Klasifikasi Spam SMS Menggunakan *Naïve Bayes Classifier* dan *K-Nearest Neighbors*

Adnan Sauddin*
Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, adnan.sauddin@uin-alauddin.ac.id

Try Azisah Nurman
Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, try.azisah@uin-alauddin.ac.id

Nur Aeni
Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, nuraeniayatullah@gmail.com

Sadem Rahayu Sudarta
Program Studi Matematika, Universitas Islam Negeri Alauddin Makassar, sademrahayusudarta@gmail.com

**Corresponding Author*

ABSTRAK, Penelitian ini membahas mengenai klasifikasi dataset Spam SMS. Indonesia menempati posisi ke-19 untuk SMS spam terbanyak di dunia. Banyak tindak kejahatan penipuan yang menimbulkan kerugian bagi pengguna berasal dari SMS spam. Klasifikasi spam SMS dapat dilakukan dengan menggunakan metode *machine learning* yaitu *Naïve Bayes Classifier (NBC)* dan *K-Nearest Neighbors (KNN)* dengan menggunakan pembobotan kata *term frequency*. Penelitian ini bertujuan untuk untuk mengetahui performa klasifikasi spam SMS menggunakan algoritma *NBC* dan algoritma *KNN*. Penelitian ini menunjukkan bahwa akurasi klasifikasi menggunakan metode *Naïve Bayes Classifier* lebih besar yaitu 98,3% dibandingkan metode *K-Nearest Neighbors* dengan akurasi 95,1% dengan rasio akurasi sebesar 1,033 yang menunjukkan metode *Naïve Bayes Classifier* memiliki performa yang lebih baik.

***Kata Kunci**: NBC, KNN, Klasifikasi, training dan testing, Spam SMS, akurasi, term frequency*

1. PENDAHULUAN

Hasil riset Truecaller menunjukkan bahwa Indonesia menempati posisi ke-19 untuk SMS spam terbanyak di dunia, dengan rata-rata orang menerima 6 pesan spam per bulan. Pesan spam yang banyak beredar yaitu spam yang menawarkan layanan promosi/iklan, informasi perbankan, diskon toko dan pesan-pesan lain yang merujuk pada suatu instansi untuk melakukan tindak kejahatan penipuan yang menimbulkan kerugian bagi pengguna.

Salah satu upaya pemerintah untuk mengatasi masalah SMS (*Short Message Service*) spam adalah dengan memberlakuan peraturan menteri Kominfo Nomor 14 Tahun 2017 tentang registrasi kartu prabayar yang bertujuan agar terdapat data yang valid bagi pengguna jasa telekomunikasi, serta dapat mengatasi modus kejahatan juga memberikan perlindungan terhadap kepentingan pelanggan jasa telekomunikasi [1]. Namun kebijakan tersebut tidak dapat mengatasi permasalahan secara utuh karena terdapat faktor lain yang menjadi penyebab spam SMS yaitu pemberian izin oleh pengguna pada aplikasi-aplikasi berbahaya untuk mengakses data penggunanya sehingga SMS dapat dikirimkan kepada pengguna yang saling kenal maupun tidak saling kenal. Oleh karena itu dibutuhkan *filter* atau penyaringan untuk mengklasifikasikan SMS spam yang didasarkan pada atribut atau kategori tertentu yang telah didefinisikan.

Penyaringan terhadap informasi yang diterima, salah satunya informasi yang berasal dari SMS dengan mengklasifikasikan spam SMS menggunakan algoritma *Naïve Bayes Classifier* karena sifatnya yang historical (mampu mengingat kondisi sebelumnya) tersebut sangat tepat digunakan untuk *spam filtering* [2]. Ketika dibandingkan antara algoritma *Naïve Bayes* dengan algoritma lainnya diperoleh bahwa algoritma *Naïve Bayes* cukup baik dalam melakukan sebuah prediksi pada mengklasifikasikan pesan teks seperti penelitian yang dilakukan oleh [3] dan [4]. Namun penelitian sebelumnya belum ada yang membandingkan algoritma *Naïve Bayes* dan *K-Nearest Neighbors* maka pada penelitian ini akan membandingkan data yang telah matang dengan klasifikasi pada data *learning* terhadap data *testing* untuk melihat keakuratan hasil prediksinya menggunakan *Naïve Bayes Classifier* dan menggunakan *K-Nearest Neighbors*.

Berdasarkan uraian tersebut maka penulis tertarik melakukan penelitian dengan judul "Klasifikasi Spam SMS Menggunakan *Naïve Bayes Classifier* dan *K-Nearest Neighbors*".

