

Analisis Sentimen Pada Ulasan Aplikasi Samsat Mobile Jawa Barat (Sambara) Menggunakan algoritma *Naive bayes Classifier* dengan seleksi fitur *Chi Square*

Muhammad Irfan¹, Didit Widiyanto², Jayanta³
Fakultas Ilmu Komputer

Universitas Pembangunan Nasional Veteran Jakarta
Jl. Rs. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12450, Indonesia
m.irfan@upnvj.ac.id¹, didit.widiyanto@upnvj.ac.id², anta.jayanta@gmail.com³

Abstrak. Aplikasi Sambara merupakan inovasi terhadap pelayanan pembayaran pajak kendaraan bermotor di daerah Jawa Barat yang dibuat oleh Bapenda Jabar. Dengan hadirnya Sambara, diharapkan dapat mempercepat dalam melakukan kewajiban membayar pajak kendaraan tahunan. Pada penelitian ini akan dilakukan proses analisis sentimen dengan mengklasifikasi ulasan pengguna aplikasi Sambara kedalam sentimen positif dan sentimen negatif. Data yang digunakan merupakan data ulasan aplikasi Sambara di Play Store berbahasa Indonesia pada tanggal 20 April hingga 31 Desember 2021. Penelitian ini menggunakan algoritma *Naive Bayes Classifier* dan seleksi fitur *Chi Square*. Lalu dilakukan pembagian data latih dan data uji dengan ratio 80:20, hasil dari penelitian dengan menggunakan seleksi fitur *chi square* dengan nilai kritis sebesar 0.837 dan jumlah fitur yang terseleksi sebanyak 244 fitur mendapatkan nilai akurasi tertinggi sebesar 94.4%, specificity 93.5%, dan recall 95.3%.

Kata kunci: Sambara, *Naive Bayes Classifier*, *Chi Square*

1. Pendahuluan

Pelayanan publik merupakan salah satu jenis interaksi yang dilakukan antara pemerintah dan masyarakatnya. Pelayanan publik di masa pandemi ini mengharuskan kualitas pelayanan masyarakat harus tetap dipertahankan. Pemerintah harus mengupayakan untuk terus memberikan pelayanan terbaik sekaligus tetap melaksanakan himbauan protokol kesehatan untuk keselamatan bersama, baik pemberi maupun penerima layanan, dengan menerapkan *social* dan *physical distancing* [1].

Salah satu pelayanan publik yang sering digunakan yaitu pembayaran pajak kendaraan. Hampir seluruh keluarga di Indonesia memiliki minimal 1 kendaraan bermotor di rumahnya. Berdasarkan data dari Badan Pusat Statistik (BPS) menunjukkan jumlah kendaraan bermotor yang terdaftar (sepeda motor, mobil penumpang, bus, dan truk) di seluruh Indonesia lebih dari 136 juta unit pada tahun 2020 dan jumlah tersebut meningkat sekitar 5% setiap tahunnya.

Aplikasi SAMBARA yang dirilis pada tanggal 2 Januari 2018 ini telah diunduh lebih dari 1 juta pengguna, aplikasi ini menawarkan beberapa layanan seperti, melakukan pembayaran pajak kendaraan secara online, mengetahui lokasi pelayanan samsat di daerah sekitar, hingga jadwal samsat keliling dan samsat gendong. Dari banyaknya pengguna yang telah mengunduh aplikasi ini, terdapat beberapa pengguna yang merasa tidak puas terhadap kinerja aplikasi SAMBARA, hal ini dapat dilihat pada ulasan aplikasi SAMBARA di playstore. Ulasan pengguna yang berisikan feedback setelah memakai aplikasi, dan rating dapat dijadikan acuan untuk menganalisis, memahami kebutuhan dan keluhan yang dialami pengguna guna meningkatkan kinerja aplikasi dengan menggunakan analisis sentimen.

