ISNN : 2338-4697
e-ISSN : 2579-3322

Perbandingan Algoritma untuk Analisis Sentimen Terhadap Google *Play Store* Menggunakan *Machine Learning*

Hery Oktafiandi [1)], Winarnie [2)], Sayyid M. Raziq Olajuwon[3)]

[1)] Teknologi Rekayasa Perangkat Lunak Politeknik Sawunggalih Aji
[2,3)] Magister Teknologi Informasi, Universitas Amikom Yogyakarta
[1)]Jl Wismoaji no 38 Kutoarjo, Purworejo 54251
[2,3)]Ringroad Utara Nringin Condong Catur Kota Yogyakarta 55283
Email : heryokta@gmail.com[1)], winzahwa@gmail.com[2] ,) olajuwonda@gmail.com[3)]

Abstrak

Banyaknya aplikasi yang tersedia pada *google playstore* memudahkan para pengguna untuk memiilih aplikasi yang sesuai dengan kebutuhannya. Pengguna aplikasi pada *google playstore* memiliki sedikit kesulitan dalam memilih aplikasi mana saja yang baik performancenya karena terlalu banyaknya pilihan aplikasi yang tersedia. Para pengembang aplikasi menyediakan kolom komentar untuk *review* para pengguna aplikasi. Dengan *review* ini para calon pengguna aplikasi dapat memutuskan apakah menggunakan atau tidak, sedangkan untuk pengembang, *review* sangat dibutuhkan karena dapat melihat pencapaian kinerja dari aplikasi yang telah dibuat. Pada penelitian ini dilakukan untuk membandingkan tiga algorima *machine learning* yaitu : *naives bayes*, *k-nearest* neighbors dan *radom fores*t untuk membandingkan nilai akurasi pada setiap algoritma berdasarkan sentimen pengguna. Pada penelitian ini dataset didapat dari *scraping* langsung dari aplikasi yang tersedia pada *google play*. Data kemudian diseleksi dan diberi label/klas. Pada penelitian ini dilakuakan percobaan dengan menggunakan jumlah dataset yang berbeda yaitu 40 dataset, 100 dataset dan 1000 dataset. Hasil akurasi yang didapar pada penggunaan dataset 1000 data diperoleh hasil akurasi untuk algoritma *naïve bayes* sebesar 79%, algoritma *k nearest neighbors* sebesar 77% dan algoritma *random forest* sebesar 75%.

Kata kunci: *naive bayes*, k-*nearest neighbors, random forest, google play, machine learning*

Abstract

A lot of applications available on the Google Playstore makes it easy for users to choose the application that suits their needs. Application users on the Google Playstore have a little difficulty in choosing which applications have good performance because there are too many application options available. The application developers provide a comment field for user reviews of the application. With this review, prospective application users can decide whether to use it or not, while for developers, reviews are needed because they can see the performance achievements of the applications that have been made. This research was conducted to compare three machine learning algorithms, namely: naives bayes, k-nearest neighbors and radom forest to compare the accuracy values of each algorithm based on user sentiment. In this study, the dataset was obtained from scraping directly from applications available on Google Play. The data is then selected and labeled/classed. In this study, experiments were carried out using a number of different datasets, namely 40 datasets, 100 datasets and 1000 datasets. The accuracy results obtained from the use of a dataset of 1000 data obtained that the accuracy for the naïve Bayes algorithm is 79%, the k nearest neighbors algorithm is 77% and the random forest algorithm is 75%.

Keywords: *naive bayes, k-nearest neighbors, random forest, google play, machine learning*

1. PENDAHULUAN

Banyaknya aplikasi yang tersedia memudahkan para pengguna aplikasi memilih aplikasi yang sesuai dengan kebutuhan tetapi sebaliknya hal tersebut juga membuat bimbang para pengguna karena terlalu banyak pilihan aplikasi yang tersedia. Untuk mengatasi masalah tersebut pertama-tama pengguna harus mengetahui fungsi dari aplikasi yang akan digunakan kemudian melihat komentar *review* dari pengguna aplikasi sebelumnya kemudian baru bisa memutuskan apakah akan menggunakan atau tidak. Dalam kolom *review* yang diperhatikan oleh calon pengguna biasanya adalah kolom *rating* dan kolom komentar. Dengan banyaknya *review* dari pengguna sebelumnya menjadikan tantangan untuk membaca setiap komentar dan dari sisi pengembang banyaknya