2. TINJAUAN PUSTAKA

Data *mining* adalah suatu proses untuk mendeteksi pola suatu data serta informasi menarik yang dapat ditemukan pada data yang ukurannya sangat besar. Data dapat berasal dari suatu database, situs web, data pada *warehouse*, data pada *repository*, atau sumbel lainnya [5].

Data *mining* dibutuhkan ketika mengelola data agar diperoleh informasi/ pengetahuan yang bermanfaat. Peranan data *mining* terbagi menjadi lima diantaranya klasifikasi, asosiasi, prediksi, *clustering*, dan estimasi. Untuk melakukan klasifikasi digunakan algoritma *Naïve Bayes, k-Nearest Neighbos (k-NN), ID3, CART, C4.5*, dll. Dalam peran estimasi algoritma yang dapat dipakai diantaranya *Linear Regression, Neural Network, Support Vector Machine*, dll. Algoritma yang biasa dimanfaatkan untuk *clustering* yaitu *K-Medoid, K-Means, Self-Organization Map (SOM), Fuzzy C-Means*, dll. Sedangkan algoritma untuk melakukan asosiasi dapat dipakai *Chi Square, FP-Growth, Coefficient of Correlation, A Priori*, dll [6].

Text mining merupakan bagian dari data *mining*, dimana jika dibandingkan antar keduanya *text mining* lebih unggul dengan nilai komersial yang jauh lebih besar, karena sekitar 80% pada tiap perusahaan memiliki data informasi yang berbentuk teks [7]. *Text mining* merupakan kegiatan menulusuri data yang berbentuk teks yang bersumber dari suatu dokumen dengan maksud untuk memperoleh kata-kata yang dapat memberikan gambaran mengenai isi dokumen sehingga kita dapat menganalisis hubungan antar dokumen tersebut.

Secara umum tahapan dalam *text mining* adalah sebagai berikut:

1. *Case Folding*

Tahapan ini dilakukan untuk menghapus karakter yang bukan huruf dan pengubahan huruf berupa huruf kecil.

2. *Tokenizing*

Tokenizing merupakan tahapan pemecahan *string input* yang didasarkan pada kata-kata penyusunnya.

3. *Filtering*

Filtering merupakan tahapan menarik kata-kata bermakna pada hasil *tokenizing*. tahapan ini dapat menggunakan metode *stop list* atau *stopword* (menghapus yang kurang bermakna) atau *word list* (mengumpulkan kata yang dianggap penting). Kata-kata hasil dari *tokenizing* akan dibandingkan dengan kamus *stopword* dari database, bila terdapat kesesuaian antara kata tersebut dengan kata dalam *stopword* maka akan dihapus namun sebaliknya jika kata tidak sesuai maka kata tersebut menuju ke tahap berikutnya.

4. *Stemming*

Stemming merupakan tahap dimana dilakukan pengubahan kata-kata hasil *filtering* menjadi kata dasarnya dengan cara menghapuskan imbuhan dari kata tersebut baik berupa imbuhan awal maupun imbuhan akhir.

5. *Analzying*

Tahap *analyzing* atau disebut juga tahapan pembobotan merupakan tahap penentuan seberapa cocok atau keterhubungan antar kata-kata antar dokumen yang ada berdasarkan hasil perhitungan frekuensi *term* dari suatu dokumen [8]. *Term frequency* (tf) itu sendiri merupakan salah satu metode pembobotan yang digunakan untuk menghitung *term-weighting* (bobot suatu kata) berdasarkan nilai frekuensi kata pada dokumen tertentu, dimana nilai frekuensi kata tersebut diperlukan untuk mendapatkan peluang bayes sehingga dapat dilakukan klasifikasi [9]. Adapun *term frequency* (tf) dinotasikan sebagai berikut.

$$dl = \sum_{i} tf_i \qquad (2.1)$$

Dimana:

tf_i : Frekuensi term t_i
dl : Jumlah frekuensi seluruh kata pada dokumen

Algoritma *Naïve Bayes Classifier*

Naïve Bayes atau lebih dengan istilah Bayesian Classification merupakan bagian dari Machine Learning yang ditemukan oleh Thomas Bayes dengan memberikan prediksi kemungkinan pada masa mendatang yang didasarkan pada kejadian masa lampau menggunakan perhitungan statistik probabilitas (Fitriana, Setifani, & Yusuf, 2020).