Dari beberapa jurnal disebutkan bahwa algoritma *Naive bayes* dan penggunaan fitur seleksi *Chi Square* cukup handal dalam melakukan proses klasifikasi pada analisis sentimen. Oleh karena itu penulis ingin melakukan penelitian Analisis Sentimen Pada Ulasan Aplikasi Samsat Mobile Jawa Barat (Sambara) Menggunakan algoritma *Naive bayes Classifier* dengan seleksi fitur *Chi Square*.

2. Landasan Teori

2.1 Text Mining

Text mining adalah proses mengatur data untuk mengekstrak pola dari kumpulan teks untuk tujuan tertentu. *Text mining* dapat didefinisikan sebagai proses intensif pengetahuan di mana pengguna berinteraksi dengan teks atau kumpulan dokumen dari

waktu ke waktu menggunakan alat analisis [2].

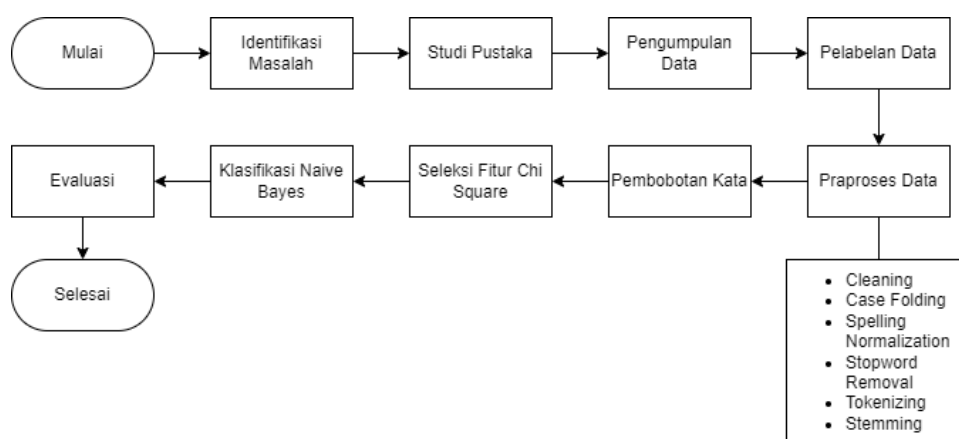
2.2 Analisis Sentimen

Analisis sentimen merupakan bidang studi yang bertujuan untuk menganalisis pendapat, penilaian, dan emosi seseorang terhadap penggunaan entitas seperti produk, layanan, peristiwa, dan lainnya [3]. Analisis sentimen digunakan untuk menentukan/ mengevaluasi opini publik mengenai kepuasan atau ketidakpuasan terhadap barang dan jasa tertentu [4]. Secara umum, analisis sentimen dibagi menjadi 3 tingkatan, yaitu document level, sentence level, dan entity and aspect level [5].

2.3 Sambara

SAMBARA adalah sebuah aplikasi yang diluncurkan oleh BAPENDA Jawa Barat untuk melakukan pembayaran pajak kendaraan bermotor di wilayah Jawa Barat pada 2 Januari 2018. Aplikasi Sambara sendiri telah diunduh lebih dari 1 juta lebih pengguna. Aplikasi ini memiliki beberapa fitur yang dapat membantu pengguna mempercepat proses pembayaran pajak kendaraan bermotor seperti melakukan pengecekan info pajak kendaraan dan PNBP, melakukan pembayaran secara online, melihat daftar lokasi pelayanan samsat hingga jadwal pelaksanaan samsat keliling (SAMLING) dan samsat gendong (SAMDONG). Aplikasi SAMBARA saat ini hanya tersedia di ponsel berbasis Android saja [6].

3. Metodologi Penelitian



Gambar. 1. Flowchart Penelitian

3.1 Identifikasi Masalah

Pada penelitian ini, hal pertama yang dilakukan peneliti adalah identifikasi masalah berdasarkan topik yang ditentukan. Permasalahan yang dibahas terkait penelitian ini adalah tentang pengklasifikasi kepuasan pengguna aplikasi Sambara menggunakan algoritma Naive Bayes Classifier dan seleksi fitur Chi Square, serta menentukan batasan pada permasalahan yang akan diangkat.

