PENERAPAN ALGORITMA NAÏVE BAYES UNTUK DETEKSI BAKTERI *E-COLI*

Laily Hermawanti
Program Studi Teknik informatika Fakultas Teknik Universitas Sultan Fatah (UNISFAT)
Jl. Diponegoro 1B Jogoloyo Demak Telpon (0291) 686227

Abstrak : Bakteri *e-coli* merupakan bakteri mikroskopik yang memiliki ukuran sangat kecil dan hanya bisa dilihat dengan mikroskop. Penelitian ini menggunakan algoritma *Naïve Bayes* untuk mendeteksi *bakteri e-coli*. Penelitian ini menghasilkan nilai akurasi untuk algoritma klasifikasi *Naive Bayes* senilai 98.18% dan nilai *Area Under Curve* (AUC) untuk algoritma *Naive Bayes* senilai 0.871, sehingga penelitian ini dalam mendeteksi bakteri *e-coli* menhasilkan hasil yang akurat.

Kata kunci : Bakteri *e-coli*, algoritma *Naïve Bayes*

PENDAHULUAN

Latar Belakang

Bakteri *e-coli* merupakan bakteri mikroskopik yang memiliki ukuran sangat kecil dan hanya bisa dilihat dengan mikroskop. Data di sini didapat perkembangan bakteri yang memiliki perkembangan sangat kecil yaitu dalam ukuran µm. Maka dari itu, perlu deteksi bakteri *e-coli.*

Rumusan Masalah

Penerapan algoritma *Naive Bayes* untuk deteksi bakteri *e-coli*, diharapkan menghasilkan tingkat akurasi yang tinggi.

Tujuan Penelitian

Menerapkan algoritma *Naive Bayes* untuk peningkatan akurasi deteksi bakteri *e-coli*.

TINJAUAN PUSTAKA

1. Bakteri *E-coli*

Bakteri *e-coli* merupakan bakteri mikroskopik yang memiliki ukuran sangat kecil dan hanya bisa dilihat dengan mikroskop.

Dataset bakteri *e-coli* yang digunakan adalah UCI *Dataset Repository*. Atribut-atribut bakteri *e-coli* pada UCI *Dataset Repository* adalah sequence name, mcg, gvh, lip, chg, aac, alm1 dan alm2.

2. Algoritma Naïve Bayes

Algoritma *naive bayes* memanfaatkan teori probabilitas yang dikemukakan oleh Thomas Bayes yaitu seorang ilmuwan dari Inggris (J. S. Badudu, 2001). Thomas Bayes memprediksi probabilitas di masa depan dengan berdasarkan pengalaman di masa sebelumnya.

Berdasarkan probabilitas dan teorema *bayesian* dengan asumsi bahwa setiap variabel bersifat bebas *(independence)* dan mengasumsikan bahwa keberadaan sebuah fitur *(variabel)* tidak ada kaitannya dengan keberadaan fitur *(variabel)* yang lain. *Naive bayes* adalah model penyederhanaan dari metode *bayes*. *Naive bayes* inilah yang digunakan di dalam *machine learning* sebagai metode untuk mendapatkan hipotesis untuk suatu keputusan. Dapat dihitung dengan persamaan di bawah ini:

Persamaan *naive bayes* (Oded Maimon and Lior Rokach, 2010)*:*

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

Keterangan:

1 = Class ke-i

$P(X|C_i)$ = Kemungkinan posterior X pada kondisi class C_i

$P(C_i)$ = Kemungkinan class C_i.

Klasifikasi *naive bayes* umumnya memiliki karakteristik sebagai berikut:

- Kokoh untuh titik *noise* yang diisolasi seperti titik yang dirata-ratakan ketika mengestimasi peluang bersyarat data. *Naive bayes classifier* dapat menangani *missing value* dengan mengabaikan contoh selama pembuatan model dan klasifikasi.
- Kokoh untuk atribut tidak relevan, jika X_i adalah atribut yang tidak relevan, maka $P(X_i|Y)$ menjadi hampir didistribusikan seragam. Peluang kelas bersyarat untuk X_i tidak berdampak pada keseluruhan perhitungan peluang *posterior*.
- Atribut yang dihubungkan dapat menurunkan *performance Naive bayes classifier* karena asumsi independen bersyarat tidak lagi menangani atribut tersebut.