Bentuk umum dari teorema bayes dinyatakan dengan:

$$P(a|b) = \frac{P(b|a)P(a)}{P(b)} \qquad (2.2)$$

dengan keterangan:
$P(a|b)$: Peluang hipotesa a didasarkan pada keadaan b
$P(b)$: Peluang hipotesa b
$P(b|a)$: Peluang hipotesa b didasarkan pada keadaan a (*posterior probability*)
$P(a)$: Peluang hipotesa a (*prior probability*)

Dengan menggunakan teorema bayes dalam meyelesaikan masalah pengklasifikasian dengan pendekatan peluang sehingga dasar dari algoritma *Naïve Bayes Classifier* adalah sebagai berikut.

$$P(V_j|a) = \frac{P(\mathrm{a}|V_j) \times P(V_j)}{P(\mathrm{a})} \qquad (2.3)$$

Dengan keterangan:
$P(V_j)$: Peluang tiap-tiap kelas dari sekumpulan dokumen
$P(a|V_j)$: Peluang munculnya kata a berdasarkan kedaan kelas V_j
$P(V_j|a)$:Peluang kemunculan kelas V_j berdasarkan kedaan a
$P(a)$: Peluang a

Multinomial Naïve Bayes adalah nama lain dari *Naïve Bayes Classifier* yang merupakan hasil penyederhanaan algoritma bayes yang cocok untuk klasifikasi teks. Selain *Multinomial Naïve Bayes* terdapat variasi lain dari algoritma *Naïve Bayes* yang dapat digunakan untuk klasifikasi teks yaitu *Multivariate Bernoulli Naïve Bayes* dimana perbedaan kedua variasi algoritma *Naïve Bayes* tersebut yaitu dengan memberikan nilai "1" jika ada fitur dalam dokumen dan "0" untuk sebaliknya pada hasil perhitungan *term* atau pada *Term Document Matrix* atau dikenal dengan istilah *Binary TF*. [10]

K-Nearest Neighbors

Algoritma *K-Nearest Neighbors (k-NN* atau *KNN)* merupakan salah satu metode klasifikasi objek berdasarkan data *learning* yang memiliki jarak terdekat dari objek tersebut. Perhitungan jarak data *query* dengan data *learning* dihitung dengan mengukur jarak titik yang menggambarkan dua data tersebut dengan rumus *Euclidean Distence*. Pada tahap klasifikasi, jarak sebuah vektor baru terhadap seluruh vektor *training sample* dihitung dan diambil sebanyak k buah terdekat. Nilai k yang paling baik didasarkan pada datanya. Secara umum efek *noise* pada klasifikasi akan berkurang jika nilai k nya tinggi, namun batasan antar tiap klasifikasi semakin kabur. [11]

Perhitungan jarak dilakukan dengan menggunakan *Euclidean Distence* atau jarak *Euclidean* yang berguna untuk menguji ukuran yang dapat digunakan sebagai gambaran atau interpretasi kedekatan jarak antara dua objek dengan rumus sebagai berikut.

$$dist = \sum_{i=1}^{p} \sqrt{(X_{1i} - X_{2i})^2} \qquad (2.4)$$

Keterangan:
$dist$: Jarak
X_1 : Data *Training*
X_2 : Data *Testing*
i : Variabel Data
p : Jumlah Atribut

Pengukuran *Performance*

Tujuan dilakukannya pengukuran performansi terhadap algoritma *classifier* yang dihasilkan adalah untuk mendeteksi tingkat akurasi berdasarkan parameter performansi yang terdiri dari nilai akurasi, *recall* dan *precision*. Tabel *confussion matrix* disajikan pada Tabel 2.1.

Tabel 2. 1 *Confussion Matrix*

Kategori		**True Value**	
		Ham	**Spam**
Hasil Klasifikasi	**Ham**	HP	SP
	Spam	SN	HN

dimana:
HP: *Ham Positive* (klasifikasi benar berdasarkan sistem)
SP: *Spam Positive* (klasifikasi salah berdasarkan sistem)
HN: *Ham Negative* (dokumen yang bukan bagian dari suatu klasifikasi kategori)
SN: *Spam Negative* (dokumen yang harusnya menjadi bagian kategori tertentu) [12].