3.2 Studi Pustaka

Studi pustaka dalam penelitian ini digunakan sebagai dasar peneliti dalam proses analisis sentimen dengan mengumpulkan buku serta jurnal terkait mengenai analisis sentiment, seleksi fitur chi square, klasifikasi, dan algoritma Naive Bayes Classifier yang dibahas dalam penelitian dan lain sebagainya. Permasalahan tersebut dicari dengan pengumpulan berbagai macam literature, jurnal dan ebook yang berkaitan dengan topik penelitian ini.

3.3 Pengumpulan Data

Pengumpulan data pada penelitian ini dilakukan dengan cara scraping data di google play store dengan bantuan library google-play-scraper. Data ulasan yang diambil adalah ulasan aplikasi berbahasa Indonesia pada versi 4.0.7 berdasarkan tanggal 20 April hingga 31 Desember 2021 sebanyak 1568 data ulasan pengguna. Data tersebut kemudian di export ke dalam format comma separated value (csv) untuk memudahkan tahap selanjutnya.

3.4 Pelabelan Data

Pada tahap ini, pelabelan data dilakukan dengan memberikan label pada ulasan aplikasi yang telah dikumpulkan berdasarkan sentimen positif dan negatif. Ketentuan pemberian label yaitu, label positif merupakan ulasan yang mendukung adanya aplikasi serta penggunaannya, label negatif merupakan kritik atau keluhan terhadap pemakaian aplikasi, dan sisanya ulasan

netral merupakan ulasan yang tidak berisi dukungan ataupun kritik terhadap aplikasi sambara misalnya, ulasan berbentuk pertanyaan akan dihapus.

Selanjutnya dilakukan uji *kappa value*, metode pengujian bernama *fleiss kappa* digunakan untuk mengukur reliabilitas antar *annotator* pada data yang telah diberi label. Perhitungan untuk mendapatkan nilai dari persamaan *fleiss kappa* pada Persamaan (1).

$$Kappa = \frac{P_o - P_e}{1 - P_e} \quad (1)$$

3.5 Praproses Data

Praproses data diperlukan karena data yang dikumpulkan masih memiliki simbol dan kata-kata yang kurang bermakna sehingga akan mempengaruhi dalam proses klasifikasinya. Praproses data juga diperlukan guna membersihkan, dan menyiapkan data menjadi data yang lebih terstruktur. Tahapan yang dilakukan pada praproses data, yaitu *cleaning*, *case folding*, *spelling normalization*, *stopword removal*, *tokenizing*, dan *stemming*.

- a) *Cleaning*
Tahapan ini membersihkan data dari karakter-karakter yang terdapat pada data seperti angka, tanda baca, simbol-simbol, dan karakter lainnya yang tidak memiliki nilai yang berarti.
- b) *Case Folding*
Tahapan ini merubah kata-kata kedalam bentuk yang sama yaitu huruf kecil (lowercase) agar memudahkan pada proses pengklasifikasiannya.
- c) *Spelling Normalization*
Pada tahapan *spelling normalization* dilakukan seleksi kata yang salah eja, singkatan, dan bahasa slang dan memperbaiki kata tersebut berdasarkan KBBI, dengan cara membandingkan tiap kata yang ada pada dataset ke dalam kamus normalisasi.
- d) *Stopword Removal*
Pada tahapan ini dilakukan proses menghilangkan term atau kata yang tidak memiliki nilai yang penting dan tidak memiliki pengaruh pada suatu kalimat tetapi kata tersebut sering muncul.
- e) *Tokenizing*
Tokenizing merupakan proses pemecahan suatu ulasan kalimat menjadi kata per kata berdasarkan karakter spasi. Hasil dari tokenizing berupa kata, frasa, atau elemen yang berarti.
- f) *Stemming*
Pada tahap ini dilakukan proses pemotongan imbuhan pada suatu kata, kemudian kata tersebut diubah ke dalam bentuk kata dasar.

3.3 Pembobotan Kata

Term Frequency-Inverse Document Frequency (TF-IDF) merupakan proses yang mengubah data tekstual menjadi data numerik yang nantinya akan dilakukan pemberian bobot untuk setiap kata di dalam dokumen. Semakin besar bobot suatu kata, semakin banyak kata tersebut muncul dalam dokumen, dan sebaliknya [7].