ISNN : 2338-4697
e-ISSN : 2579-3322

review juga menyulitkan jika harus dilihat satu persatu. Berdasarkan permasalahan tersebut maka dibutuhkan sebuah sentiment analisis yang dapat mengolah sejumlah *review* untuk memperoleh informasi yang bernilai positif atau negatif.

Metode yang digunakan untuk mengklasifikasikan *review* adalah dengan menggunakan algoritma *naïve bayes*, *k-nearest neighbors* dan *random forest*. Penelitian mengenai analisis sentiment pada *google play* pernah dilakukan diantaranya oleh Wahyudi dkk [1]yang melakukan penelitian menganalisis *review* pengguna aplikasi Grab pada *Google Play Store* menggunakan metode *support vector machine (SVM)*. Penelitian lainnya dilakukan oleh Rahman dkk [2] yang melakukan penelitian untuk meningkatkan akurasi penelitian sebelumnya dengan menggunakan algoritma *naïve bayes* dan algoritma *genetic*.

Pada penelitian lain dilakuakan oleh Putri dkk [3] yang melakukan penelitian menganalisis sentimen *review* tentang aplikasi *marketplace* yang semakin kompetitif dengan menggunakan metode klasifikasi *supervised learning* di antaranya *support vector machine* (SVM), naïve bayes, dan *logistic regression* yang mana metode klasifikasi tersebut kemudian dibandingkan kinerja performa klasifikasinya untuk mendapatkan metode terbaik untuk mengklasifikasikan kelas sentiment.

Dari beberapa penelitian yang pernah dilakukan berkaitan dengan sentiment analisis pada *google playstore* yang pernah diteliti sebelumnya menggunakan metode algoritma yang berbeda dengan yang akan penulis teliti. Dari hasil *review* beberapa penelitian yang sudah ada maka dalam penelitian ini dilakukan pengujian dengan 3 algoritma yang berbeda yaitu *naive bayes*, *k-nerest neighbors* dan *random forest* untuk mengetahui tingkat akurasi. Dalam penelitian ini dilakukan 3 percobaan yaitu dengan menggunakan jumlah dataset yang berbeda

2. TINJAUAN PUSTAKA

2.1 Naïve Bayes

Algoritma yang mempelajari probabilitas suatu objek dengan ciri-ciri tertentu yang termasuk dalam kelompok/kelas tertentu. Intinyaa algoritma ini adalah pengklasifikasi probabilistic p(c), yaitu probabilitas kelas jika diketahui I dokumen. *Naive bayes* menganggap sebuah dokumen sebagai kumpulan dari kata-kata yang menyusun dokumen tersebut, dan tidak memperhatikan urutan kemunculuan kata pada dokumen perhitungan probabilitasnya dapat dianggap sebagai hasil perkalian dari probabilitas kemunculan kata-kata pada dokumen[4]

Probabilitas sebuah dokumen berada di kelas c dapat dihitung menggunakan rumus posterior probability sebagai berikut:

$$P\ cj\ wi = P\ cj\ xP\ w1\ cj\ x\ \dots\ xP\ wk\ cj$$ [4] (1)

Keterangan :
P(CjWi) = *Posterior Probability* adalah peluang kelas C.
P(Cj) = *Prior Probability* adalah peluang kemunculan tiap kelas.
P(Wi|Cj) = *Conditional Probability (Likelihood)* adalah peluang kata-kata pada kelas (dokumen) tertentu.