Parameter-parameter tersebut dimanfaatkan untuk melakukan perhitungan dengan 3 metode evaluasi yaitu:

Recall adalah tingkat keberhasilan sistem untuk menemukan kembali suatu informasi. *Recall* dapat dinyatakan sebagai berikut:

$$Recall = \frac{HP}{HP + SP} \quad (2.5)$$

Precision adalah tingkat ketepatan informasi yang diberikan oleh sistem yang sesuai dengan keinginan pengguna. *Precision* dapat dinyatakan dengan persamaan:

$$Precision = \frac{HP}{HP + SN} \quad (2.6)$$

Akurasi diartikan sebagai taraf kedekatan dari nilai hasil prediksi dengan nilai yang sebenarnya. Akurasi dimanfaatkan untuk menilai banyaknya label hasil prediksi yang cocok dengan label konkret. Performansi suatu klasifikasi makin baik jika nilai akurasinya makin besar. Berikut ini adalah persamaan untuk Akurasi:

$$\text{Akurasi} = \frac{HP + HN}{(HP + SP + HN + SN)} \quad (2.7)$$

***Short Message Service* (SMS)**

Pesan singkat atau sering disebut *Short Message Service* (SMS) adalah suatu fasilitas untuk mengirim dan menerima pesan singkat melalui telepon selular yang dapat dikirimkan tanpa menggunakan layanan internet. [13].

Adapun karakter dalam SMS dapat berupa teks (*alphanumeric*) atau *binary non text short messages.* Cara mengirim teks yang tepat dalam komunikasi cepat atau ketika jaringan telepon tengah sibuk, karena tidak memerlukan komputer atau akses internet dan membutuhkan waktu yang relatif singkat [14].

***Spam Short Message Service* (SMS)**

Spam merupakan informasi yang dikirimkan ke beberapa pengguna dimana informasi yang dikirim tidak berhubungan dengan penerima informasi tersebut [15]. Orang yang melakukan spam disebut *spammer*. Sedangkan tindakan spam dikenal dengan nama *spamming* [16]. Secara umum spam memiliki bentuk: spam pada surat elektronik, spam iklan berbasis online, spam pada wiki, spam pada alat penulusuran web (*web search engine spam*), spam pada blog, spam pada media sosial serta spam pesan singkat (SMS). [17]

SMS terbagi menjadi dua yaitu SMS spam dan SMS bukan spam (*ham).* SMS spam adalah pesan sampah atau pesan yang tidak diinginkan untuk diterima yang dikirim ke sebuah telepon genggam sebagai pesan teks melalui SMS. Sedangkan SMS *ham* adalah pesan yang memuat percakapan normal antar sesama pengguna [18]. Jika terjadi pengiriman pesan antar operator yang berbeda dimana skema bisnis antar operator tersebut berbayar maka penerima pesanlah yang akan mengalami kerugian. Kerugian juga dapat terjadi karena adanya SMS spam yang isi SMS tidak diinginkan oleh penerima sehingga menimbulkan rasa tidak nyaman bahkan penipuan. [19]

3. METODOLOGI

Data yang digunakan pada penelitian ini adalah data sekunder. Data set spam SMS yang digunakan pada penelitian ini bersumber dari website UCI Machine Learning Repository di *https://archive.ics.uci.edu/ml/index.php*.

Tahapan Analisis

Adapun langkah-langkah yang ditempuh pada penelitian ini adalah:
1. Menginput data
2. Membersihkan teks dengan menghilangkan simbol ataupun hal lain selain huruf
3. Menghapus kata-kata yang termasuk *stopword*
4. Melakukan *stemming* dengan mengubah kata-kata yang tersisa ke bentuk dasarnya
5. Memisahkan teks menjadi potongan kata/*term*
6. Melakukan pembobotan pada tiap kata

7. Melakukan klasifikasi algoritma *Naïve Bayes Classifier*
8. Melakukan klasifikasi algoritma *K-Nearest Neighbors*
9. Melakukan perbandingan performa hasil klasifikasi antara algoritma *Naïve Bayes Classifier* dan algoritma *K-Nearest Neighbors*

4. HASIL

Analisis Data

Data yang digunakan dalam penelitian merupakan data set spam SMS yang berasal dari website UCI *Machine Learning Repository* dengan nama folder SMS *Spam Collection Data Set* yang berisi 5572 data (SMS) dengan dua atribut (kolom) yaitu *message* dan *category*. Dimana atribut *category* data tersebut terdiri dari 4825 data ham dan 747 data spam.

Pra Pemrosesan Data

Pra pemrosesan data terdiri dari beberapa tahap diantaranya *case folding*, *stopword*, *stemming*, *tokenizing*, dan pembobotan. Tindakan atau perlakuan pada teks SMS di tahap *case folding* diantaranya membersihkan teks dari simbol maupun emotikon, menghilangkan angka, menghilangkan spasi berlebih, menghapus url atau link menggunakan fungsi gsub(), sedangkan untuk mengubah seluruh teks menjadi *lowercase* atau huruf kecil menggunakan fungsi content_transformer(tolower).

Proses yang dilakukan pada tahap *stopword* yaitu menghilangkan *stopword* atau kata-kata yang kurang memiliki arti atau makna menggunakan fungsi removeWords. Proses *stemming* atau pengubahan kata ke bentuk dasar dilakukan menggunakan fungsi stemDocument().