Rumus perhitungan *term frequency* (TF) pada Persamaan (2).

$$TF = W_{dvtj} \quad (2)$$

Rumus perhitungan *inverse document frequency* (IDF) pada Persamaan (3).

$$IDF = \log\left(\frac{n}{d_{ft}}\right) \quad (3)$$

Rumus perhitungan TF-IDF pada Persamaan (4).

$$W_{t,d} = TF \times IDF \quad (4)$$

Keterangan :

- W_{dvtj} = bernilai 1 jika kata tersebut terdapat didalam dokumen, bernilai 0 jika di dalam dokumen tidak terdapat kata.
- N = Total dokumen.
- D_{ft} = Total dokumen yang memiliki kata (t)

3.4 Seleksi Fitur *Chi-Square*

Proses pemilihan fitur yang relevan dengan tujuan mendapatkan fitur yang optimal dan mengurangi dimensi data dikenal

sebagai seleksi fitur. Dari ribuan fitur yang telah didapat, sangat umum terdapat sejumlah fitur yang tidak informatif karena tidak relevan dengan kelas yang dicari. Oleh karena itu, penghapusan fitur yang tidak relevan dan berlebihan ini dapat meningkatkan kinerja klasifikasi [8].

Seleksi fitur Chi Square berdasar pada teori statistika untuk menguji derajat kepentingan dari sebuah term dengan kategorinya. Penggunaan uji chi square dapat mendeteksi dan menghilangkan fitur tanpa mengurangi tingkat akurasi [9]. Perhitungan nilai Chi Square pada Persamaan (5).

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i} \quad (5)$$

Keterangan:

- O_i = *Observed value* (nilai yang sebenarnya)
- E_i = *Expected value* (nilai yang diharapkan)

3.5 Klasifikasi Naïve bayes

Naive Bayes merupakan algoritma pembelajaran mesin, yang sering digunakan untuk klasifikasi teks. Pengklasifikasi Naive Bayes memperlakukan fitur secara kondisional independen satu sama lain. Dalam naive bayes, probabilitas kelas untuk dokumen tertentu dihitung, dan dokumen ditempatkan di kelas yang memiliki probabilitas tertinggi [10].

Pada saat proses *training* akan dihitung probabilitas kemunculan suatu dokumen pada kategori yang tersedia, proses tersebut menggunakan Persamaan (6).

$$P(c_j) = \frac{|dokumen\ c|}{|dokumen|} \quad (6)$$

Keterangan :

- $P(c_j)$: probabilitas munculnya suatu dokumen dalam kategori j
- $dokumen\ c$: jumlah dokumen untuk setiap kategori j
- $dokumen$: jumlah semua dokumen

Setelah itu hitung nilai probabilitas kemunculan setiap kata pada suatu kategori j berdasarkan Persamaan (7).

$$P(w_i | c_j) = \frac{n(w_i, j) + 1}{|C| + n(kosakata)} \quad (7)$$

Keterangan :

- $P(w_i | c_j)$: probabilitas dari kemunculan kata w_i pada suatu kategori j
- $n(w_i, j)$: jumlah kemunculan kata w_i dalam suatu kategori j
- $|C|$: semua kata yang terdapat pada kategori j
- $n(kosakata)$: jumlah keseluruhan kata

Saat data ulasan mengalami proses klasifikasi, pendekatan metode ini akan menyaring kategori dengan nilai probabilitas tertinggi (c_{MAP}) menggunakan Persamaan (8).

$$(c_{MAP}) = \operatorname{argmax}_{c_j} P(c_j) \prod_i P(w_i | c_j) \quad (8)$$

Keterangan :

- $P(c_j)$: probabilitas munculnya kategori j dalam suatu dokumen.
- $P(w_i | c_j)$: probabilitas munculnya kata w_i pada kategori j

3.6 Evaluasi

Setelah dilakukan klasifikasi, diperlukan evaluasi mengenai kinerja klasifikasi tersebut. Confusion matrix merupakan salah satu metode evaluasi, metode ini memiliki fungsi menganalisis apakah classifier tersebut baik dalam memprediksi kelas atau label. [11].