2.2 K-Nearest Neighbors

Algoritma K-NN adalah metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasi sebelumnya. Tahapan dalam proses ini adalah memprediksi menggunakan dataset dan melakukan perhitungan dengan *confusio matrix.*

Untuk menghitung jarak antara dua titik, algoritma KNN menggunakan metode *Euclidean Distance* [5] Metode Euclidean distance [5] menggunakan formula berikut:

$$(x, y) = \sqrt{\sum}\ (x1i - y1i)2 + (x2i - y2i)2 + .$$ (2)

Keterangan:
(x, y) : jarak antara x dan y
x : data yang terklasifikasi
y : data sekitar
I : jumlah fitur

2.3 Random Forest

Metode *Random Forest* (RF) adalah pengembangann dari metode *Classification* and *Regression Tree* (CART), yaitu dengan menerapkan metode *bootstrap aggregating* (bagging) dan *random feature selection* [6]. *Random Forest* merupakan salah satu metode yang digunakan untuk klasifikasi dengan membangun banyak pohon klasifikasi. Metode ini dapat meningkatkan hasil akurasi, dengan cara membangkitkan simpul anak untuk setiap *node*

ISNN : 2338-4697
e-ISSN : 2579-3322

(simpul diatasnya) dan dilakukan pemilihan secara acak.

2.4 Pre-processing

Pre processing adalah sebuah proses pengolahan dataset /data mentah untuk dijadikan data yang bisa dipakai untuk proses selanjutnya sampai ke proses pengujian [7]. Dalam *pre processing* ada beberapa proses yaitu :

1. *Filtering* merupakan proses pengolahan data dengan menghilangkan variabel-variabel yang tidak dibutuhkan
2. *Case folding* merupakan proses pengolahan data dari yang menggunakan huruf besar diubah menjadi huruf kecil.
3. *Tokenizing* merupakan proses pengolahan data dengan mengubah kalimat dalam dokumen menjadi kata atau *token*. Dalam proses *tokenizing* ini juga menghilangkan *mantion*, *hastag*, karakter, *punctuation*, angka dan spasi berlebih.
4. Stemming adalah proses pengolahan data dari kata berimbuhan diubah menjadi kata dasar.
5. *Stopword* adalah proses pengolaha data dimana menghilangkan kata yang berulang

2.5 Confosius Matrix

Metode *Confusion Matrix* adalah pengukuran masalah untuk performa *machine learning* dimana keluarannya merupakan dua kelas atau lebih dari itu. *Confusion Matrix* juga merupakan kombinasi dari 4 tabel dari nilai prediksi dan nilai aktual yang berbeda-beda. Ada 4 hasil yang merupakan istilah proses representasi yaitu TP (*True Positive*), FP (*False Positive*), FN (*False Negative*), TN (*True Negative*).

Confusion Matrix merupakan alat untuk evaluasi visual yang digunakan dalam pembelajaran mesin. Prediksi kolom *Confusion Matix* adalah mewakili hasil prediksi kelas. [8]

Ini merupakan 4 kombinasi table nilai aktual dan nilai prediksi.

	Nilai Aktual	
	Positif (1)	Negatf (0)
	TP	FP
	FN	TN

Gambar 1. *Tabel Confosius Matrix*

Hal ini berguna untuk mengukur *Accuracy, Precision, Recall* dan *Specificity*

3. METODE PENELITIAN

Dalam melakukan penelitian ini dilakukan beberapa tahap proses percobaan , yaitu :

1 Pengumpulan data

Data diperoleh dari *google playstore* pada aplikasi menonton film netflix, *google translate* dan tiktok . Jenis data yang diambil berupa *text* dengan jumlah *review* yang diambil sebanyak 40 *review*, 100 review dan 1000 review. Data diambil dengan *menscapping review* pada ketiga aplikasi tersebut.

2. Pelabelan / memberi nilai klas

Setelah data discrapping selanjutnya diberi label nilai positif atau negatif. Data juga dipecah menjadi data *training* dan data *test.*

3 Pre processing

Pre-processing adalah proses perubahan data agar bisa diolah dalam pengujian. [7]Dalam tahap ini dilakuakan proses *case folding* (merubah huruf kecil), *tokenizing* (merubah kalimat menjadi *token*), *stemming* (merubah kata imbuhan menjadi kata dasar), dan *stopword* (menghilangkan kata yang berulang)

