | --- | --- |
| X1 | X2 | Y | Jarak (7,4) |
| 8 | 4 | Baik | 1 |
| 4 | 5 | Jelek | 10 |
| 4 | 6 | Jelek | 13 |
| 7 | 7 | Baik | 9 |
| 5 | 6 | Jelek | 8 |
| 6 | 5 | Baik | 2 |

Tabel 4: Hasil perhitungan menggunakan Euclidean distance

1. Tahap keempat adalah mengurutkan hasil perhitungan jarak dari kecil hingga yang terbesar.

|  |  |  |  |
| --- | --- | --- | --- |
| X1 | X2 | Y | Jarak (7,4) |
| 8 | 4 | Baik | 1 |
| 6 | 5 | Baik | 2 |
| 5 | 6 | Jelek | 8 |
| 7 | 7 | Baik | 9 |
| 4 | 5 | Jelek | 10 |
| 4 | 6 | Jelek | 13 |

Tabel 5: Hasil pengurutan jarak dari kecil ke besar

1. Di tahap ke lima ini, kita akan menentukan data 7,4 pada apakah baik atau jelek klasifikasinya. Yang harus dilakukan adalah memilih tiga data terdekat terlebih dahulu (K=3). Jika dilihat di tabel dibawah ini terlihat dengan jelas bahwa ‘baik’ muncul lebih banyak dibandingkan dengan jelask, maka dari itu data testing dengan X = 7 dan Y = 4 masuk ke dalam kategori ‘Baik”

|  |  |  |  |
| --- | --- | --- | --- |
| X1 | X2 | Y | Jarak (7,4) |
| 8 | 4 | Baik | 1 |
| 6 | 5 | Baik | 2 |
| 5 | 6 | Jelek | 8 |

Jupyter Notebook adalah sebuah tools yang biasa di gunakan oleh kalangan orang yang biasa menggunakan bahasa python untuk menghasilkan sebuah document atau file yang berekstensi ipynb, di dalam document tersebut berisikan kode komputer dan rich text element seperti paragraf, persamaan matematik, gambar dan tautan (links). (Rossum, 1993)

Python adalah sebuah bahasa pemrograman yang interpreted, mudah di pahami, berorientasi objek dan dapat di gunakan untuk banyak platform seperti : web development, mobile apps dan desktop Apps. Python juga merupakan sebuah bahasa pemrograman yang interpreted, mudah di pahami, berorientasi objek, cocok untuk di gunakan pada program atau aplikasi yang rumit dan dapat di gunakan untuk banyak platform seperti : web development, mobile apps dan desktop Apps. (Setiabudidaya, 2018)

## C. Penutup

Algoritma K-Nearest Neighbor atau K-NN merupakan metode yang menerapkan supervised (Mustakim, 2016). Metode K-Nearest Neighbor ini juga merupakan salah satu dari sekian banyak metode pengklasifikasian data yang paling umum ditemui dan diimplementasikan karena K-Nearest Neighbor menggunakan basis data yang sudah ada sehingga dapat berjalan dengan cepat dan juga fleksibel (Safri, 2018). Selain itu, metode ini juga merupakan salah satu yang paling mendasar atau *basic* dalam hal pengklasifikasian pengenalan pola atau klasifikasi data. Metode ini berbasis konsep intuitif yang menyatakan bahwa data yang mirip seharusnya memang ditempatkan di tempat dengan data yang mirip juga. Karena kesederhanaan inilah KNN sering digunakan sebagai solusi mendasar untuk mengenali suatu pola (Olson dan Delen, 2008)

#### Referensi

1. Andi Bode. 2017. K-NEAREST NEIGHBOR DENGAN FEATURE SELECTION MENGGUNAKAN BACKWARD ELIMINATION UNTUK PREDIKSI HARGA KOMODITI KOPI ARABIKA. ILKOM Jurnal Ilmiah Volume 9 Nomor 2
2. Ni Luh Gede Pivin Suwirmayanti. 2017. Penerapan Metode K-Nearest Neighbor Untuk Sistem Rekomendasi Pemilihan Mobil. Techno.COM, Vol. 16, No. 2.
3. Suntoro, J. (2014) ‘DATA MINING Algoritme dan Implementasi Menggunakan Bahasa Pemrograman PHP’, 9(9), pp. 259–278.
4. Nurjanah, W. E., Perdana, R. S. and Fauzi, M. A. (2017) ‘Analisis Sentimen Terhadap Tayangan Televisi Berdasarkan Opini Masyarakat pada Media Sosial Twitter menggunakan Metode K-Nearest Neighbor dan Pembobotan Jumlah Retweet’, Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer (J-PTIIK) Universitas Brawijaya, 1(12), pp. 1750–1757. doi: 10.1074/jbc.M209498200.
5. Prajarini, D. (2016) ‘Perbandingan algoritma klasifikasi data mining untuk prediksi penyakit kulit’, Informatics Journal, 1(3), pp. 137–141.
6. Widaningsih, S. (2019) ‘Perbandingan Metode Data Mining Untuk Prediksi Nilai Dan Waktu Kelulusan Mahasiswa Prodi Teknik Informatika Dengan Algoritma C4,5, Naïve Bayes, Knn Dan Svm’, Jurnal Tekno Insentif, 13(1), pp. 16–25. doi: 10.36787/jti.v13i1.78.
7. Mustakim, G. O. (2016) ‘Algoritma K-Nearest Neighbor Classification’, *Jurnal Sains, Teknologi dan Industri*, 13(2), pp. 195–202. Available at: http://ejournal.uin-suska.ac.id/index.php/sitekin.
8. Safri, Y. F., Arifudin, R. and Muslim, M. A. (2018) ‘K-Nearest Neighbor and Naive Bayes Classifier Algorithm in Determining The Classification of Healthy Card Indonesia Giving to The Poor’, 5(1), pp. 9–18.
9. Rossum, G. Van (1993) ‘Python for Unix/C Programmers Copyright 1993 Guido van Rossum 1’, *Proc. of the Nluug Najaarsconferentie. Dutch Unix Users Group*, pp. 1–8.
10. Setiabudidaya, D. (2018) ‘PENGGUNAAN PIRANTI LUNAK JUPYTER NOTEBOOK DALAM’, pp. 2–5.
11. Yustanti, W. (2012) ‘Algoritma K-Nearest Neighbour untuk Memprediksi Harga Jual Tanah’, *Jurnal Matematika statistika dan komputasi*, 9(1), pp. 57–68.
12. Li, H. and Wu, G. (2014) *Map matching for taxi GPS data with extreme learning machine*, *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. doi: 10.1007/978-3-319-14717-8\_35.
13. S.Raikwal, J. and Saxena, K. (2012) ‘Performance Evaluation of SVM and K-Nearest Neighbor Algorithm over Medical Data set’, *International Journal of Computer Applications*, 50(14), pp. 35–39. doi: 10.5120/7842-1055.
14. Wettschereck, D. and Dietterich, T. G. (1994) ‘Locally adaptive nearest neighbor algorithms’, *Advances in Neural Information Processing Systems*, 6, pp. 184–191. doi: 10.1.1.55.499.
15. Olson, D. L. and Delen, D. (2008) *Advanced data mining techniques*, *Advanced Data Mining Techniques*. doi: 10.1007/978-3-540-76917-0.