Tahap-tahap algoritma *naive bayes* (Larose, 2005):

- Menyiapkan data training
- Setiap data direpresentasikan sebagai vektor berdimensi-n yaitu $X=(x_1,x_2,x_3,.....,x_n)$
- n adalah gambaran dari ukuran yang dibuat di test dari n atribut yaitu $A_1,A_2,A_3,.....,A_n$
- M adalah kumpulan kategori yaitu $C_1,C_2,C_3,.....C_m$.
- Diberikan data test *X* yang tidak diketahui kategorinya, maka *classifier* akan memprediksi bahwa *X* adalah

milik kategori dengan *posterior probability* tertinggi berdasarkan kondisi *X*.

- *Naive bayesian classifier* menandai bahwa test *X* yang tidak diketahui tadi ke kategori C_i jika dan hanya jika $P(C_i|X) > P(C_j|X)$ untuk $1 \leq j \leq m, j \neq i$
- Kemudian kita perlu memaksimalkan $P(C_i|X)$.
- $P(C_i|X) = \frac{P(X|C_i).P(C_i)}{P(X)}$
- Dimana x adalah nilai-nilai atribut dalam sampel *X* dan probabilitas $P(x_1|C_i)$, $P(x_2|C_i)$, ..., $P(x_n|C_i)$ dapat diperkirakan dari data training.

3. Alat Ukur Evaluasi dan Validasi

Evaluasi model merupakan tahapan yang juga dikerjakan dalam penelitian dengan tujuan untuk memperoleh informasi yang terdapat pada hasil klasifikasi terhadap kedua algoritma yang digunakan. Dalam RapidMiner hasil klasifikasiyang diperoleh dengan beberapa alat ukur yang tersedia di dalamnya, diantaranya adalah sebagai berikut:

Confusion matrix

Dalam penelitian ini dipilih alat ukur evaluasi berupa confusion matrix yang terdapat pada RapidMiner dengan tujuan untuk mempermudah dalam menganalisis performa algoritma karena *confusion matrix* memberikan informasi dalam bentuk angka sehingga dapat dihitung rasio keberhasilan klasifikasi.

Dalam kasus dengan dua klasifikasi data keluaran (Jiawei Han,2010) seperti contoh ya dan tidak, atau contoh lainnya, tiap kelas yang diprediksi memiliki empat kemungkinan keluaran yang berbeda, yaitu *true positive* (TP) dan *true negative* (TN) menunjukkan ketepatan klasifikasi. Jika prediksi keluaran bernilai *positive* sedangkan nilai aslinya adalah *positive* maka disebut dengan *false negative* (FN). Berikut ini pada tabel 3 disajikan bentuk *confusion matrix* seperti yang telah dijelaskan sebelumnya.

Tabel 1. Hasil yang Diperoleh dari Dua Kelas Prediksi (Jiawei Han, 2010)

Predicted Class			
Observed Class		Yes	No
	Yes	A *True Positive* – TP	b *False Negative* – FN
	No	C *False Positive* – FP	d *True Negative* – TN

Beberapa kegiatan yang dapat dilakukan dengan menggunakan data hasil klasifikasi dalam *confusion matrix* diantaranya:

- Menghitung nilai rata-rata keberhasilan klasifikasi *(overall success rate)* ke dalam kelas yang sesuai dengan cara membagi jumlah data yang terklasifikasi dengan benar, dengan seluruh data yang diklasifikasi
- Selain itu dilakukan pula penghitungan persentase kelas *positive* (*true positive* dan *false positive*) yang diperoleh dalam klasifikasi, yang disebut dengan lift chart.
- Lift chart terkait erat dengan sebuah teknik dalam mengevaluasi skema data mining yang dikenal dengan ROC *(Receiver Operating Characteristic)* yang berfungsi mengekspresikan persentase jumlah proporsi *positive* dan *negative* yang diperoleh.
- Recall Precision berfungsi menghitung persentase *false positive* dan *false negative* untuk menentukan informasi di dalamnya.