Tabel 1. Confusion Matrix

		Kelas Prediksi	
		Positif	Negatif
Kelas	Positif	TP	FP
	Negatif	FN	TN

Sebenarnya	Negatif	FN	TN
------------	---------	----	----

Keterangan:

True Positive (TP) : Total data yang diprediksi benar positif.
 False Positive (FP) : Total data bernilai positif yang diprediksi negatif.
 False Negative (FN) : Total data bernilai negatif yang diprediksi positif.
 True Negative (TN) : Total data yang diprediksi benar negatif.

Rumus perhitungan nilai accuracy pada Persamaan (9).

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+FP+FN+TN)} \quad (9)$$

Rumus perhitungan nilai recall pada Persamaan (10).

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (10)$$

Rumus perhitungan nilai specificity pada Persamaan (11).

$$\text{Specificity} = \frac{TN}{(TN+FP)} \quad (11)$$

4. Hasil dan Pembahasan

Penelitian ini menggunakan data ulasan pengguna aplikasi yang dibuat oleh Puslia Bapenda Provinsi Jabar yaitu "SAMBARA". Dalam proses pengumpulan data ini dilakukan dengan teknik *scraping* ulasan menggunakan library python, yaitu *google-play-scrapper*. Data ulasan yang digunakan merupakan ulasan aplikasi pada versi 4.0.7. yang diunggah pada tanggal 20 april 2021 hingga tanggal 31 desember 2021. Pada penelitian ini hanya dibutuhkan beberapa data yang diambil antara lain, isi dari ulasan pengguna(*content*), jumlah bintang yang diberikan pengguna(*score*), dan waktu pengiriman ulasan pada *play store(at)*. Lalu didapatkan data ulasan sebanyak 1567 yang akan masuk ke tahap selanjutnya.

Lalu dilakukan pelabelan data dilakukan dengan memberikan label pada ulasan aplikasi yang telah dikumpulkan berdasarkan kelas sentimen positif dan negatif. Dimana label positif merupakan ulasan yang berisikan dukungan ataupun kelebihan setelah pemakaian aplikasi, label negatif merupakan ulasan keluhan atau kritik penggunaan aplikasi, dan sisanya ulasan netral merupakan ulasan yang tidak bersifat mendukung atau mengkritik aplikasi, seperti, ulasan berbentuk pertanyaan akan dihapus.

Pemberian label ini dilakukan oleh 3 *annotator* berdasarkan pemahaman masing-masing *annotator* setelah membaca dan memahami ulasan tersebut. Rincian dari pelabelan yang dilakukan oleh *annotator* dideskripsikan pada Tabel 2.

Tabel 2. Rincian Pelabelan Data *Annotator*

Keterangan	Data positif	Data negatif	Total
<i>Annotator 1</i>	400	492	892
<i>Annotator 2</i>	404	488	892
<i>Annotator 3</i>	402	490	892
Total	1206	1470	2676

Data yang awalnya berjumlah 1568, setelah dilakukan pelabelan menjadi 892 data. Setelah itu dilakukan perhitungan kappa menggunakan Persamaan 1 berdasarkan hasil dari proses pelabelan keseluruhan data ulasan pengguna aplikasi Sambara yang berjumlah 892 data setelah dihilangkan ulasan netral, didapatkan nilai *fleiss kappa* sebesar 0.97.

Berdasarkan interpretasi *kappa* pada Gambar 1, maka pelabelan data ulasan aplikasi sambara yang dilakukan oleh 3 *annotator* berada pada kategori *Almost perfect agreement* yang berarti reliabilitas dari label cukup baik.