4 Evaluasi

Dalam tahap evalusi merupakan proses perhitungan bobot tiap *term* pada setiap dokumen sehingga diketahui ketersediaan dan kemiripan suatu *term* dalam dokumen. Proses evaluasi dimulai dengan menghitung *vector*, kemudian mengubah *vector* menjadi *term* dan terakhir menggunakan perhitungan TF-IDF[9] Setelah tahap TF-IDF maka dilakukan pengujian dengan menggunakan *confosius matrix* .[10]

ISNN : 2338-4697
e-ISSN : 2579-3322

4. PEMBAHASAN

Pengumpulan dataset ini menggunakan *tools* yang ada pada bahasa pemrograman *python* yaitu *web scrapper*, dataset diambil dari 3 aplikasi yang berbeda yaitu aplikasi menonton Netflix, aplikasi *google translate* dan aplikasi tiktok. Pada aplikasi netflik dataset yang diambil sejumlah 40 data, pada aplikasi *google translate* diambil data sejumlah 100 data dan pada aplikasi tiktok diambil data sejumlah 1000 data. Dataset kemudian diberi pelabelan atau klas, label yang diberikan adalah berupa sentiment positif dan sentiment negatif, Pelabelan dilakukan dengan menggunakan library yang ada di *python*. Proses selanjutnya adalah proses *pre processing*, dalam tahap ini terdapat proses *case folding* (merubah huruf kecil), *tokenizing* (merubah kalimat menjadi token), *stemming* (merubah kata imbuhan menjadi kata dasar), dan s*topword* pada penelitian ini menggunakan bahasa Indonesia (menghilangkan kata yang berulang).

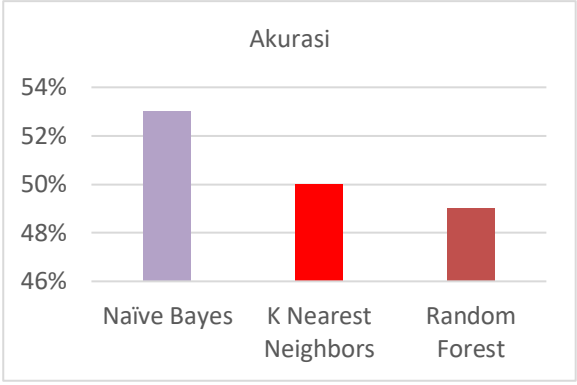
Setelah proses *pre processing* selanjutnya ke tahap evaluasi dengan menggunakan proses tf-idf. Dalam proses ini adalah pehitungan bobot tiap *term* dalam dokumen untuk mencari ketersediaan dan kemiripan term atau kata pada dokumen. Setelah proses TF-IDF maka hasil yang didapat akan diolah pada percobaan pengujian algoritma dengan menggunakan *confosius matrik*.

4.1 Percobaan Pertama

Pada percobaan pertama menggunakan dataset sejumlah 40 data yang terdiri dari data training sejumlah 28 data dan data test sejumlah 12 data. Hasil dari percobaan 1 diperoleh hasilseperti pada tabel 1 dan gambar 2.

Tabel 1 Hasil Percobaan 1

Algoritma	Akurasi
Naïve Bayes	53%
K-Nearest Neighbors	50%
Random Forest	49%



Gambar 2. *Grafik Hasil Percobaan 1*

Dilihat dari hasil percobaan pertama nilai akurasi tertinggi ada padaalgoritma naïve bayes sebesar 53%, k-*nearest neighbors* sebesar 50% dan *random forest* sebesar 49%. Pada percobaan pertama nilai akurasi yang didapat masih bernilai kecil.