Tahap *tokenizing* dilakukan dengan memecah atau membagi *text* atau *document* menjadi penggalan kata-kata penyusunnya menggunakan fungsi DocumentTermMatrix. Berdasarkan analisis yang dilakukan diperoleh 5944 $term$ dari 5572 document. Untuk mengefisiensikan waktu analisis maka dilakukan filter menggunakan findFreqTerms untuk hanya mengambil kata/*term* yang frekuensi kemunculannya minimal 10 sehingga didapatkan kandidat predictor yang berpengaruh. Adapun nilai *term* yang diperoleh yaitu 827 term.

Pembobotan yang digunakan pada penelitian ini menggunakan metode *term frequency* dengan menggunakan fungsi DocumentTermMatrix. Namun untuk perhitungan peluang nilai frekuensi pada tahap pembobotan perlu diubah ke kondisi muncul (1) atau tidak (0) menggunakan *Bernoulli Converter* dengan ketentuan jika jumlah kata yang muncul ≥ 1(muncul) = 1 sedangkan jika jumlah kata yang muncul 0 (tidak muncul) = 0.

Klasifikasi Menggunakan Metode *Naïve Bayes Classifier*

Penelitian ini menggunakan pembagian partisi data dengan proporsi 80% (4457) untuk data *training* dan 20% (1115) untuk data *testing.* Setelah data *training* dan data *testing* telah ditentukkan selanjutnya adalah menghitung nilai probabilitas pada setiap kelas $P(V_j)$ pada data *training* yaitu Ham (V_1) dan Spam (V_2). Adapun hasil perhitungan probabilitas pada setiap kelas seperti pada Tabel 4.1 berikut.

Tabel 4. 1 Probabilitas Data *Training*

Kelas (V_j)	**N**	**Probabilitas** $P(V_j)$
Ham	3862	0,8665021
Spam	595	0,1334979
Total	**4457**	

Langkah selanjutnya yaitu menghitung peluang setiap kata ($\boldsymbol{a_i}$) pada tiap kelas (V_j) pada data *training* $P(\boldsymbol{a_i}|V_j)$. Misalkan menghitung probabilitas $abiola = \text{Muncul}$ dengan (V_1) yaitu kelas Ham. Banyaknya $abiola = \text{Muncul}$ yang kelasnya Ham adalah 9 dan banyaknya kelas Ham adalah 3862. Adapun perhitungan nilai peluang untuk atribut $abiola$ yang muncul pada SMS diklasifikasi sebagai Ham sebagai berikut.

$$P(abiola = \text{Muncul}|\text{Ham}) = \frac{n(\text{Muncul} \cap \text{Ham})}{n(\text{Ham})} = \frac{9}{3862} = 0{,}002330399$$

Hasil perhitungan tersebut menunjukan bahwa jika terdapat suatu SMS yang

diklasifikasikan sebagai Ham dan ketika dilakukan pengecekan munculnya kata *abiola* pada SMS tersebut diperoleh peluang sebesar 0,002330399.

Selanjutnya menghitung peluang tiap SMS pada data *testing* $P(V_j)\prod_i P(a_i|V_j)$ dan menentukan label yang tepat untuk tiap SMS tersebut berdasarkan nilai probabilitas tertinggi. Adapun beberapa hasil perhitungan peluang dan hasil klasifikasi dapat dilihat pada Lampiran 3 dan beberapa pada Tabel 4.2 berikut.

Tabel 4. 2 Nilai Probabilitas dan Hasil Klasifikasi *Naïve Bayes Classifier*

Ham	Spam	Hasil Klasifikasi
0,6353421	0,3546579	Ham
0,9999836	0,00001642102	Ham
0,995096	0,9994904324	Ham
0,9983207	0,001679346	Ham
0,999953	0,000004719534	Ham
0,999995	0,0000004041225	Ham
1	0	Ham
0,9994293	0,0005701615	Ham
0,9999811	0,00001886589	Ham
0,9957712	0,004228820	Ham

Klasifikasi Menggunakan Metode *K-Nearest Neighborss*

Klasifikasi *K-Nearest Neighbors* menggunakan data *training* dan data *testing* yang sama dengan klasifikasi menggunakan *Naïve Bayes Classifier*. Langkah selanjutnya yaitu menghitung jarak antara tiap titik data *training* dan data *testing* dengan menggunakan persamaan *Euclidean Distence*. Selanjutnya melakukan prediksi data *testing* berdasarkan kelas dari k data *training* terdekatnya, dimana k = 1. Adapun hasil klasifikasinya yaitu kategori Ham sebanyak 1013 dan kategori Spam sebanyak 102.