K	Interpretation
<0	Poor agreement
0.01 – 0.20	Slight agreement
0.21 – 0.40	Fair agreement
0.41 – 0.60	Moderate agreement
0.61 – 0.80	Substantial agreement
0.81 – 1.00	Almost perfect agreement

Gambar. 2. Interpretasi kappa

Data ulasan yang telah diberi label kemudian masuk ke tahap praproses data. Tahapan yang dilakukan pada praproses data, yaitu *cleaning*, *case folding*, *spelling normalization*, *stopword removal*, *tokenizing*, dan *stemming*. Hasil dari perubahan data sebelum dan sesudah praproses di deskripsikan pada Tabel 3.

Tabel 3. Hasil Praproses Data Sampel

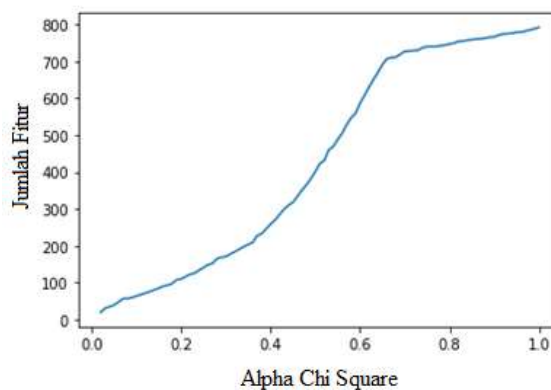
Sebelum Praproses	Setelah Praproses
Oke banget, sangat membantu.. Mempermudah para pengendara dan wajib pajak kendaraan bermotor	['oke', 'bantu', 'mudah', 'kendara', 'wajib', 'pajak', 'kendara', 'motor']
Seharian kemaren server sibuk terus. Sempet ganti device , ganti ISP masih sama. Hingga tengah malam baru bisa .	['hari', 'kemarin', 'server', 'sibuk', 'ganti', 'perangkat', 'ganti', 'isp', 'malam']
2x melakukan pajak motor online pake aplikasi Sambara lancar... top...	['pajak', 'motor', 'jaring', 'pakai', 'aplikasi', 'sambara', 'lancar', 'bagus']

Setelah itu dilakukan pembobotan kata, tahapan ini bertujuan untuk memberikan nilai pada setiap kata agar memudahkan dalam proses seleksi fitur dan klasifikasi nantinya. Perhitungan pembobotan kata TF-IDF menggunakan Persamaan (2), (3), dan (4). Pada tahapan ini menggunakan data dari hasil praproses *stemming* sebanyak 794 kata dari jumlah ulasan sebanyak 892 data. Hasil dari pembobotan TF-IDF menggunakan 3 sampel data ulasan yang telah di praproses pada Tabel 3 dideskripsikan pada Tabel 4.

Tabel 4. Perhitungan TF-IDF Data Sampel

Kata	TF			DF	IDF	TF-IDF		
	d1	d2	d3			d1	d2	d3
aplikasi	0	0	1	1	0.477121	0	0	0.477121
bagus	0	0	1	1	0.477121	0	0	0.477121
bantu	1	0	0	1	0.477121	0.477121	0	0
daring	0	0	1	1	0.477121	0	0	0.477121
ganti	0	2	0	1	0.477121	0	0.954243	0
hari	0	1	0	1	0.477121	0	0.477121	0
isp	0	1	0	1	0.477121	0	0.477121	0
kemarin	0	1	0	1	0.477121	0	0.477121	0
kendara	2	0	0	1	0.477121	0.954243	0	0
lancar	0	0	1	1	0.477121	0	0	0.477121
malam	0	1	0	1	0.477121	0	0.477121	0
motor	1	0	1	2	0.176091	0.176091	0	0.176091
mudah	1	0	0	1	0.477121	0.477121	0	0
oke	1	0	0	1	0.477121	0.477121	0	0
pajak	1	0	1	2	0.176091	0.176091	0	0.176091
pakai	0	0	1	1	0.477121	0	0	0.477121
perangkat	0	1	0	1	0.477121	0	0.477121	0
pernah	0	1	0	1	0.477121	0	0.477121	0
sambara	0	0	1	1	0.477121	0	0	0.477121
server	0	1	0	1	0.477121	0	0.477121	0
sibuk	0	1	0	1	0.477121	0	0.477121	0
wajib	1	0	0	1	0.477121	0.477121	0	0