4.2 Percobaan Kedua

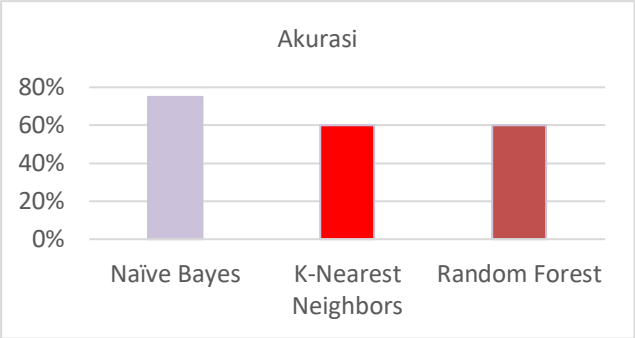
Pada percobaan kedua ini menggunakan dataset sebanyak 100 . dengan terdiri data training sebanyak 30 data dan data test sebanyak 70 data. Hasil dari percobaan kedua dapat dilihat pada table 2 dan gambar 3. Dalam percobaan kedua dataset dibagi menjadi 30%.

Percobaan dilakukan dengan mengubah jumlah dataset. Pada percobaan pertama menggunakan data sebanyak 40 data dan dipecah menjadi 30%, pada percobaan kedua data ditambah menjadi 100 data .

Tabel 2 . Hasil Percobaan 2

Algoritma	Akurasi
Naïve Bayes	75%
K-Nearest Neighbors	60%
Random Forest	60%

ISNN : 2338-4697
e-ISSN : 2579-3322



Gambar 3. *Grafik percobaan 2*

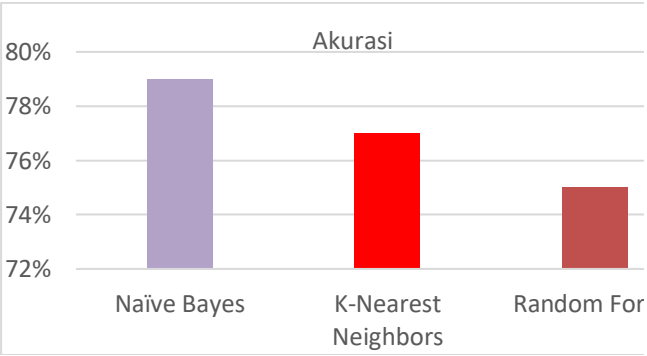
Dari hasil percobaan kedua didapat hasil akurasi terbesar pada algoritma *naïve bayes* sebesar 75% kemudian algoritma *k-nearest neighbors* sdan algoritma *random forest* sebesar 60%., dilihat dari hasil percobaan 2 ada peningkatan nilai akurasi pada ketiga algoritma dibandingkan dengan percobaan 1.

4.3 Percobaan Ketiga

Pada percobaan ketiga dilakukan dengan menggunakan dataset sebanyak 1000 data yang terdiri data training 250 data dan data test sebanyak 750 data.. Hasil percobaan ketiga dapat dilihat pada table 3 dan gambar 4

Tabel 3 Hasil Percobaan 3

Algoritma	Akurasi
Naïve Bayes	79%
K-Nearest Neighbors	77%
Random Forest	75%



Gambar 4. *Grafik Percobaan 3*

Dari tabel 3 dan gambar 3 dapat dilihat hasil akurasi algoritma *naïve bayes* sebesar 79%, algoritma *k-nearest neighbors* sebesar 77% dan algoritma *random fores*t sebesar 75%. Dari hasil ketiga percobaan dapat dilihat adanya peningkatan nilai akurasi pada setiap algoritma. Pada algoritma *naïve bayes* terjadi peningkatan akurasi sebesar 19,6 %, algoritma *k nearest neighbors* terjadi peningkatan akurasi sebesar 21 % dan algoritma *random forest* sebesar 20%. Prosentasi peningkatan nilai akurasi dihitung dengan membandingkan hasil percobaan pertama dan percobaan ketiga.

5. KESIMPULAN

Dari hasil ketiga percobaan yang telah dilakukan pada sejumlah dataset yang berbeda-beda maka didapat hasil nilai akurasi tertinggi dengan nilai 79 % pada algoritma *naïve bayes* dengan jumlah dataset sebanyak 1000 data pada percobaan 3. Sedangkan dengan algoritma *k nearest neighbors* diperoleh nilai sebesar 77 %, dan algoritma *random forest* sebesar 75%. Dari hasil ketiga percobaan dapat dilihat adanya peningkatan nilai akurasi pada setiap algoritma. Pada algoritma *naïve bayes* terjadi peningkatan akurasi sebesar 19,6 %, algoritma *k nearest neighbors* terjadi peningkatan akurasi sebesar 21 % dan algoritma *random forest* sebesar 20%. Prosentasi peningkatan nilai akurasi dihitung dengan membandingkan hasil percobaan pertama dan percobaan ketiga.