Perbandingan Performa *Naïve Bayes Classifier* dan *K-Nearest Neighbors*

Pengukuran performa atau *performance* algoritma *classifier* dapat dilihat pada nilai akurasi, *recall*, dan *precision.* Dimana nilai-nilai tersebut dapat dihitung berdasarkan tabel *confussion matrix*. Adapun tabel *confussion matrix* untuk algoritma *Naïve Bayes Classifier* dan *K-Nearest Neighbors* berturut-turut dapat dilihat pada Tabel 4.3 dan Tabel 4.4 berikut.

Tabel 4. 3 *Confussion Matrix Naïve Bayes Classifier*

Klasifikasi	Asli	
	Ham	**Spam**
Ham	958	14
Spam	5	138

$$\text{Akurasi} = \frac{HP + HN}{(HP + SP + HN + SN)} = \frac{958 + 138}{(958 + 14 + 138 + 5)} = 0,983$$

$$Recall = \frac{HP}{HP + SP} = \frac{958}{958 + 14} = 0,986$$

$$Precision = \frac{HP}{HP + SN} = \frac{958}{958 + 5} = 0,995$$

Tabel 4. 4 *Confussion Matrix K-Nearest Neighbors*

Klasifikasi	Asli	
	Ham	**Spam**
Ham	960	51
Spam	3	101

$$\text{Akurasi} = \frac{HP + HN}{(HP + SP + HN + SN)} = \frac{960 + 101}{(960 + 51 + 101 + 3)} = 0,951$$

$$Recall = \frac{HP}{HP + SP} = \frac{960}{960 + 51} = 0,949$$

$$Precision = \frac{HP}{HP + SN} = \frac{960}{960 + 3} = 0,996$$

Adapun perbandingan performa untuk algoritma *Naïve Bayes Classifier* dan *K-Nearest Neighbors* dapat dilihat pada Tabel 4.5 berikut.

Berdasarkan Tabel 4.5 dapat diketahui bahwa akurasi dan nilai *Recall* tertinggi untuk klasifikasi spam SMS yaitu menggunakan metode *Naïve Bayes Classifier* dengan nilai akurasi 0,983 atau 98,3% dan nilai *Recall* 0,986 atau 98,6%. Sedangkan nilai *Precision* tertinggi

yaitu menggunakan metode *K-Nearest Neighbors* dengan nilai 0,996 atau 99,6%.

Tabel 4. 5 Perbandingan Performa *Naïve Bayes Classifier* dan *K-Nearest Neighbors*

Metode	Akurasi	Recall	Precision
Naïve Bayes Classifier	0,983	0,986	0,995
K-Nearest Neighbor	0,951	0,949	0,996

5. PEMBAHASAN

Berdasarakan hasil analisis terhadap data set spam SMS menggunakan metode Naïve Bayes Classifier menunjukkan hasil klasifikasi yang cukup baik dan cukup tepat yang ditunjukkan dengan nilai akurasi prediksi sebesar 0,983 atau 98,3%, nilai Recall sebesar 0,986 atau 98,6% dan nilai Precision sebesar 0,995 atau 99,5% dengan menggunakan perbandingan proporsi training dan testing 80% dan 20%. Dimana dari 1115 data testing, 972 diklasifikasi sebagai Ham dan 143 diklasifikasi sebagai Spam.

Klasifikasi menggunakan metode K-Nearest Neighbors dengan menggunakan perbandingan proporsi yang sama juga menunjukkan hasil yang cukup baik dengan nilai akurasi sebesar 0,951 atau 95,1%, nilai Recall sebesar 0,949 atau 94,9% dan nilai Precision sebesar 0,996 atau 99,6%, dimana klasifikasi kategori Ham sebanyak 1013 dan kategori Spam sebanyak 102.

Jika dilihat dari nilai rasio akurasi antara metode *Naïve Bayes Classifier* dan metode *K-Nearest Neighbors* dengan nilai 1,033 menunjukkan metode *Naïve Bayes Classifier* lebih baik dibandingkan metode *K-Nearest Neighbors*, jika dilihat dari rasio Recall metode *Naïve Bayes Classifier* dan metode *K-Nearest Neighbors* dengan nilai 0,989 menunjukkan metode *Naïve Bayes Classifier* tidak lebih baik dibandingkan metode *K-Nearest Neighbors*, dan jika dilihat dari rasio *Precision* metode *Naïve Bayes Classifier* dan metode *K-Nearest Neighbors* dengan nilai 0,998 menunjukkan metode *Naïve Bayes Classifier* tidak lebih baik dibandingkan metode *K-Nearest Neighbors*.