Setelah seluruh kata diberi bobot menggunakan TF-IDF, kemudian dilakukan perhitungan *chi square* pada masing-masing kata berdasarkan Persamaan (5). Setelah nilai *chi square* pada tiap kata didapat, dilakukan seleksi fitur berdasarkan nilai kritis pada taraf nyata α 0.01 hingga 0.99 dengan kenaikan α sebesar 0.01. Jika nilai *chi square* pada kata tersebut lebih besar atau sama dengan nilai kritis pada taraf nyata α , maka kata tersebut dipertahankan, tetapi jika kata tersebut memiliki nilai *chi square* lebih kecil daripada nilai kritis taraf nyata maka kata tersebut dihapus dari dataset. Hasil perbandingan fitur yang terseleksi menggunakan *chi square* di ilustrasikan pada Gambar 2.



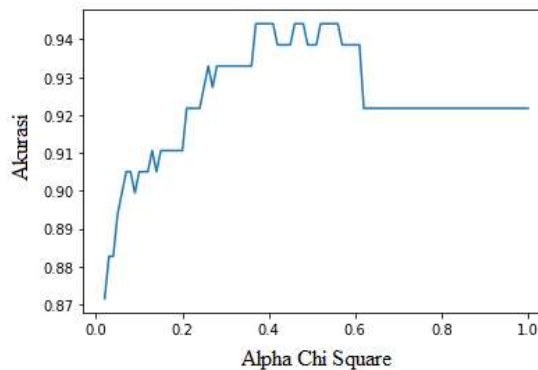
Gambar. 3. Perbandingan Fitur yang Terseleksi

Sebelum masuk ke tahapan klasifikasi, dilakukan pembagian data uji dan data latih berdasarkan fitur yang telah terseleksi oleh seleksi fitur *chi square*. Pada penelitian ini pembagian data menggunakan ratio 80:20. Data yang digunakan sebagai data latih untuk membuat model berjumlah 80% dari 892 yaitu 713 data, dan untuk pengujian menggunakan data uji sebanyak 20% dari 892 yaitu 179 data. Rincian data latih dan data uji di deskripsikan pada Tabel 5.

Tabel 5. Rincian data latih dan data uji

Keterangan	Data positif	Data negatif	Total data
Data Latih	319	394	713
Data Uji	86	93	179

Setelah itu dilakukan eksperimen pengujian *naïve bayes* sebanyak 99 kali dengan menggunakan data uji sebesar 20% yaitu 179 data yang telah dilakukan seleksi fitur *chi square* berdasarkan nilai kritis pada alpha 0.01 hingga alpha 0.99 dengan kenaikan alpha sebesar 0.01. Setelah itu dilakukan perhitungan akurasi berdasarkan jumlah prediksi yang dikategorikan dengan benar. Hasil perbandingan akurasi di ilustrasikan pada Gambar 3.



Gambar. 4. Perbandingan akurasi dengan penggunaan chi square

Evaluasi ini dilakukan berdasarkan pemodelan *naïve bayes* dan seleksi fitur dengan akurasi tertinggi yaitu 94.4% pada saat nilai kritis *chi square* sebesar 0.837. Lalu dilakukan evaluasi dengan menggunakan *confusion matrix* untuk mendapatkan nilai akurasi, *recall*, dan *specificity* berdasarkan jumlah data yang dikategorikan benar ataupun salah pada masing-masing kelas yang dideskripsikan pada Tabel 6.