Setelah data uji dimasukkan ke dalam *confusion matrix*, hitung nilai-nilai yang telah dimasukkan tersebut untuk dihitung jumlah *sensitivity*, *specificity*, *precision* dan *accuracy*. *Sensitivity* digunakan untuk membandingkan jumlah *true positives* terhadap jumlah tupel yang *positives* sedangkan *specificity* adalah perbandingan jumlah *true negatives* terhadap jumlah tupel yang *negatives*. Untuk menghitung digunakan persamaan di bawah ini (Jiawei Han,2010)

$$Accuracy = \frac{TN + TP}{TP + FP + FN + TN}$$

$$Precision = \frac{TP}{TP + FN}$$

$$Sensitivity = \frac{TN}{TN + FP}$$

$$Sensitivity = \frac{TN}{TN + FP}$$

$$Recall = \frac{FP}{FP + TN}$$

Dimana :

t_pos = Jumlah *true positives*

t_neg = Jumlah *true negatives*

pos = Jumlah tupel positif

neg = Jumlah tupel negatif

f_pos = Jumlah *false positives*

ROC (***Receiver Operating Characteristic)*** **Curve**

Kurva ROC merupakan ilustrasi grafis dari kemampuan diskriminan (Jiawei Han,2010)

Biasanya ROC diterapkan untuk masalah klasifikasi. Apabila garis diagonal diatas, maka klasifikasi baik sedangkan garis diagonal dibawah, maka klasifikasi buruk.

METODE PENELITIAN

Metode penelitian pada penelitian ini adalah penelitian eksperimen dengan urutan sebagai berikut:

1. Pengumpulan Data (*Data Gathering*)

   Tahap ini dilakukan sebagai langkah awal dari suatu penelitian.Untuk memperoleh data yang benar-benar akurat, maka penentuan jenis dan sumber data sangatlah penting. *Dataset* bakteri *e-coli* yang digunakan adalah 3 UCI *Dataset Repository.*
2. Pengolahan Awal Data (Data *Pre-processing*)

   Data yang didapat diolah untuk mendapatkan atribut yang relevan dan sesuai.
3. Metode Yang Diusulkan (*Proposed Model/Method*)

   Tahap ini akan membahas metode yang akan digunakan untuk penelitian nanti. Dalam penelitian, setelah dilakukan studi literatur dari buku dan jurnal, ditemukan bahwa salah satu cara yang dapat membantu mengklasifikasikan diagnosis penyakit kanker payudara dari UCI Dataset Repository dengan menggunakan algoritma *Naïve Bayes.*
4. Eksperimen dan Pengujian Model/Metode (*Method Test and Experimen*)

   Metode eksperimen dan pengujian ini dengan algoritma *Naïve Bayes.*
5. Evaluasi dan Validasi Hasil (*Result Evaluation*)

   Tahap ini akan membahas tentang hasil evaluasi dari eksperimen yang telah dilakukan. Pengujian hasil implementasi dengan menggunakan model *ROC CURVE (AUC).* Hasil pengujian yang di dapat dari metode *Confusion Matrix* adalah akurasi.

EVALUASI DAN VALIDASI PADA ALGORITMA NAÏVE BAYES

Penelitian ini evaluasi dan validasi hasil menggunakan *confusion matrix* (accuracy) dan ROC *Curve*.