6. KESIMPULAN

Adapun kesimpulan yang diperoleh berdasarkan penelitian ini yaitu performa klasifikasi spam SMS menggunakan metode Naïve Bayes Classifier maupun metode K-Nearest Neighbors cukup baik karena keduanya menghasilkan nilai akurasi yang cukup besar, namun akurasi klasifikasi menggunakan metode Naïve Bayes Classifier lebih besar yaitu 98,3% dibandingkan metode K-Nearest Neighbors dengan akurasi 95,1% dengan rasio akurasi sebesar 1,033 yang menunjukkan metode Naïve Bayes Classifier memiliki performa yang lebih baik.

7. DAFTAR PUSTAKA

[1] Herwanto, N. L. Chusna and M. S. Arif, "Klasifikasi SMS Spam Berbahasa Indonesia Menggunakan Algoritma Multinomial Naïve Bayes," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 5, no. 4, 2021.

[2] B. Indiarto, "Klasifikasi SMS Spam Dengan Metode Naive Bayes Classifier Untuk Menyaring Pesan Melalui Selular," *Jurnal TELEMATIKA MKOM*, vol. 8, no. 2, 2016.

[3] D. N. Fitriana, N. A. Setifani and A. Yusuf, "Perbandingan Algoritma Naïve Bayes, Svm, Dan Decision Tree Untuk Klasifikasi SMS Spam," *JUSIM (Jurnal Sist. Inf. Musirawas)*, vol. 5, no. 2, 2020.

[4] A. S. Dharma, O. . Y. Silitonga and . H. J. Manurung, "Perbandingan Algoritma Naive Bayes, ID3 dan TAN Pada Klasifikasi SMS Spam," *J. Marit. Educ*, vol. 1, no. 2, 2019.

[5] M. A. Muslim, B. Prasetiyo, E. L. H. Mawarni, A. J. Herowati, Mirqotussa'adah, S. H. Rukmana and A. Nurzahputra, Data Mining Algoritma C4.5 Disertai contoh kasus dan penerapannya dengan program computer, Semarang, 2019.

[6] J. Suntoro, Data Mining Algoritme dan Implementasi Menggunakan Bahasa Pemograman PHP, Semarang, 2018.

[7] A. Firdaus and W. I. Firdaus, "Text Mining dan Pola Logoritma dalam Penyelesaian Masalah Informasi: (Sebuah Ulasan)," *Jurnal JUPITER,* vol. 13, no. 1, 2021.

[8] D. . W. B and A. Hetami, "Perancangan Information Retrieval (IR) untuk Pencarian Ide Ppkok Teks Artikel Berbahasa Inggris dengan Pembobotan Vector Space Model," *Jurnal Ilmiah Teknologi dan Informasi ASIA,* vol. 9, no. 1, 2015.

[9] W. Sulistyo, "Klasifikasi Dokumen Berbahasa Inggris Berdasarkan Weighted-Term," *Jurnal Teknologi Informasi-Aiti,* vol. 5, no. 1, 2008.

[10] Widyawati and Sutanto, "Perbandingan Kinerja Variasi Naive Bayes Multivariate Bernoulli dan Naive Bayes Multinomial Dalam Pengklasifikasian Dokumen Teks," *JURNAL OF INNOVATION AND FUTURE TECKHNOLOGY,* vol. 2, no. 1, 2020.

[11] O. S. Y. Prakasa and K. M. Lhaksamana, "Klasifikasi Teks dengan Menggunakan Algoritma K-Nearest Neighbors pada Kasus Kinerja Pemerintah di Twitter," *e-Proceeding of Engineering,* vol. 5, no. 3, 2018.

[12] A. Deolika, K. and E. . T. Luthfi, "Analisis Pembobotan Kata Pada Klasifikasi Text Mining," *JTI (Jurnal Teknologi Informasi),* vol. 3, no. 2, 2019.

[13] Pratama, A. Gumilar, Anton and Firmansyah, "Implementasi Aplikasi Enkripsi Short Message Service (SMS) Berbasis Android," *JURNAL TEKNIK KOMPUTER AMIK BSI,* vol. 1, no. 1, 2015.

[14] A. Zuliharti, Klasifikasi Spam Pada Konten SMS Menggunakan Metode Naive Bayes dengan Featur Selection Document Frequency Thresholding, Malang: Universitas Brawijaya, 2012.

[15] Sandag and G. Arther, "Klasifikasi SMS Spam Menggunakan Algoritma Support Vector Machine (SVM)," *Seminar Nasional Sistem Informasi dan Teknologi Informasi 2018,* 2018.