Tabel 6. Confusion Matrix

		Kelas sebenarnya	
		positif	negatif
Kelas prediksi	positif	82 (TP)	6 (FP)
	negatif	4 (FN)	87 (TN)

$$\text{Akurasi} = \frac{TP+TN}{TP+FP+FN+TN} = \frac{82+87}{82+4+87+6} = \frac{169}{179} = 0.944$$

$$\text{Recall} = \frac{TP}{TP+FN} = \frac{82}{82+4} = \frac{82}{86} = 0.953$$

$$\text{Specificity} = \frac{TN}{TN+FP} = \frac{87}{87+6} = \frac{87}{93} = 0.935$$

Dari data yang ada kemudian dilakukan visualisasi. Pada tahap visualisasi ini, dilakukan representasi dari tiap kata yang muncul pada ulasan positif ataupun ulasan negatif, dengan menggunakan *wordcloud*. Tahapan ini dilakukan dengan bantuan library *wordcloud python*, dengan cara memasukkan semua kalimat pada ulasan positif atau negatif yang telah di praproses, lalu library *wordcloud* akan menghitung kemunculan kata yang ada, semakin banyak suatu kata muncul, semakin besar ukuran kata tersebut. Hasil Visualisasi *wordcloud* untuk sentimen positif pada ulasan pengguna masyarakat diilustrasikan pada Gambar 4 dan Visualisasi *wordcloud* untuk sentimen negatif pada ulasan pengguna masyarakat diilustrasikan pada Gambar 5.

5.2 Saran

Dari hasil penelitian yang telah dibuat ada beberapa saran untuk pengembangan penelitian, yaitu:

- a) Penelitian berikutnya diharapkan menggunakan metode seleksi fitur lainnya seperti Information gain, Odds Ratio, Bi-Normal Separation, dan lain sebagai pembandingan dengan metode chi square.
- b) Penelitian berikutnya diharapkan menggunakan label netral pada proses klasifikasinya.
- c) Penelitian berikutnya diharapkan menggunakan metode klasifikasi lainnya seperti Linear Regression, K-Nearest Neighbor, Support Vector Machine, Long Short Term Memory.

Referensi

- [1] A. Rohman and D. C. Larasati, "STANDART PELAYANAN PUBLIK DI ERA TRANSISI NEW NORMAL," *REFORMASI*, pp. 151-163, 2020.
- [2] E. P. Nirwadani, I. and R. C. Wihandika, "Analisis Sentimen Pada Ulasan Pengguna Aplikasi Mandiri Online Menggunakan Metode Modified Term Frequency Scheme Dan Naïve Bayes," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 1039-1047, 2021.
- [3] B. Liu, *Sentiment Analysis and Opinion Mining*, Chicago: Morgan & Claypool, 2012.
- [4] L. Luthfiana, J. C. Young and A. Rusli, "Implementasi Algoritma Support Vector Machine dan Chi Square untuk Analisis Sentimen User Feedback Aplikasi," *ULTIMATICS*, pp. 125-128, 2020.
- [5] A. R. Isnain, A. I. Sakti, D. Alita and N. S. Marga, "Algoritma Naive Bayes: Sebuah Literature Review JAKARTA MENGGUNAKAN ALGORITMA SVM," *JDMSI*, pp. 31-37, 2021.
- [6] Bapenda Jabar, "Sambara – Samsat Mobile Jawa Barat," 15 November 2021. [Online]. Available: <https://bapenda.jabarprov.go.id/samsat-mobile-jawa-barat-sambara/>.
- [7] J. A. Septian, M. T. Fahrudin and A. Nugroho, "Analisis Sentimen Pengguna Twitter Terhadap Polemik Persepakbolaan Indonesia Menggunakan Pembobotan TF-IDF dan K-Nearest Neighbor," *Journal of Intelligent System and Computation*, pp. 43-49, 2019.
- [8] F. Septianingrum and A. S. Y. Irawan, "Metode Seleksi Fitur Untuk Klasifikasi Sentimen Menggunakan Algoritma Naive Bayes: Sebuah Literature Review," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, pp. 799-805, 2021.
- [9] C. Ardiansyah, Indriati and Marji, "Klasifikasi Emosi pada Komentar YouTube Menggunakan Metode Modified K-Nearest Neighbor(MKNN) dengan BM25 dan Seleksi Fitur Chi-Square," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, pp. 1027-1032, 2020.
- [10] B. Agarwal and N. Mittal, *Prominent Feature Extraction for Sentiment Analysis*, Switzerland : Springer International , 2016.
- [11] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," *J-SAKTI*, pp. 697-711, 2021.