Confusion Matrix

Evaluasi dengan *confusion matrix* menggunakan tabel matrix seperti di bawah ini

Tabel 2. Konversi Naive *Bayes* ke *confusion matrix*

	cp	im
cp (Cytoplasm)	143	4
im (inner membrane without signal sequence)	0	73

Kemudian masukkan nilai yang ada di dalam *confusion matrix* ke dalam persamaan di bawah ini:

Accuracy = 143+73 / (143+4+0+73) = 216 / 220 = 0.9818

Gambar *Confusion Matrix* algoritma *Naive Bayes* menggunakan RapidMiner dapat dilihat pada Gambar 1.

accuracy: 98.18% +/- 3.02% (mikro: 98.18%)

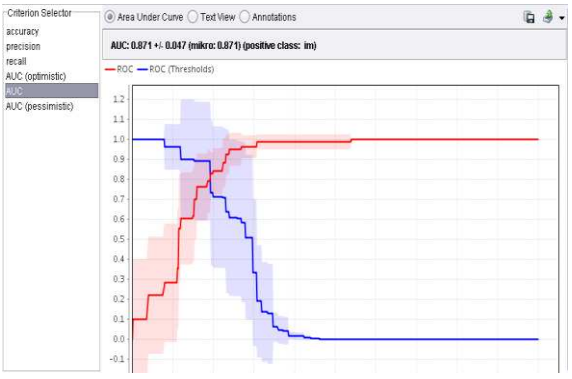
	true cp	true im	class precision
pred. cp	143	4	97.28%
pred. im	0	73	100.00%
class recall	100.00%	94.81%	

Gambar 1 : *Confusion Matrix* algoritma *Naive Bayes* menggunakan RapidMiner

Berdasarkan hasil perhitungan, tingkat akurasi menggunakan algoritma *Naive Baye*s sebesar 98.18%.

Kurva ROC (*Receiver Operating Characteristic*)

Grafik ROC dengan nilai AUC (*Area Under Curve*) dengan algoritma *naive bayes* sebesar 0.548 dan nilai AUC yang menggunakan algoritma *naive bayes* berbasis *AdaBoost* mencapai angka 0.871 seperti terlihat pada gambar di bawah ini. Akurasi AUC dikatakan sempurna apabila nilai AUC mencapai 1.000 dan akurasinya buruk jika nilai AUC dibawah 0.500. Nilai AUC dalam grafik ROC dengan Algoritma *Naive Bayes* dapat dilihat pada Gambar 2.



Gambar 2. Nilai AUC dalam grafik ROC dengan Algoritma *Naive Bayes*

Analisa dan Validasi Model

Penggunaan algoritma *Naive Bayes* akurasinya tinggi dalam mendeteksi bakteri *e-coli.* Seperti tabel di bawah ini:

Tabel 3. Analisa hasil evaluasi dan validasi

	Algoritma Naive Bayes
Accuracy	98.18%
AUC	0.871

Tabel 3 menunjukkan analisa evaluasi hasil *naive bayes* nilai akurasi sebesar 98.18% dan nilai AUC sebesar 0.871.

KESIMPULAN DAN SARAN

Kesimpulan

Setelah dilakukan evaluasi dengan algoritma *Naive Bayes*, hasil akurasinya tinggi. Nilai akurasi untuk algoritma klasifikasi *Naive Bayes* senilai 98.18% dan nilai AUC untuk algoritma *Naive Bayes* senilai 0.871.

Saran

Untuk menambah akurasi algoritma, akan lebih baik apabila dioptimasi dengan algoritma seperti *Particle Swarm optimization* (PSO), *Genetic Algorithm* (GA) ataupun algoritma-algoritma optimasi yang lain pada algoritma *Naïve Bayes* supaya didapatkan hasil akurasi yang lebih tinggi.

DAFTAR PUSTAKA

J. S. Badudu, *Kamus Umum Bahasa Indonesia*. Jakarta: Pustaka Sinar Harapan, 2001.

Oded Maimon and Lior Rokach, *Data Mining and Knowledge Discovery Handbook*, Second Edition ed., Oded Maimon and Lior Rokach, Eds. London, New York: Springer, 2010.

D. T Larose, *Discovering Knowledge in Data*.: New Jersey: John Willey, 2005.

Jiawei Han and Micheline Kamber, *Data Mining Concepts And Techniques*, 1st ed., Asma Stephan, America: Diane Cerra, 2007.