Dengan semakin banyak dataset yang digunakan maka akurasi akan semakin baik karena semakin banyak data yang dipakai sebagai data training sehinggga pemodelan menjadi lebih terlatih. Ketiga algoritma yang digunakan dalam percobaan ini bisa dipakai untuk analisis sentiment pada *google playstore*.

Untuk penelitian selanjutnya bisa menggunakan optomasi pada algoritma sehingga nilai akurasi yang didapat bisa lebih baik lagi.

DAFTAR PUSTAKA

[1] R. Wahyudi and G. Kusumawardana, “Analisis Sentimen pada Aplikasi Grab di Google Play Store Menggunakan Support Vector Machine,” *J. Inform.*, vol. 8, no. 2, pp. 200–207, 2021, doi: 10.31294/ji.v8i2.9681.

[2] A. Rahman, E. Utami, and S. Sudarmawan,

ISNN : 2338-4697
e-ISSN : 2579-3322

"Sentimen Analisis Terhadap Aplikasi pada Google Playstore Menggunakan Algoritma Naïve Bayes dan Algoritma Genetika," *J. Komtika (Komputasi dan Inform.*, vol. 5, no. 1, pp. 60–71, 2021, doi: 10.31603/komtika.v5i1.5188.

[3] M. I. Putri and I. Kharisudin, "Analisis Sentimen Pengguna Aplikasi Marketplace TokopediaPada Situs Google Play Menggunakan Metode SupportVector Machine (SVM), Naïve Bayes, dan LogisticRegression," *Prism. Pros. Semin. Nas. Mat.*, vol. 5, pp. 759–766, 2022, [Online]. Available: https://journal.unnes.ac.id/sju/index.php/prisma/

[4] Azhar, S. U. Masruroh, L. K. Wardhani, and Okfalisa, "Perbandingan Kinerja Algoritma Naive Bayes Dan K-Nn Pendekatan Lexicon Pada Analisis Sentimen Di Media," *Pros. Semin. Nas. Fis. Univ. Riau IV*, no. September, pp. 978–979, 2019.

[5] A. M. Tamrizal and A. Yaqin, "Perbandingan Algoritma Naïve Bayes , K-Nearest Neighbors dan Random Forest untuk Klasifikasi Sentimen Terhadap BPJS Kesehatan pada Media Twitter," vol. 12, no. 1, pp. 1–10, 2022.

[6] G. A. Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest," *CogITo Smart J.*, vol. 6, no. 2, p. 167, 2020, doi: 10.31154/cogito.v6i2.270.167-178.

[7] A. Wandani, F. Fauziah, and A. Andrianingsih, "Sentimen Analisis Pengguna Twitter pada Event Flash Sale Menggunakan Algoritma K-NN, Random Forest, dan Naive Bayes," *J-SAKTI (Jurnal Sains Komput. dan Inform.*, vol. 5, no. 2, pp. 651–665, 2021.

[8] Y. Kustiyahningsih, "Feature Selection and K-nearest Neighbor for Diagnosis Cow Disease," *Int. J. Sci. Eng. Inf. Technol.*, vol. 5, no. 02, pp. 249–253, 2021, doi: 10.21107/ijseit.v5i02.10218.

[9] S. W. Irianandа *et al.*, "Analisis Sentimen Dan Analisis Data Eksploratif Ulasan," no. Ciastech, pp. 473–482, 2021.

[10] R. Putri Fitrianti, A. Kurniawati, D. Agusten, J. Sistem Informasi, and F. Ilmu Komputer dan Teknologi Informasi, "Implementasi Algoritma K-Nearest Neighbor Terhadap Analisis Sentimen Review Restoran Dengan Teks Bahasa Indonesia," *Semin. Nas. Apl. Teknol. Inf.*, pp. 1907–5022, 2019.