[16] F. R. Nasution, Shaufiah and M. A. Bijaksana, "SMS Classification Deteksi Spam dengan menggunakan Algoritma Artificial Immune System dan Apriori Frequent Itemset," *eProceedings of Engineering,* vol. 2, no. 3, 2015.

[17] . T. H. A. Sugianto and C. Agus, "Analisis Komparasi Machine Learning Pada Data Spam SMS," *TEDC,* vol. 12, no. 1, 2018.

[18] R. D. H. Lumbantobing, E. M. Manalu, D. S. P. Sitinjak and T. W. Manurung, "Rancangan Aplikasi Mobile Pendeteksi Spam SMS di Indonesia," *jurnaltio,* vol. 2, no. 1, 2021.

[19] J. Na'am, "Pembobotan Kata SMS Spam," *Jurnal Ilmiah Media SISFO,* vol. 9, no. 2, 2015.

[20] K. S. Berberian, Introduction to Hilbert Space, New York: Oxpord University Press, 1961.

[21] Dwijanto, Analisis Real, Semarang: IKIP Semarang Press, 1994.

[22] E. Kreyzeq, Introduction Functional Analysis with Application, Canada: John Wiley & Son, 1978.

[23] T. P. Nababan, Teorema Titik Tetap di Ruang Metrik dan Aplikasinya, Bandung: Institut Teknologi Bandung, 1992.

[24] B. I, "Groups DCS," 13 Mei 2008. [Online]. Available: http://www.group.dcs.stand.ac.uk. [Accessed 13 Mei 2011].

[25] M. Bulmer and Carter, M, Integer Programming with Mathematica, USA: Inc. USA, 1996.

[26] J. A. 2. Bychmann, Introduction to Cryptography, New York: Inc. USA, 2000.

[27] G. Ginan, "Teori Bilangan dalam Persamaan Diophantine Journal," *Teknik Elektro dan Informatika,* 2008.

[28] A. Setiawan, Pengantar Teori Probabilitas, Salatiga: Tisara Grafika, 2015, pp. 25-26.

[29] Noeryanti, Pengantar Teori Probabilitas Edition One, Yogyakarta: AKPRIND Press, 2021, p. 21.

[30] Irwan, "perancangan Aplikasi SMS (Short Message Service) dengan Enskripsi Teks Menggunakan Algoritma Block Cipher Aes (Advanced Encryption Standard) Berbasis Mobile Pada Platform Android," *Jurnal Sistem dan Teknologi Informasi,* vol. 1, no. 1, 2013.

[31] M. Shihab, Tafsir Al-Misbah: Pesan, Kesan, dan Keserasian Al-Qur'an (Vol. 13), Jakarta: Lentera Hati, 2002.

[32] S. A. Hasanah, "Larangan Membicarakan Semua yang didengar," 30 03 2020. [Online].

[33] Butar and Ricky Kristian, "Implementasi dan Analisis Klasifikasi Spam Pada Pesan Singkat Seluler Dengan Pendekatan Collaborative Filtering Menggunakan Naïve Bayes," vol. 2, no. 3, 2015.

[35] A. N. Fajar, "Pemanfaatan Teknologi SMS (Short Message Service) Dalam Instituti Perguruan Tinggi," *Jurnal FASILKOM,* vol. 4, no. 1, 2006.

[36] J. Brown, B. Shipman and R. Vetter, "SMS: The Short Message Service," *ResearchGate,* 2007.

[37] H. Muhamad, C. A. Prasojo, N. A. Sugianto, L. Surtiningsih and I. Cholissodin, "Optimasi Naive Bayes Classifier dengan Menggunakan Particle Swarm Optimiation Pada DataIRIS," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK),* vol. 4, no. 3, 2017.

[38] A. Wahid, M. Baharulloh, R. Kahfiansyah, T. Abrilianto, A. Saifudin and S. Mulyati, "Identifikasi SMS Spam Menggunakan Metode Naive Bayes," *Jurnal Informatika Universitas Pamulang,* vol. 6, no. 3, 2021.

[39] R. J. Mooney., "CS 391L: Machine Learning Text Categoriation," 2006.

[40] Lajnah Pentashih Mushaf Al-Qur'an Departemen Agama RI, Al-Qur'an dan Teremahnya, Fajar Mulya.

[41] A. Saleh, "Implementasi Metode Klasifikasi Naive Bayes dalam Memprediksi Besarnya Penggunaan Listrik Rumah Tangga," *Citec Journal,* vol. 2, no. 3, 2015.